Software development for 3D digitising data in reverse engineering

V. Carbone, E. Savio, J.J. Yang, G. Berti
DIMEG, Università degli studi di Padova, via Venezia 1, 35131 Padova

Summary

Digitising systems that integrate mechanical and optical scanning technology offer the potential advantage of shorter data acquisition if compared with other methods such as manual digitisation using CMMs. They have as output a big amount of points even for a moderately complex part. This oversized information could be undesirable during the reconstruction of CAD models as required in reverse engineering. Furthermore, the distribution of digitised points could not meet the local curvature trend of the physical model. In this paper, algorithms that allow the manipulation, filtering and ordering of digitised points using not contact systems are discussed.
The methodology is implemented in software and can be applied to objects of different shape and size. It automatically creates interpolating spline curves used in CAD modelling to define the natural boundaries of the component. The implemented procedures are discussed in detail in the paper. Some industrial case studies are then presented to validate the approach and to show the applicability of the proposed algorithms.

Key words: reverse engineering, CAD modelling, digitalisation, CAD/CMM integration.

1. Introduction

Reverse engineering (RE) is the process that allows to generate CAD models from an existing physical object.
There are several application areas of the reverse engineering process. In general, the aim is to define the detailed shape characteristics of a product in order to facilitate and shorter its development and engineering process.
Reverse engineering is usually performed as an interactive process where the reconstruction of the 3D CAD model is necessary and no drawings are available. In other cases, a CAD model already exists, but the component has been modified,

i.e. on the shop floor, and it is needed to put back the changes into the CAD models.
In most current reverse engineering systems, basically two main functional steps are required.
The first step is represented by the digitisation of surfaces on the physical model. A 3D mapping of the product is obtained through the acquisition of the *x*, *y* and *z* co-ordinates of multiple points on the product surface.
The second step of RE is represented by CAD modelling operations of surfaces corresponding to the features that need to be joined with positional, tangent or curvature continuity. It can result to be critical, especially in the choice of segmentation and definition of complex, free form surface boundaries [1].
There are several different methods by which surface data can be acquired [2]. In order to short RE process lead-time, different optical, such as stereo image detectors, structured lighting devices, laser systems, and mechanical systems, using contact probes, have been developed. These digitisers may be characterised by a redundant amount of data, i.e. in flat areas, that are not suitable for specific applications in RE [3-4]. In fact, most CAD/CAM systems are not able to map and process the copious data files into surface models, especially when the 3D data of the digitised workpiece are unstructured and the density of the acquired points does not meet the surface trend.
This paper describes a developed software, part of reverse engineering system which uses cooperative sensor integration between a vision system and a CMM equipped with a touch probe [5]. The software is an intelligent interface between the digitising systems and the CAD environment, where complex surface forms can be easily reconstructed invoking the appropriate surface modelling routines [6, 7, 8].
The approach implemented in the software enables to construct the functional curves in an intuitive and interactive way and to produce the final mathematical model directly in commercial CAD systems.

2. The software

2.1. *Interface*

After the rough data have been acquired by means of an optical or mechanical digitising system, they have to be *massaged* and converted in organised data before being imported in CAD environment and other manufacturing software [9]. In fact, inadequate data manipulation may cause the loss of important information, and a good support in 3D point editing and manipulation is required.
The reconstruction of functional curves is essential in the first stage of RE process, when geometric data is not available. They are used to define the rough surface CAD model that has to be re-digitised in a second phase.
The developed software, namely Visual Point, is an interactive program for the manipulation of cloud of points during the pre-processing phase. It is recommended in case of points from random and/or manual sampling, when the data are not structured in a sequence of scanlines. The aim of the implemented functions is to improve the automatization in 3D CAD model reconstruction.
The following features are available:

- an interface for decomposing/composing the set of points in subsets, that can be manipulated and assembled;
- algorithms for reconstructing automatically curves in 3D CAD environment from unstructured sets;
- different filters adapted to typologies of data sets;
- an interface between the different digitisers involved and the CAD system.

