A Unique Sensor Fusion System for Coordinate Measuring Machine Tasks

Marilyn Nashman, Billibon Yoshimi, Tsai Hong Hong, William G. Rippey, Martin Herman
National Institute of Standards and Technology
Bldg. 220 B124
Gaithersburg, MD 20899

Abstract

This paper describes a real-time hierarchical system that combines (fuses) data from vision and touch sensors to simplify and improve the operation of a coordinate measuring machine (CMM) used for dimensional inspection tasks. Our emphasis is on sensory processing techniques that can aid CMM applications rather than on the analysis of CMM performance measurements. Our system consists of sensory processing, world modeling, and task decomposition modules. It uses the strengths of each sensor-- the accuracy of the CMM scales and the analog touch probe and the global information provided by a low resolution camera--to simplify the inspection task while maintaining CMM accuracy. In the experiment described, the vision module performs all computations in image coordinate space. The system fuses data obtained from the vision system in image coordinates with the velocity and probe position provided by the CMM controller.The fused information provides feedback to the motion controller as it guides the probe during a raster scan.

We also describe a method for combining information from the vision system and the probe in real-time to simplify the data acquisition process required for camera calibration tasks. We autonomously register 2-D and 3-D points as the probe moves along a pre-programmed path. These corresponding points are used as input to a calibration algorithm

Keywords: calibration, computer vision, Coordinate Measuring Machine (CMM), Real-time Control System (RCS), sensor fusion, touch probe.

1. Introduction

A coordinate measuring machine (CMM) is a highly accurate multi degree-of-freedom robot often used for dimensional inspection of manufactured mechanical parts. Dimensional inspection involves measuring the relative geometry of surface features and determining whether they are within tolerance. Examples of geometries evaluated by such a system include shapes of smooth surfaces, distances between edges, positions of holes, and diameters and shapes of holes. Virtually all CMMs in use today use touch-trigger probes. While extremely accurate, the data acquisition rate of touch-trigger probes is very low, usually about one point per second. This type of probe is not suited for gathering dense surface data important for measurement of parts with complex geometries, or for locating part edges which contain important measurement information.

To overcome the deficiencies of current CMM technology, we are performing experiments to increase the speed and flexibility of CMM's data acquisition while maintaining accuracy, and to simplify the measurement process. As part of this effort, we are investigating the interaction of a video camera and an analog touch probe within a hierarchical control system for controlling probe motion. We have developed vision algorithms which enable a CMM to sense how it interacts with objects in the 3-D world.

Because of the low resolution of the camera, visual data are not accurate enough to use for precise part measurements. However, vision can provide position estimates of features of interest on the part. Using real-time vision, the probe can be guided to features of interest, and probe measurements can then provide the inspection data. Probe motion is controlled by using feedback from the vision system as it tracks the moving probe. This allows parts to be measured even if an accurate *a priori* model is not available.When CAD models are available, vision processing can be used to register the part with the model and thereby avoid the need for precise positioning of the part prior to inspection.

As the probe scans across a surface, the motion of the probe is controlled by information from three sensors: the camera, the machine scales, and the probe itself. Vision provides information about positions of part features (e. g. edges, holes, grooves, protuberances) as well as the image coordinates of the probe. The machine scales, when used in conjunction with vision, provide the distance of the probe from these features. The probe data provide the displacement of the probe from the part surface. By combining all sensor information, we are able to demonstrate the ability to scan part surfaces quickly and to use visually detected edge proximity as feedback to control arm motion.

In the following section, we discuss the strengths and weaknesses of camera sensors and touch probes. Section 3 describes the hierarchical control system architecture. Section 4 describes the hardware and software used in the testbed. Section 5 describes the integrated vision-probe surface scanning experiment, the vision algorithms used, and a brief discussion of other experiments being performed in the testbed. Our conclusions and future research are discussed in Section 6.

2. Vision and Touch Sensors

To use the combination of camera and touch probe to its best advantage in an inspection task, we compare the strengths and weaknesses of each sensor. In this discussion, we do not assume the use of high magnification lenses. A camera is a non-contact sensor and, therefore, camera measurements have no impact on part inspection accuracy. Although camera data are generally noisy, an image containing between 65,000 and 262,000 pixels can be read in 16 milliseconds (msec) Image processing algorithms can then locate and measure global features of interest in the scene such as object edges, corners, and centroids.

