Table of contents

Off-line programming of coordinate measuring machines using a hand-held stylus

Author: Medeiros, D J; Thomas, G; Ratkus, A B; Cannon, D

Publication info: Journal of Manufacturing Systems 13.6 (1994): 401.
ProQuest document link

Abstract: A new method for programming a coordinate measuring machine (CMM) is described. A six degrees of freedom position-sensing device is used to identify measurement pints on an actual part, thus creating a CMM program. Experimental results indicate that the position-sensing device, although not accurate enough for part inspection, possesses sufficient accuracy for machine programming. These results are verified by writing and executing a CMM program.

Links: Check InfoLink for Availability

Full text: Abstract

A new method for programming a coordinate measuring machine (CMM) is described. A six degrees of freedom position-sensing device is used to identify measurement points on an actual part, thus creating a CMM program. Experimental results indicate that the position-sensing device, although not accurate enough for part inspection, possesses sufficient accuracy for machine programming. These results are verified by writing and executing a CMM program.

Keywords: Coordinate Measuring Machine, Machine Programming, Inspection, Measurement

Introduction

Coordinate measuring machines (CMMs) have traditionally been used for qualification of received parts and for inspection of manufactured parts to assure that geometric design tolerances are achieved. They improve the flexibility of inspection operations by reducing the need for fixturing and special gauges, thereby enabling inspection of a large variety of small-quantity items. Many CMMs are computer controlled, allowing storage and automatic execution of inspection programs.

More recently, CMMs have begun to play a pivotal role in assuring that manufacturing processes are in control. This new focus has caused CMMs to be more tightly integrated into total quality efforts and to become closely associated with the shop floor. sup1

Use of CMMs for process quality monitoring and the trend to small-batch manufacturing results in increased demands on CMMs. Programs for new parts must be created quickly, and CMM availability must be increased. Most CMMs are programmed by driving the probe to contact the part at desired points; sup2 this process is tedious, exacting, and reduces the time available for part measurement.

Off-Line programming systems can resolve the problems of traditional programming methods; several such systems have been developed. sup3-5 Off-Line programming uses a CAD database along with additional software; the approach is similar to the CAM systems used for programming CNC machines and the workcell simulation systems used for programming robots. Like these systems, a CAD model of the part (and sometimes the CMM itself) is required. Typically these systems involve selecting the feature to be measured by pointing to the appropriate geometry on the display; selecting a probe; defining the tolerance type, tolerance limits, and number of points to be used in measuring the feature; and verifying that the calculated probe path does not collide with the part, fixture, or CMM frame.

This paper presents an alternative off-line programming method for which no CAD model is required. Instead, programming is done using an actual part and a hand-held stylus equipped with a position-sensing device. An operator identifies the type of feature to be inspected and manipulates the stylus to define the points on the feature to be measured (Figure I). (Figure 1 omitted). The position-sensing system incorporated in the stylus calculates the location of the tip and orientation of the stylus. This information is recorded on a portable computer connected to the sensing device. After all features are measured, the recorded data is converted into a CMM program and downloaded to the CMM.

Because an actual part is used for off-line programming, the method is also applicable to reverse engineering, which is constructing a model from an existing specimen. CMMs are frequently used for reverse engineering; sup6-7 depending on part geometry, on-line programming can be difficult and time consuming.

The position-sensing system used for off-line programming is the 3Space(R) Isotrak(R) system from Polhemus Electronics Co., referred to in this paper as the Polhemus device (PD). The PD outputs the position and orientation of a movable sensor, sup8-10 which in this case is attached to a stylus. The position and orientation information (relative to a fixed source) is provided in the form of a 4 X 4 homogeneous transformation matrix. Both the fixed source and the movable sensor are connected to an interface unit, which can in turn be connected to a computer via a serial connection (see Figure 1).

The PD and other similar devices have been used for several years in medical experiments as a non-invasive technique to measure the three-dimensional movement of bones, such as those in the wrist. sup11-14 These devices offer a unique way to circumvent the difficult process of deriving three-dimensional motion from analysis of static CAT scans or inherently two-dimensional X-ray images.

Because of limited precision and sensitivity to electromagnetic noise and ferric materials, the present level of PD technology is not suitable for part inspection; however, its flexibility and portability offers great potential as a programming tool for more accurate inspection devices such as CMMs. The most time-consuming task in programming a CMM is defining the probe points and intermediate points for feature measurement, a task that can be accomplished quickly and easily using a hand-held stylus.