Particularly, the data can be stored as curves in order to reduce the necessary processing operations during the surface reconstruction. The formats supported by the software in input and output are: *.ibl*, *.vda*, *.iges* formats.
The user interface is windows-based where point clouds are visualised, rotated and translated. It is possible to open many editing windows and part of the data can be dragged to any location with the mouse pointer using the operations cut, copy and paste. In Figure 1, a typical editing window that contains some information about the set of points (i.e. the points number, the min. and max. point, the origin point, etc.) is shown.

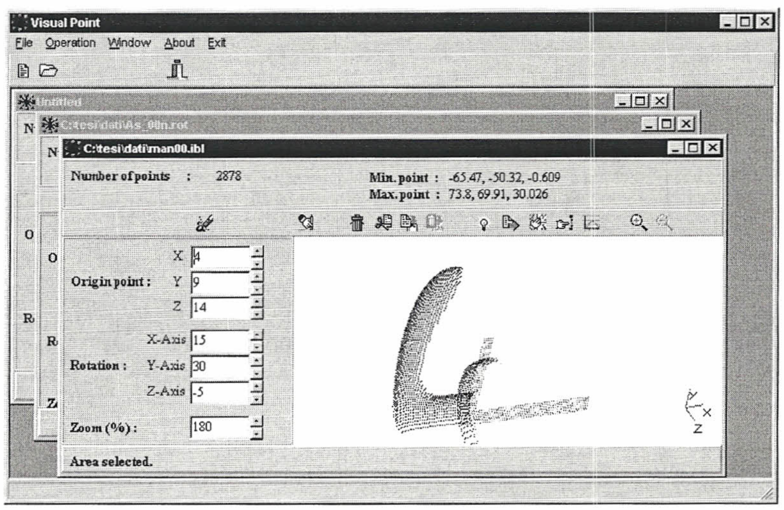

Figure 1. User interface.

2.2. *Features*
The software structure is represented by the flowchart shown in Figure 2.
The main routines are implemented in the *Point tool* module, which is used as basis for organising and structuring the data files. By means of *Points* and *Zone* modules, the point clouds are manipulated and grouped into sets and the scancurves are automatically defined. *Point translation*, *Point processor*, and *Visual poi*nt, are the applications developed for the visual interface.
The system has an interface with PRO/Engineer and its module Pro/Scan Tools dedicated to reverse engineering.
The implemented functions will be discussed in the following sections.

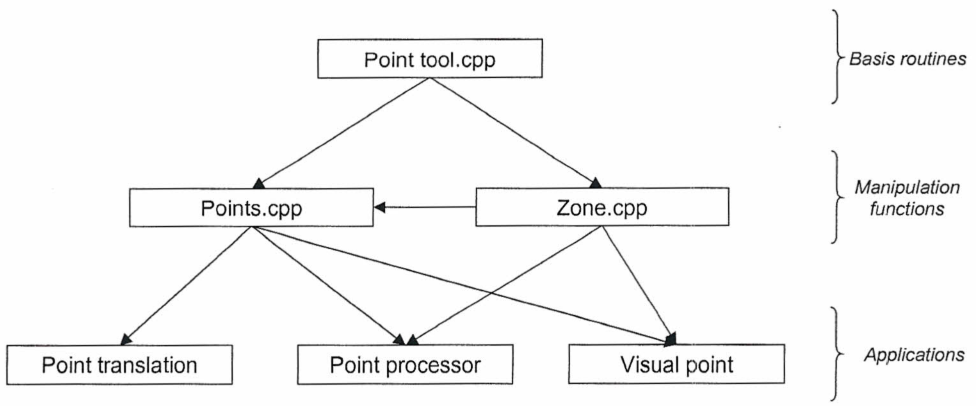

Figure 2. Visual Point software: the structure.

2.2.1 *Function: order*

A remarkable problem in the reconstruction of surfaces is that some CAD algorithms need ordered point sets to reconstruct scancurves [10]. In fact, digitised point clouds can be very far from data structures for representing shapes.
To overcome this difficulty, an algorithm of ordering unstructured point clouds is proposed. The function requires as specific parameters a reference plane and an ordering direction to divide the cloud in a set of bands (see Figure 3).
The enveloping direction of the bands is perpendicular to the defined ordering direction and belongs to the reference plane. The band width in the data file is calculated using an iterative procedure. In this way, scanning points, manual points and in general data from optical system, are automatically divided in bands and ordered along different directions. After that, the versus of adjacent curves is alternatively changed. This allows to reduce the human intervention in the curves reconstruction. The data are thus exported in the selected CAD format.