The problems associated with using camera data can be divided into two classes: geometric constraint problems and radiometric constraint problems [1]. Geometric constraints include visibility, field of view, depth of field, and pixel resolution. The radiometric constraints include illumination, specularity (glare), dynamic range of the sensor, and contrast. Section 4 briefly discusses our use of polarizing filters and polarized lighting to reduce specularity.

Touch trigger probes used in most CMM applications are contact sensors. They are highly accurate measuring sensors, and there is very little noise associated with their data [2]. However, the data they extract are of a local nature; they only apply to the specific points touched. Since information is read one point at a time, the touch probe is unsuitable for rapid high-density data acquisition. Touch probe systems are also crash prone if the part being inspected is not exactly in accordance with the CMM program.

To take advantage of the strengths of both the camera and the touch sensor and to overcome their individual shortcomings, our system is designed as an integrated vision touch probe system. Single sensor systems are limited in their ability to sense and identify meaningful features under varying conditions. The use of multiple sensors to perform a task overcomes the problems caused by relying on a single sensory input, but creates other problems concerning the interpretation and possible merging (fusion) of the outputs generated by these sensors. Our system combines the strength of a vision system, the ability to quickly generate global information, with the strength of a touch probe, the ability to obtain highly accurate measurement information. Before describing our application, we describe the system architecture.

3. Integrated System Architecture

The parts inspection testbed is designed according to the architecture guidelines of the Real-time Control System (RCS) described in [3]. The architecture defines a hierarchy of controller nodes, each with an assigned set of responsibilities that include sensory processing (SP), world modeling (WM) and behavior generation (BG)(Figure 1). Several locations at NIST and industry have adopted RCS for building robot controllers. It has been used on welding, deburring, measuring, and milling machine robots. The inspection testbed uses the RCS libraries to control the CMM, probe and vision systems. Each level operates on a different time and space scale. RCS is conducive to rapid prototyping, code-reuse and concurrent software development. The current system implements the four lowest levels of RCS control: *servo*, *prim*, *e-move* and *task*.

Figure 1 . RCS Controller Node

4. Testbed Environment

The CMM controlled by the system is a four degree-of-freedom Cordax [1](Figure 2). The rotational degree of freedom associated with the table has been disabled for our experiments. The CMM table moves in the Z direction, and the arm moves the probe in positive or negative XY directions. The controller positions the CMM table and the probe in a coordinate system relative to a pre-defined origin in order to position the probe.

We have experimented with several different probes in this system Currently, this complement of probes includes a simple 1D LVDT, a 3 DOF LVDT from API, a capacitive sensor from ExtrudeHone and

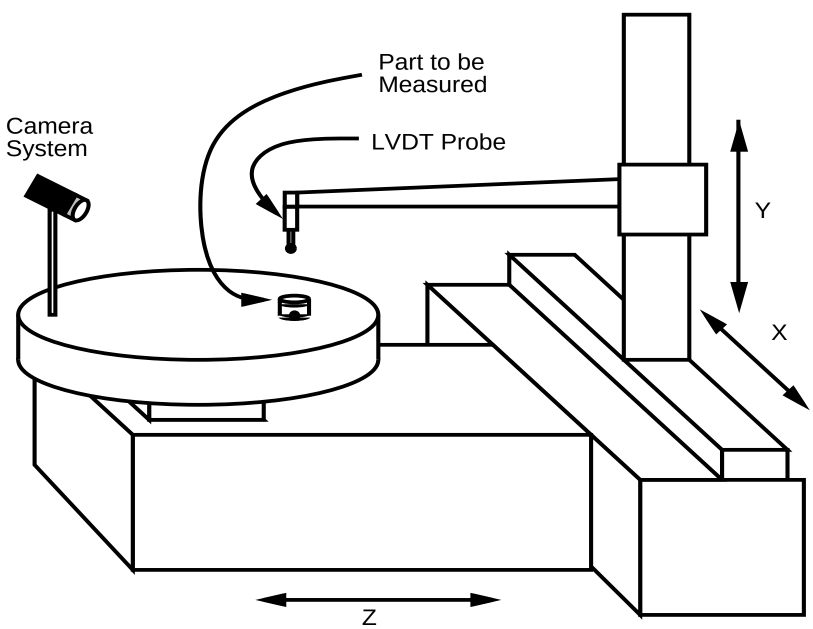

Figure 2. Testbed: CMM Probe and Camera

1. Certain commercial equipment, instruments, or materials are identified in this paper in order to adequately specify the experimental procedure. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the materials or equipment identified are necessarily best for the purpose.

a laser triangulation probe from Sensor Adaptive Machines, Inc. (SAMI). A kinematic base has been attached to the base of each probe to facilitate the connection of each probe to the pan-tilt wrist. The vision system consists of a miniature cylindrical ("lipstick") black and white CCD camera with a 12 mm lens. The camera is mounted on the CMM table, but it is stationary relative to the part on the table and generates both SVideo and NTSC output. The SVideo output is fed into a Sun Video digitizer mounted in a Sun Ultrasparc workstation running Solaris 2.5. Images are digitized to 8-bit grey-scale at a resolution of 320x240 pixels.