This paper first describes the new off-line programming methodology. Next, issues related to analysis of PD performance are addressed, including a calibration method and an assessment of PD accuracy. Creation and execution of an inspection program to test the methodology is then described. Finally, future research activities are discussed.

Off-Line Programming Methodology

O-line programming using the PD is pet-formed by completing three tasks in sequence. The first task is to create an inspection program using the PD to identify measurement points by touching the appropriate locations on the part. A computer interfaced to the PD records the feature type and the sampled points in a generic format. The second task is to postprocess the inspection information to create a program in the format required by the CMM. Finally, the inspection program is downloaded to the CMM and the part is inspected.

Creating an Inspection Program

This CMM off-line programming system relies on identifying locations at the tip of the measuring stylus relative to a coordinate frame defined on a part. The determination of this relative measurement is achieved through a series of three steps. The first step finds the relative position of the pointed end of the stylus with respect to the PD sensor, which is affixed to the opposite end; it is described in the section on stylus calibration. Using the calibrated stylus, the second step defines the part coordinate system relative to the PD source (see the section on defining the part coordinate frame). The third step creates the inspection program by identifying the feature to be measured, positioning the stylus at several points on the feature, transforming these points to the part coordinate frame, and recording the points, as discussed in the section on measuring features.

Stylus Calibration

A nonmetallic stylus holds the sensor. The stylus is 20 cm long, with the PD sensor (a cubical object approximately 3 cm in size) mounted at one end. The PD reports the relative position and orientation of the sensor's coordinate frame with respect to a base coordinate system defined by the location of the source. Figure 2 illustrates the various coordinate frames used. (Figure 2 omitted). Unfortunately, the origin of the sensor's coordinate frame is physically inaccessible because it is embedded inside the sensor's resin and plastic protective cover; however, because the sensor is secured to the stylus, the stylus tip is located at a constant relative position with respect to the sensor. The location of the tip of the stylus, therefore, can be calculated by estimating this constant offset vector and applying a homogeneous transformation. sup15 Note also that a vector representing a useful measuring approach angle for the CMM probe may be obtained from the orientation of the stylus.

To find the offset vector, the stylus tip is held in a fixed location while the stylus is rotated. The sensor position is sampled at several orientations of the stylus (Figure 3). (Figure 3 omitted). Two or three different sensor locations must be sampled, but in practice a larger number of points is used to achieve greater calibration accuracy. The constant offset vector can be determined by applying the method of least squares. sup6

The linear regression contains six unknowns: the three stylus offset vector coordinates relative to the sensor's coordinate frame, OV subx , OV suby , and OV subz , and the three coordinates of the stylus tip with respect to the base coordinate system, C subx , C suby , and C subz . Because the tip of the stylus is not moved during calibration, the point C = (C subx , C suby , C subz) is constant. The PD provides a 4 X 4 homogeneous transform, supB T subs , from the sensor coordinate frame to the base (source) coordinate system. Assuming perfect accuracy in the PD and no slippage of the stylus tip, the following relationship would hold:

C = (supB T subs) OV

Expanding with the matrix notation of Fu et. al., sup17 where: (Formulas omitted).

At this point, the stylus is calibrated. The calibration vector is stored for use in defining the part coordinate frame and creating the inspection program.

Defining the Part Coordinate Frame

Before creating a part coordinate frame, the part to be inspected is located at a convenient position and orientation, and the PD source is positioned at an appropriate location. The source and the part are not moved until programming is complete.

The part coordinate frame is used to locate measured points and is also used in setting up an alignment system on the CMM. All stylus measurements are described with respect to this part reference frame, which during programming is defined with respect to the base coordinate system, as illustrated in Figure 2.

A simple technique for defining a part reference frame was implemented. The part reference frame is assumed to coincide with the intersection of three orthogonal planar surfaces on the part. Thus, it is assumed that the part has three planar datum surfaces that define its location and orientation for the purposes of measurement. One of these datum surfaces is selected to represent the z = 0 plane. Measurements are made by placing the stylus tip at several positions on the surface; linear regression is used to obtain the plane equation. The process is repeated for two additional orthogonal planar surfaces. The origin of the part reference frame is located at the mutual intersection of the three fitted planes. The X-axis lies along the line formed by the intersection of the first and second fitted planes. The Y-axis lies in the first fitted plane and is perpendicular to the X-axis. The Z-axis is defined as the cross product of the X and Y axes.