2.2.2 *Function: extract*

This function allows the areas segmentation and extraction from the original cloud of points. A series of windows are created where the extracted subsets are visualised in different colours.

2.2.3 *Functions: decimate and filter*

An ideal digitiser should collect more points at high curved surfaces in reasonably short periods of time and with a good accuracy. Unfortunately, such a device does not exist at present and the density data acquisition is not adaptive [11].
Some acquisition systems, as those based on the optical technology, are characterised by a big amount of points. Sometimes, this overabundance of information results to be undesirable, especially if interpolation methods are employed for identifying curves features.

The function *decimate* performs a non-homogeneous reduction of the cloud and is useful in the first rough data manipulation. No control is applied and points are randomly deleted. Consequently, the digitised data could loose surface characteristics and become invalid for further applications.
In order to perform an intelligent deleting operation, the filter function has been developed. Two different methods are available:

- tangent method;
- curvature method.

A set of interpolating cubic splines is reconstructed connecting the points that belong to each band. Then, the implemented functions remove the data characterised by a curvature or tangent no contained within a defined range.

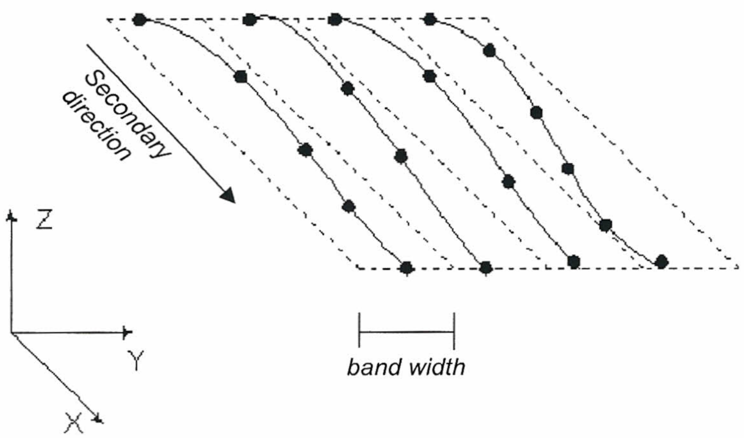

Figure 3. Reconstruction of the scan curves.

3. Application examples

Two industrial cases have been investigated to show the applicability and the validity of the proposed approach. Unstructured data files from a vision system and scanning points from a CMM have been used for testing the algorithms.
The aim is to produce a higher level representation of the shape of the object, by means of scancurves. The software provides accurate tools for the sculptured surface reconstruction using curves set interpolation obtained by choosing the number of control points equal to (or less than) the number of measured points contained in the scancurves. In most reverse engineering applications, interpolation surfaces are not desired because of random errors contained in the cloud of points, thus scancurves can be used as scan data for approximation within a given tolerance. In both these cases, the existing usual work time for defining the initial CAD data model, which takes account of the peculiarities of free-formed surfaces, is greatly reduced.

3.1 Example 1: door handle

A first test used a handle (see Figure 4). The scancurves were automatically defined in the software and transferred into PRO/Scan Tools module. The rough CAD model was constructed using a curve reverse engineering approach. The final surface was then a result of isoparametric curves blending.

No limitations have been found in areas characterised by tight curvature.

3.2 Example 2: turbine blade

The sculptured surface of a turbine blade was another example of surface reconstruction using the proposed system. The scancurves were imported in the CAD environment in order to define style characteristic curves, which might be sharp edges of the boundary, lines of symmetry, lines of critical curvature. Furthermore, non uniform rational B-splines (NURBS) surfaces have been reconstructed and blended in order to generate the solid model.

