Computers in Industry 48 (2002) 241–251

COMPUTERS IN INDUSTRY

www.elsevier.com/locate/compind

Probe-radius compensation for 3D data points in reverse engineering

Shuh-Ren Liang[a], Alan C. Lin[b,*]

[a]Department of Mechanical Engineering, Yung-Ta Institute of Technology and Commerce, Ping Tung 909, Taiwan
[b]Department of Mechanical Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan

Received 22 January 2001; accepted 8 February 2002

Abstract

The very first step in the process of reverse engineering is to acquire data points by measuring the points on the surface of an object. The fineness and correctness of the data significantly affect the quality of the final CAD model as well as the machined workpiece. However, the point data measured with a typical coordinate measuring machine (CMM) have often been the data of the probe center rather than those of the object's surface, and this poses a great challenge in reverse engineering. This paper, thus, aims to resolve a practical problem in reverse engineering: what is the best way to compensate the measured data points with the value of probe radius? An approach to connecting corresponding points between every two adjacent measuring trajectories is proposed to formulate triangular meshes directly from the massive data points, so as to determine the normal vector of each point for probe-radius compensation. In the paper, the methodology and algorithms to implement the formulation of triangular meshes will be presented in detail with expository examples.

Keywords: Reverse engineering; CAD/CAM; CMM

1. Introduction

Over the past few years reverse engineering has been a very popular field of research. The technological processes involved include the coordinate measuring machine (CMM), or other specialized measuring devices for implementing point-data measurement. Data points are generally taken in high density to reflect the real shape of the measured object. Upon obtaining the point data, the following two processes are typically applied: (1) generate the NC cutter path directly from the point data, or (2) input the point data into CAD software packages to build curves and surfaces, then proceed with the surface modification or engineering analysis, and at last generate the NC cutter path. In either of the methods, the measured point data have to reflect the surface of the measured object. However, the point data gathered by the CMM often are probe-center data, rather than the surface data of the measured object. It, thus, becomes a need to compensate the probe radius before the measured point data can be used to undertake various tasks in reverse engineering.

As for the compensation of probe radius, most of the previous studies are on the basis of the CAD model. For instance, Duffie and Feng [1] built a parametric surface with the point data, and then used the least-square method to obtain an offset surface.

* Corresponding author. Tel.: +886-227-37-6495;
fax: +886-227-37-6495.
E-mail address: alin@mail.ntust.edu.tw (A.C. Lin).

0166-3615/02/$ – see front matter
PII: S0166-3615(02)00038-6

While Jeong and Kim [2] proposed to slice the 3D CAD model to obtain a number of 2D contours, and then offset the contours with discrete distance maps in order to obtain the NC cutter path. In order to avoid gouging during NC cutting, Chen and Ravani [3] stated one should first identify the intersecting surfaces, and then use these to obtain the offset surface. The above-mentioned methods are all based on parametric surfaces to produce the offset surface. But when handling a large quantity of measuring data it is extremely difficult to construct parametric surfaces directly, thus, making the above methods unfit for constructing the offset surface. This research, therefore, proposes a methodology, which receives as input massive 3D point data and develops a point-data conversion algorithm, without reconstructing the surface model, to find every point's normal direction. Each measured data-point is then compensated by offsetting along its normal direction with the value of probe radius. The compensated data facilitate the subsequent tasks in reverse engineering, such as direct generation of NC cutter path or construction of 3D CAD model.

As for the procedure proposed to automate the compensation of massive 3D data points, the point data are firstly input and sorted according to the moving direction of the probe (x- or y-direction). After that, triangular meshes are built, and the normal direction of every triangular mesh is determined. Finally, the compensation of the point data with the value of probe radius is performed in order to find the contact points between the probe and the measured object. Obviously, the major benefit of the approach is the elimination of tedious, time-consuming, and professional-skill-required tasks in CAD modeling. This is especially true when the measured object is of a complex shape. In addition, since the input data points are massive and the pitch of data points is very small, the triangular-mesh model built using the proposed method will retain tiny features of the object, resulting in high accuracy in the overall processes of reverse engineering. One thing to notice is that the developed method is practically usable if the 3D points are distributed densely enough. For cases that 3D points are sparsely distributed or the measured surface is very coarse, CAD models still have to be built in order to find normal vectors of all data points for probe-radius compensation.

What follows is a detailed explanation of methods proposed for the following three tasks: sorting of point data, construction of triangular meshes and compensation of probe radius.

2. Sorting of measured point-data

Fig. 1 illustrates the basic concept of sorting of point data. The input raw point data do not undergo compensation of probe radius, and thus, are rather messy. Sorting aims to regain regularity of data points on the surface of the measured object, and facilitate the subsequent calculations. The QuickSort algorithm by Sedgewick [4] was one of the approaches proposed to solve the sorting problem, which cut apart the whole data array with the so-called divide-and-conquer recursive method. However, the algorithm was developed primarily for points with 1D coordinates. It was, therefore, revised in this study to meet the requirements of sorting massive data points in the form of 3D coordinates. To avoid misunderstanding, explanations of technical terms are given in advance (Fig. 2):

(1) Measuring trajectory: A trajectory formed by point clouds in a finish of measurement. The points on each measuring trajectory possess the

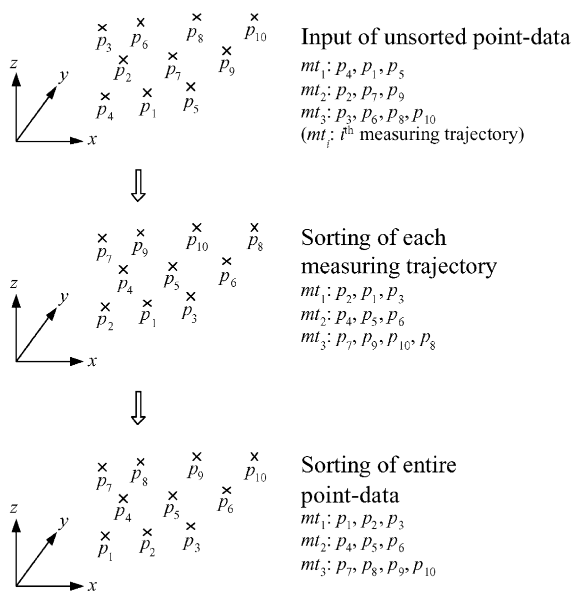

Fig. 1. Basic concept of sorting of 3D point data.