A 4 X 4 homogeneous transformation supB T subp is calculated to relate the part coordinate frame to the base coordinate frame. After moving the stylus to a position and reading the transformation matrix supBT subs output by the PD, a point P in the part coordinate system may be calculated as follows:

P = (supB T subp) sup-1 (supB T subs) OV

where OV is the 4 X 1 stylus offset vector, with the last row containing a 1.

Measuring Features

The user identifies a feature to be measured by selecting the feature type from a menu of choices on the computer screen and then positioning the stylus at the desired probe points on the actual part feature. Because the CMM will move in a straight line between points (ignoring a small offset on probe points to be described later), intermediate points must be used to ensure a collision-free path for the probe. Intermediate points are programmed in the same manner as probe points: the user selects the intermediate point menu item and positions the stylus at the desired location of the intermediate point. Depending on the part geometry, several intermediate points may be required.

Presently, measured points, planar and cylindrical features, and intermediate points have been implemented, although other features can be easily added. The probe points and intermediate points are computed in the part coordinate system, as described above.

In addition, an approach vector is required for each of the probe points (but not for the intermediate points). The approach vector is used by the CMM to determine the direction in which to move toward the programmed probe point. The CMM will move at full speed to a point offset a predetermined distance from the programmed point along the approach vector, then move at a slower measuring speed along the approach vector until the probe tip contacts the surface of the part.

The approach vector is calculated from the normalized offset vector after it is transformed to the part coordinate frame. This implies that the probe will approach the target point along the line from the sensor origin to the stylus tip. Thus the position of the stylus tip determines the point to be measured, and the orientation of the stylus determines the approach vector for this point.

For each feature, the feature type and its associated intermediate points, probe points, and approach vectors are stored in an ASCII file in a generic format.

Postprocessing the Inspection Program

Because the programming methodology described here was implemented for a specific inspection machine (a Numerex Minicoord DCC CMM), the program was translated from the generic ASCII form directly into the binary format required by the machine. An alternative approach that would support more CMM types would be to use the dimensional measurement interface specification (DMIS). sup18 The DMIS is an ANSI standard for bidirectional transmittal of inspection programs between CAD-based programming systems and CMMs and between different manufacturers' CMMs.

Running the Inspection Program on the CMM

The translated inspection program contains a list of features to be measured, probe points, and intermediate points. Additional information is required before running an inspection program. This information includes probe calibration, part alignment system, machine speed and prehit distance, and tolerances.

The probe tip may require calibration; this is done using a calibration routine on the CMM. An alignment system must be established that corresponds to the part coordinate frame used while writing the inspection program; this can be accomplished by measuring the same three planes on the part using the CMM. It may be desirable (particularly when running a program for the first time) to adjust the measuring speed and positioning speed of the CMM.

The prehit distance is an offset from the programmed point along the approach vector; the CMM will move at positioning speed to the prehit distance, then move at measuring speed until the probe contacts the part. The prehit distance should be specified based on the geometry of the part, the desired inspection time, and the estimated accuracy of the points measured using the PD.

Finally, desired tolerance types and limits on the features must be specified, and additional geometric elements constructed from measured elements if necessary. The CMM software provides the capability to easily add this information in an off-line editor, so this approach was taken to simplify the use of the PD-based programming system. The PD system provides the required geometrical information while the tolerance information (and any constructed features) are added using the CMM program editor.

The approach minimizes downtime for programming the CMM. All of the items discussed above may be performed off-line, with the exception of calibration and setting up an alignment system. Calibration is required whenever a probe is changed unless an autochange probe system is used. An alignment system must be established for each part loaded on the machine unless dedicated fixturing is employed. By far the most time-consuming task in programming a CMM is defining the probe points and intermediate points for feature measurement; this task is completely removed from the CMM and greatly simplified by use of the PD-based programming system.

Analysis of PD Performance

Because the PD has limited precision and is known to be sensitive to electromagnetic noise and ferric materials, some testing is required before using the device in this application. It is necessary to determine that the device (and the system that uses it) can generate data that is accurate enough, not to characterize the part, but to program the CMM, which is responsible for the actual characterization. Accordingly, a number of experiments were conducted with the PD. Described below are experiments with stylus calibration, device repeatability, source position, and part material. All experiments were conducted using a laptop personal computer with an LCD screen to avoid any electromagnetic interference from a CRT monitor.