The software is constructed on top of a heterogeneous operating system environment. It is designed according to the guidelines described in [4]. Currently, the CMM is controlled under VxWorks, the probes are controlled either under VxWorks or Microsoft Windows NT, the vision system is controlled under Solaris, and the user interfaces are controlled under Java Virtual Machine. In most typical development environments, this number of operating systems would make the cost of maintaining communications libraries prohibitively expensive. The system uses the RCS Neutral Manufacturing Language (NML) and the Communication Management System (CMS) to overcome this problem. NML and CMS allow many different processes running on different computers to communicate with one another. CMS supports single read/write channels and read/write channels with multiple readers and writers. In addition, it supports queueing messages, backplane, shared memory, and internet communication. The NML libraries are available free from NIST and are compilable under VxWorks, Solaris, Lynx OS, and the many forms of Microsoft Windows. [5]

Lighting conditions and specularity create image processing problems for this application since many of the parts being measured have highly reflective machined surfaces. The glare and reflections from overhead lighting introduce shadows and artifacts that interfere with the image processing algorithms. Efforts to reduce specularity in this environment must be practical as well as effective. We have introduced a relatively simple and inexpensive technique, polarization of light, to reduce specularity. Sheets of polarizing filters attached to the fluorescent light fixtures in the laboratory serve as polarizers at the light source. In addition, we have a rotatable polarizing filter attached to the camera lens that is adjusted to select light polarized at 90 degrees from the overhead filters. The use of polarizing filters does not eliminate the problems of specularity, but it greatly reduces many of these effects [6][7].

In addition to using filters to reduce specularity, we have designed and installed a diffuse illuminator. The illuminator is suspended over the CMM work area and consists of a large shield and an independent light source. The shield blocks scattered light from the overhead light sources but allows auxiliary lighting to be introduced to the work area.

5. Vision-Probe Experiments and Algorithms

The experiment is a visually servoed raster scan of a piston surface. We assume only that we know the initial probe position and that the start position and goal edge are in the camera's field of view. The part can be placed anywhere in the camera's field of view. Visual tracking is used to guide the probe to the piston edge. We fuse information generated by the vision system with data generated by the machine scales and the probe. The fused information is used to guide the movement of the touch probe as it performs a raster scan inspection across the surface of a piston part. Vision provides information about positions of part features of interest (e.g. edges,) recognizes the probe, and tracks its position as it inspects the part. All vision data are represented in 2-D image coordinates and the camera is not calibrated. The probe data provide displacement from the part surface, and, in conjunction with the machine scales, the probe position. Probe data are expressed in a 3-D coordinate system relative to a pre-defined origin relative to the CMM table. When the fused visual-probe information determines that the probe is within a pre-defined distance from the edge, the controller is preprogrammed to move a fixed offset in the *z* direction from that position and resume scanning in the opposite direction (Figure 3). Visual tracking is terminated after a pre-programmed number of scans is completed.

Visually derived feedback for the system consists of the 3-D distance between the current position of the probe and the goal edge. The vision system performs the following processing steps to provide this information.

(1) Segmentation of the part surface to be scanned from the total scene.
(2) Extraction of pixels representing edge or perimeter points.
(3) Fitting edge pixels to lines or curves.
(4) Defining the initial probe position in image coordinates via a human-machine interface.
(5) Tracking the probe as it moves along the part's surface.

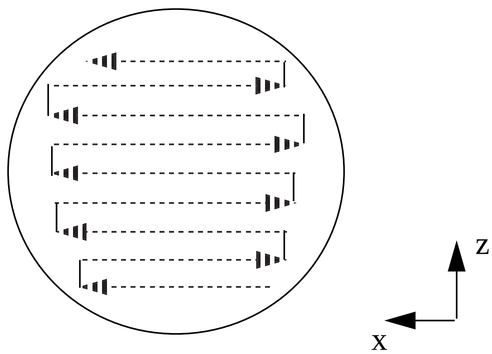

Figure 3. Path of Raster Scan

The following sections briefly describe each step in the vision processing. See [8] for further detail.