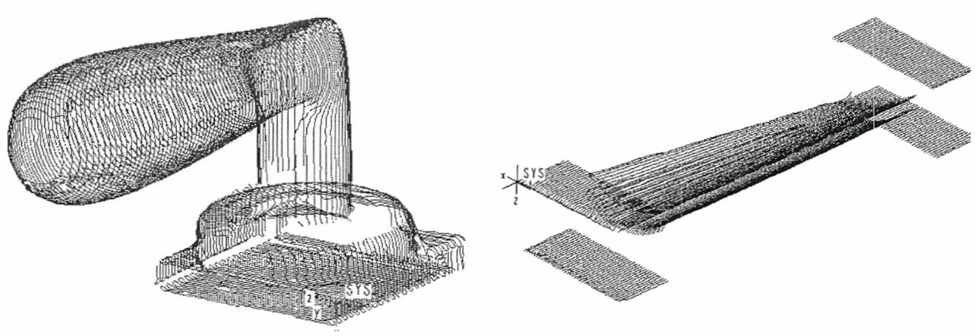

Figure 4. Test objects: the handle and the turbine blade.

4. Conclusions

Capturing shape and translating it into a mathematical model is a difficult task in the reverse engineering process, particularly when complex surfaces are involved. Moreover, huge point sets are quite common when working with scanners and vision systems, and most CAD environments can not handle and parameterise redundant data that needs to be manipulated prior to the modelling operations.

The developed procedures implemented in the software facilitate the conversion of measured points with noise and random distribution into scancurves that represent the basis for NURBS curves and surfaces reconstruction. Thus, the CAD modelling features accepts the 3D data files in any distribution, regular, irregular or random.

Two experimental results have been presented to verify that by means of the procedures implemented in the software the long time required by the manual intervention in case of copious points cloud can be effectively reduced.

Acknowledgements

The authors would like to acknowledge Prof. G. Sansoni, Mr. R. Rodella and Mr. M. Carocci of the Departments of Optoelectronics, University of Brescia, for their fundamental support in the project. Special thanks are due to Mr. M. Mocellin, for his help in software development.

References

[1] Kruth, J. P. and Kersten, A.; 1998, Reverse engineering modelling of free-form surfaces from point clouds subject to boundary conditions, Journal of Materials Processing Technology, 76: 120-127.

[2] Chen, L. C. and Lin, Grier C. I.; 1997, An integrated reverse engineering approach to reconstructing free-form surfaces, Computer Integrated Manufacturing Systems, 10, 1: 49-60.

[3] Bidanda, B. and Hosni, Y. A.; 1994, Reverse engineering and its relevance to industrial engineering: a critical review, Computers ind. Engng., 26, 2: 343-348.

[4] Chen, L. C. and Lin, Grier C. I.; 1997, A vision-aided reverse engineering approach to reconstructing free-form surfaces, Robotics & Computer - Integrated Manufacturing, 13, 4: 323-336.

[5] Carbone, V., Savio, E., Carocci, M., Sansoni, G. and De Chiffre, L., Combination of a vision system and a coordinate measuring machine for the reverse engineering of free-form surfaces, submitted to Int. J. Adv. Manuf. Technol.

[6] Motovalli, S., Suharitdamrong V. and Alrashdan, A.; 1998, Design model generation for reverse engineering using multi-sensors, IEE Transactions, 30: 357-366.

[7] Lai, J. Y. and Lu, C. Y.; 1996, Reverse engineering of composite sculptured surfaces, Int. J. Adv. Manuf. Technol., 12: 180-189.

[8] Kersten, A., Kruth, J. P. and Janssens M.; 1997, Reverse engineering of free form objects: an automatic solution, 30th International Symposium Automative Technology and Automation Mechatronics/Automative Electronics, 1: 101-109.

[9] Bidanda, B., Motovalli, S. and Harding, K.; 1991, Reverse Engineering: an evaluation of prospective non-contact technologies and applications in manufacturing systems, Int. J. Computer Integrated Manufacturing, 4, 3: 145-156.

[10] Ueng, W. D., Lai, J. Y. and Doong, J. L.; 1998, Sweep-surface reconstruction from three-dimensional measured data, Computer-Aided Design, 30, 10: 791-805.

[11] Várady, T., Martin, R. R. and Cox, J.; 1997, Reverse engineering of geometric models - an introduction, Computer-Aided Design, 39, 4: 255-268.