Calibration Experiment

The first experiment was designed to determine if there were any significant factors affecting the calibration of the stylus. The experiment was a 3 sup3 factorial design. The factors involved were the setup of the Polhemus device, the position of the stylus while collecting the data points, and the number of data points collected during calibration.

The setup of the PD consisted of changing the position of the source relative to the sensor. The source was mounted above the sensor/stylus, below the sensor/stylus, and beside the sensor/stylus. The source was approximately 30-40 cm away from the sensor in all three positions. The position of the stylus while taking the points during calibration was also varied. Although the tip of the stylus remained in a constant position, the stylus was rotated in a circular motion in a "near vertical" position, "near horizontal" position, and in both positions. Finally, the number of points taken while calibrating the stylus was varied from 5 to 20 to 50.

The response measured was the length of the offset vector. The origin of the reference frame for the sensor is unknown; therefore, the true length of the offset vector is also unknown. The experiment was a completely randomized 3 sup3 fUll factorial design with five replications. An analysis of variance (ANOVA), shown in Table 1, indicated that the setup of the PD, the position of the stylus during calibration, and the number of points taken during calibration were significant factors. (Table 1 omitted).

Results of this first experiment showed that when the source is positioned below the sensor/stylus, the mean offset vector is larger than with the source in the other two positions (Table 2). (Table 2 omitted). This difference may be due to the fact that the source is slightly further away from the sensor when it is in the below position. (The accuracy of the PD decreases as the distance between the source and sensor increases.) The mean offset and standard deviation are larger when five points (rather than 20 or 50) are used to calibrate the stylus. When the stylus is rotated in both the vertical and horizontal positions, the mean offset vector is slightly larger than in the other two cases. These results indicate that two points are probably not sufficient for calibration. Despite the slightly higher offset vector, we believe that moving the stylus through a full range of motion while calibrating may result in a better estimate of the offset, although this was not rigorously tested.

The two-way interaction between the setup of the Polhemus device and the number of points taken during calibration was found to be significant, but the interaction effect occurs when calibrating with five points, which was shown to be insufficient. The three-way interaction is also a result of the poor calibration achieved with five points.

Repeatability Experiment

A second experiment was conducted to determine the inherent variation in the PD when collecting data points because prior use of the device indicated that readings varied even though the stylus was held in a constant position. This second experiment consisted of collecting a number of data points while holding the tip of the stylus in a constant position on a nonmetallic part. One data point was measured 50 times without moving the stylus and 50 times by rotating the stylus while keeping its tip in a divot on the part. This procedure was done three times; each time the position of the source was changed with respect to the sensor using the same positions as in the first experiment.

This experiment resulted in a range of approximately 2 cm when the stylus is rotated compared to 0.1 cm when the stylus was held still. Results were consistent under all positions of the source. This variation should be considered when using the PD to write an inspection program for the CMM. The user should recognize that this variation will occur when sampling single points and compensate for this on the CMM itself by changing the prehit distance. If increased accuracy is needed, the user may repeatedly sample the same point with the PD and use the average of the points in the CMM program.

Material vs. Sensor Position Experiment

Next an experiment was designed to determine the amount of variation that is present when measuring an aluminum piece as opposed to a wooden part or any other nonmetallic material. The wooden part was approximately 30 X 15 X 3 cm, while the aluminum part was approximately 13 x 10 X 1.5 cm. Both parts contained several holes. The two factors considered were the material of the part being measured and the position of the source relative to the sensor. The response was the measured distance between points known to be 2.54 cm apart. When measuring one point, five readings were taken at that point and the average of those readings was used. The experiment was completely randomized and repeated three times.

An ANOVA indicated that the significant factor was the setup of the PD and that the best setup is with the source positioned above the sensor. When the source is positioned above the sensor, the aluminum part did not lie between the sensor and the source. Average errors were 0.71, 3.95, and 2.73% when the source was above, below, and beside the sensor, respectively.

When measuring a metallic part, the position of the source relative to the sensor should be considered. The results of this experiment indicate that the best setup results when the source is positioned in such a way that the metallic part has minimal interference with the electromagnetic field, as expected.

Gauge R&R Study

It was desired to know if different operators had any significant effect on the output when using the PD. A gauge repeatability and reproducibility (R&R) study was conducted. A gauge R&R study breaks down the measurement variance into two components: variance due to the operator and variance due to the measuring device. The study consisted of three operators (one experienced with the system and two inexperienced) measuring three different points on a nonmetallic part. Each operator measured each point repeatedly.

The percentage of measurement variance due to the operator was 5.95%, while the percentage of measurement variance due to the PD was 94.05%. Thus the operator is not a significant factor in using the system.

Next, each operator calibrated the stylus five times, and the length of the offset vector was measured. A one-way ANOVA was used; the effect of different operators on stylus calibration was not significant.

Example of Programming Methodology

Experimentation with the PD clearly demonstrated that the device is not accurate enough for measurement of part features. To demonstrate that it can be used for programming, two simple inspection programs were created and executed on a Numerex Minicoord CMM Model 1398-10. An aluminum part was selected and two features were measured: a 15 X 15 cm planar surface and a 2.54 cm cylindrical hole. Prior to making any measurements, the stylus was calibrated using 30 sampled data points.

In writing the first program, one sample was obtained from the PD at each point indicated by the operator. For the second program, three samples were obtained from the PD and averaged to create one inspection point. It was anticipated that this procedure would reduce the variability obtained from the PD.

The first step in creating an inspection program is to create a part reference frame. The part reference frame was defined by measuring three orthogonal planar surfaces, as described earlier. We began by touching the stylus at four to six points on each of the three planar surfaces. The software then fitted a plane to each group of points and calculated the reference zero and axis alignments for the part. We positioned the stylus at the part reference zero and programmed an intermediate point at the stylus location. Because data points are displayed on the computer screen for verification, we could use this method to determine if the calculated part reference zero was correct.

We noted that the intermediate point coordinates were not close to zero (that is, the part reference frame was not correct). The problem arose because we used a small number of points (4-6) to define the planes of the reference system. Because of the inherent variability in the PD, and because the least squares method was used to fit the planes, an error in one of the sampled points can change the orientation of the fitted plane. Because the reference zero was defined as the intersection of three planes, small orientation errors can cause significant movement of the reference zero. After this behavior was observed, a large number of points (20 or more) was routinely used to define the datum planes. The problem did not reoccur.

After a satisfactory procedure was developed for defining a reference system, an inspection program was written. First, we selected the intermediate point menu item and programmed an intermediate point above the planar surface. Next, we selected the plane menu item and moved the stylus to touch four points on the planar surface. Two more intermediate points were programmed: one above the planar surface and one above the cylinder. Finally, the cylinder menu item was selected, and we moved the stylus to touch six points inside the cylindrical hole. (These are the minimum number of points required to measure a plane and a cylinder.)

The postprocessor was run, and the resulting inspection program was downloaded to the Numerex CMM. The part was placed on the CMM table, and an alignment system was set up by measuring the three datum planes. The postprocessor automatically appends instructions at the beginning of the CMM program to move the machine to its home position and recall the defined alignment system, and at the end of the CMM program to return the machine to its home position. A prehit distance of 1.3 cm was specified, and the inspection program was executed. The program ran successfully, measuring the desired features without machine errors, collisions, or missed surfaces.

The procedure was repeated for the second CMM program, except that when the stylus was touched to the part or an intermediate point was requested, three samples were obtained from the PD and averaged to create one probe point. In watching the programs execute on the CMM, it was observed that the second program positioned the probe much closer to the desired points than the first program did. We did not change the prehit distance from the first program, but a smaller prehit distance would have sufficed because of the improved positioning accuracy.

Positioning the probe close to the desired points is especially important when measuring a feature that is very small, such as the inside diameter of a small hole. Accurate positioning is also important for parts with complex shapes or that require fixturing, to avoid collisions with the part or fixtures. The two programs demonstrated that the required level of accuracy is obtainable using our method, though multiple samples may be required at each point if the part or feature being measured is relatively small or complicated.

Economic Considerations

The benefit of this system is that it allows operators to create new CMM programs while the CMM measures other parts. The value of the system lies in the amount of CMM time saved, assuming that there is other work available to utilize the time. In some cases, the programs created may be less efficient than those created using the CMM because of larger prehit distances chosen to protect the CMM stylus from sensor and calibration inaccuracies. Generated programs that take longer to execute will dilute the time savings.

Direct computer-controlled CMMs range in cost from approximately $60,000 to several hundred thousand dollars, depending on size of the machine, type of probe selected, presence of probe autochange capability, and accuracy required. A Polhemus device costs approximately $4000. Because of the wide variation in CMM costs, we will consider time savings from the methodology rather than dollar savings.

To determine if this system would be economically effective for a particular application, we divide the CMM's use into three categories: programming, setup, and execution: Further, separate programming time (P) into two activities: definition of measuring and intermediate positions (TM), and specification of tolerances and administrative information (TS). Let setup (S) consist of fixturing, alignment, and any test runs. Let execution (E) consist of the time to execute a program and inspect a part once the part is in place and aligned.

The off-line programming system will eliminate the time consumed at the CMM for the definition of measuring and intermediate positions (TM), but may increase the execution time (E) due to program inefficiencies. If the program inefficiency (I) is a factor of present efficiency, the additional time available on the CMM is:

$$NewCMMTime = TM - E * I$$

For example, consider a hypothetical small-batch manufacturer that presently spends 20% of its available CMM time in setup, so S = .20. Further assume that 90% of the programming time consists of defining measuring and intermediate points (so TM = .9P) and that for the part types inspected the programs generated by the off-line system will be one-fifth less efficient, so that I = 1.2. The execution time can be expressed as E = 1 - S - P, or .8 - P. One can easily determine the breakeven point, in terms of programming time (P), for saving time utilizing the off-line programming system:

$$NewCMMTime = .9P -(.8 - P) * 1.2 = 0$$

$$P = .46$$

Thus, to justify the off-line programming system based on time savings, a large percentage of time must be spent in programming the CMM for either the current or anticipated part mix. This example indicates that benefits of the system will derive to operations using CMMs for the creation of many programs (as might be found in a small-batch, high-variety environment) or for the creation of complex programs (which require large amounts of programming time). Results are highly dependent on the efficiency factor, which derives from an assumed larger prehit distance. The prehit distance depends on the complexity and size of the part, as well as the accuracy of the PD. We note that accuracy can be improved significantly by averaging several readings from the PD at each point, as discussed in the example (because the PD sampling rate is 60 Hz, additional readings can be easily obtained). Therefore, the efficiency factor is controllable to a certain extent.

Future Research Issues

Future research is proceeding in two directions: application of the present method for surface-based programming and use of virtual reality interfaces. Each of these issues is described below.

Surface-Based CMM Programming

The present off-line programming method may be characterized as point-based programming-the user specifies the specific points that are to be used in measuring a feature. If a large number of points is required, this method could become tedious. An alternative approach is to use the PD to identify a surface to be inspected and to subsequently calculate appropriate points to measure from the surface information. This approach holds particular promise in measuring sculptured surfaces for reverse engineering, but it is applicable to any surface type.

Surface-based programming involves sweeping a stylus over the surface in a regular pattern while sampling stylus position and orientation using the PD continuous sampling mode. After collecting the surface trace data or boundary data, three tasks must be performed: filtering the sampled data points, fitting an appropriate surface to the points, and generating the desired number of inspection points in approximately equal spacing on the surface.

Virtual Reality Based CMM Programming

The idea of tapping the workpiece with a stylus to designate points can be extended using virtual reality techniques (Figure 4). (Figure 4 omitted). In this concept, a human operator flies a virtual tool-an interactive graphic representation of a real tool such as a CMM probe--to designate points of interest. This virtual tool is interwoven in proper depth perspective within video images of the mounted workpiece.

The operator in this virtual reality based point-and-direct (VR-PAD) concept is equipped to make quick programming modifications and to see potential collisions in virtual overlay before sending in the real probe. He or she is able to look and plan before leaping into action with real hardware so collision-prone sequences and probe overshoot are avoided. Programming time, like in our basic stylus approach, remains essentially the time needed for freehand motions. Virtual features also include an engulfing tool to specify regions for the surface-based programming concept described above. Vision-based models, built in conjunction with this concept, allow autonomous CMM programming for routine sequences while human interaction is reserved for designating customized regions and points. A similar VR-PAD interface for programming robots using phrase gestures such as "put that there" is largely implemented, to date, in the Computer Integrated Manufacturing Laboratory of The Pennsylvania State University.

For this futuristic concept, experts need not physically attend each individual CMM machine, and individual machines need not have duplicate programming paraphernalia. From a central workstation the operator writes CMM programs for multiple machines as fast as technicians are able to set up. With this approach, future flexible facilities conduct multiple workpiece inspections in parallel and create models such as CAD drawings for as-built components. Humans contribute reasoning via a natural interface, and machines do rote planning and trajectory execution (at which they excel).

Conclusions

The coordinate measuring machine is a uniquely flexible inspection device. Its method of operation is suitable for manufacturing environments with high-variety, low-volume parts. Difficulty of programming has limited its use in these applications.

We have developed a system that greatly simplifies the programming task while moving it off-line to increase the availability of the CMM for inspection tasks. The method is faster than on-line programming; the user can simply move the stylus to the desired location by hand, rather than using joysticks to drive motorized axes. Programming time is also saved because on-line programming requires rather slow, careful moves when measuring to avoid damage to the probe. If an approach vector normal to the surface is desired (for example, when measuring a sculptured surface or an angled plane or hole), this can be achieved much more easily with a stylus than by moving the machine axes. In addition, machine time that would otherwise be used for writing programs is available for inspecting parts.

Previous off-line programming methods have been CAD based. Our method does not require the quality technician to be a CAD operator. In addition, it does not require a CAD model of the part, and thus it can be used in reverse engineering applications, for inspecting parts manufactured by outside suppliers, and in other cases where a CAD model is not available or it is not desirable to place a CAD system in the inspection department. The inspector can distribute measurement points as desired, rather than interacting with a CAD system to move or delete automatically positioned points.

We have demonstrated that a Polhemus device can be effectively utilized for CMM programming. The accuracy of the Polhemus device is sufficient for some (but not all) CMM programming tasks. The accuracy can be improved by averaging several points to produce each probe point or intermediate point, but this method is limited as well. Recently, other manufacturers have begun to produce a newer generation of devices similar to the PD; some of these are reported to have greater accuracy, more noise immunity, and less sensitivity to metallic objects. The method described herein could be implemented using one of these newer position-sensing devices if greater accuracy were required. Additional capabilities such as remote programming and surface-based measurement can also be added to the base system.

Acknowledgments

This research was partially supported by NSF Grant GER-9023507.

References

1. William Tandler, "Critical Concepts of High Performance Coordinate Measuring Systems," SME Technical Paper MS84-713 (Dearborn, MI: Society of Manufacturing Engineers, 1984).

2. Frederick Mason. "Program Your CMM Off-Line," American Machinist (Oct. 1992), pp 45-47

3. Mark Traband and D.J. Medeiros, "CAD-Directed Programming of a Vision-Based Inspection System," Journal of Manufacturing Systems (v8, n3, 1989), pp 215-223.
4. Hong-Tzong Yau and Chia-Hsiang Menq, "The Development of an Intelligent Dimensional Inspection Environment in Manufacturing," Proceedings of the 1990 Japan-USA Symposium on Flexible Automation (1990), pp 1059-1065.
5. Christopher Garcia, "Driving CMMs," Quality (Aug. 1988).
6. Keytak H. Oh, Robert J. Abella, and James M. Daschbach, "CMM Applications in Reverse Engineering-Integrating CMM with CAD/CAM for Existing Parts Without Drawings," SME Technical Paper MS89-529 (Dearborn, MI: Society of Manufacturing Engineers, 1989).
7. Wai-Lun Kwok and Paul J. Eagle, "Reverse Engineering: Extracting CAD Data from Existing Parts," Mechanical Engineering (Mar. 1991), pp 52-55.
8. J. Kuipers, "Object Tracking and Orientation Determination Means, System, and Process," US Patent 3,868,565 (1975).
9. J. Kuipers, "Tracking and Determining Orientation of Object Using Coordinate Transformation Means, System, and Process," US Patent 3,983,474 (1976).
10. J. Kuipers, "Apparatus for Generating a Mutating Electromagnetic Field," US Patent 4,017,858 (1977).
11. S.J. Bresina et al., "Three Dimensional Wrist Imaging: Evaluation of Functional and Pathologic Anatomy by Computer," Clinics in Plastic Surgery (v13, n3, 1986), pp 389-404.
12. S.E. Logan et al., "Kinematic Analysis of Human Wrist Motion Using a Six-Degree-of-Freedom Digitizer and Computer-Assisted Design System," Surgery Forum, American College of Surgeons (v36, 1985), pp 568-571.
13. M.J. Pearcy and R.J. Hindle, "New Method for the Non-Invasive Three-Dimensional Measurement of Human Back Movement," Clinical Biomechanics (v4, nl, 1989), pp 73-79.
14. C.F. Hildebolt and M.W. Vannier, "Three-Dimensional Measurement Accuracy of Skull Surface Landmarks," American Journal of Physical Anthropology (v78, 1988), pp 497-503.
15. R.S. Hartenberg and J. Denavit, Kinematic Synthesis of Linkages (New York: McGraw-Hill, 1964).
16. G. Strang, Linear Algebra and Its Applications, 3rd ed. (New York: Harcourt, Brace, 1988).
17. K.S. Fu, R.C. Gonzalez, and C.S.G. Lee, Robotics: Control, Sensing, Vision, and Intelligence (New York: McGraw-Hill, 1987).
18. Computer Aided Manufacturing-International, Inc., Dimensional Measuring Interface Specification Version 2.1, R-89-DMIS-01 (Arlington, TX: 1989).
19. Amy Ratkus, "Testing a New Off-Line Programming System for a Coordinate Measuring Machine," MS Thesis (University Park, PA: The Pennsylvania State University, 1993).

Authors' Biographies

D.J. Medeiros is an associate professor of industrial engineering at Penn State. Her interests are in computer-integrated manufacturing, computer simulation, and material handling systems. She holds a BSIE from the University of Massachusetts and an MSIE and PhD from Purdue University. She is a member of IIE and the College-Industry Council on Material Handling Education.

Geb Thomas is a PhD candidate and research assistant in the Department of Industrial Engineering at Penn State. His research interests include telerobotics and computer-integrated manufacturing, especially the design and development of novel information transfer mechanisms for the operator and system. He is a student member of SME and IIE.

A.B. Ratkus is a product engineer in the Plastics and Trim Products Div. of Ford Motor Co. She holds a BSIE and MSIE from Penn State; her specialty is quality control and total quality management.

David J. Cannon received his BS, MS, and PhD degrees in mechanical engineering from Stanford University, with a stint in industry intervening between degrees. Dr. Cannon joined the faculty in the Department of Industrial and Management Systems Engineering at Penn State in 1991. His areas of teaching and research include robotics, flexible manufacturing, control theory, and human-machine systems. He has developed a virtual reality based point-and-direct (VR-PAD) research program that uses virtual reality tools in telerobotics with funding from the National Science Foundation and the Sandia National Laboratories. His industrial experience includes work for Fluor Corp. as a lead engineer and Global Marine Development Inc. as a mechanical engineer. Dr. Cannon has been a member of IEEE, ASME, and AIAA.

D.J. Medeiros, G. Thomas, A.B. Ratkus, and D. Cannon, Pennsylvania State University, University Park, Pennsylvania

Subject: Studies; Pen based computers; Measuring instruments; Computer programming;

Classification: 9130: Experimental/theoretical treatment; 5310: Production planning & control; 5240: Software & systems

Publication title: Journal of Manufacturing Systems

Volume: 13

Issue: 6

Pages: 401

Number of pages: 11

Publication year: 1994

Publication date: 1994

Year: 1994

Publisher: Society of Manufacturing Engineers (publishers)

Place of publication: Dearborn

Country of publication: United States

Publication subject: Computers--Computer Systems, Engineering

ISSN: 02786125

CODEN: JMSYEB

Source type: Scholarly Journals

Language of publication: English

Document type: PERIODICAL

Accession number: 00947757

ProQuest document ID: 197592215

Document URL:
http://proxy.lib.uiowa.edu/login?url=http://search.proquest.com/docview/197592215?accountid=14663

Last updated: 2010-06-06

Database: ABI/INFORM Global

Bibliography

Citation style: APA 6th - American Psychological Association, 6th Edition

Medeiros, D. J., Thomas, G., Ratkus, A. B., & Cannon, D. (1994). Off-line programming of coordinate measuring machines using a hand-held stylus. Journal of Manufacturing Systems, 13(6), 401. Retrieved from http://search.proquest.com/docview/197592215?accountid=14663