5.1. Segmentation

The first step in vision processing requires that the part being inspected be separated or segmented from other things in the field of view. Segmentation algorithms are scene dependent: there is no single segmentation algorithm that is always successful. Our initial experiments used block parts; subsequent experiments used a piston.

We segmented the block from the workplace scene by using a simple thresholding algorithm . When working with the piston, a connected component algorithm was used to segment the piston from the background after the thresholding process. Using *a priori* knowledge about the part, we are able to identify the region that represents the part surface. (Reference [9] contains descriptions of thresholding operations, edge extraction algorithms, and the connected component algorithm.)

5.2. Extraction of Edge Pixels

A number of edge detection algorithms can be used to extract those pixels that represent edges. An edge is defined as the transition between a dark and a light region, or, conversely, a light and a dark region. When a part is cleanly segmented from any background information, an edge detection algorithm can easily label all points representing its boundary points. When using the connected component algorithm for segmentation, a boundary tracing algorithm [10] is used to extract the edge pixels representing the boundary points on the region of interest.

5.3. Line or Curve Fitting

A Hough transform (described in [11]) is applied to the extracted edge points of a step-block to fit the points into straight lines. The lines computed by the Hough transform represent the edges or boundaries of the block.

To fit the edge pixels extracted from the circular piston surface to a curve (Figure 4), we use a least squares fitting algorithm. In the image plane, the projected image of the piston appears elliptical in shape, and therefore the points are fit to an ellipse rather than to a circle. The resultant symbolic representation of the feature boundary is used to determine the intersection point of the probe trajectory and the part boundary.

Figure 4. Piston Surface

5.4. Initial Probe Position

In determining the initial position of the probe, we assume that the controller always places the probe in a pre-defined 3-D position somewhere on the part's surface. The part is not fixtured in this experiment; the fixed 3-D position can lie anywhere on the part surface. Using a man-machine interface, the system operator determines the approximate location of the probe in 2-D image coordinates by examining the image. We compute a vector in the image plane in the direction of motion using knowledge of the probe position and motion. The intersection in the image plane of the motion vector and the nearest part edge represents an initial estimate of the probe's 2-D goal point. The distance between the initial position and the goal position is measured in pixels in the image plane.

5.5. Tracking

We track the probe in 2-D as the arm moves towards its goal position using the predicted probe image velocity and sum of absolute differences (SAD) correlation algorithm [9]. A predictive filter is used to filter and predict the probe position and velocity at the next time interval [12][13][14]. The prediction is based on a weighted sum of the current position and velocity and a history of past positions and velocities. Depending on the weights used, the predictions can be tuned to be more responsive to new readings or to respond smoothly over time. At each processing iteration, the search direction and window used for the SAD correlation are recomputed based on the predicted probe image position and velocity. The size of the search window is determined by the probe velocity magnitude, and the direction of search is biased in the direction of velocity. The probe position is computed to be the position which yields a minimum value for the sum of absolute differences over the search space. The correlation template is updated each cycle to reflect the current grey scale information representing the probe position.

5.6. Discussion

Visual feedback to the system consists of updates of the 2-D distance between the probe and the nearest edge. Both the current position and the intersection point on the edge are updated each processing cycle. The world model fuses the visual information with the 3-D position and velocity information supplied by the CMM in order to predict the 3-D distance remaining between the probe and the part's edge.

We have demonstrated that we can successfully perform multiple scans of the piston surface using visual feedback with a non-calibrated system. The path of the probe is not preprogrammed. Our only requirement is that the probe start position be on the piston surface and that the entire surface be visible in the camera field of view. During the raster scan, data are collected for analysis after a run. This data includes the probe measurements, controller variables such as position, velocity, and acceleration, vision data, and fused data. We have shown that it is possible to gather dense probe data quickly while performing multiple scans. (As mentioned earlier, it was not in the scope of this experiment to analyze the probe data collected during the scans; our primary objective was to demonstrate the interaction of a vision sensor and a CMM.) We varied the speed of the CMM arm as it performed the scan and achieved a maximum velocity of 30 mm/sec. At velocities greater than 30 mm/sec, the probe overshot the piston edge because there was insufficient time for the controller to decelerate after the vision system detected the part boundary. Figure 5 is a graph of the logged 2-D distances to the piston edge as the scan is performed. The x axis represents time and the y axis is the distance in image coordinates (pixels) to the edge. The distances measured vary for two reasons: (1) the length of the scan varies as the probe measures different widths across the surface, and (2) the measured distance is a function of the time required for the controller to halt.

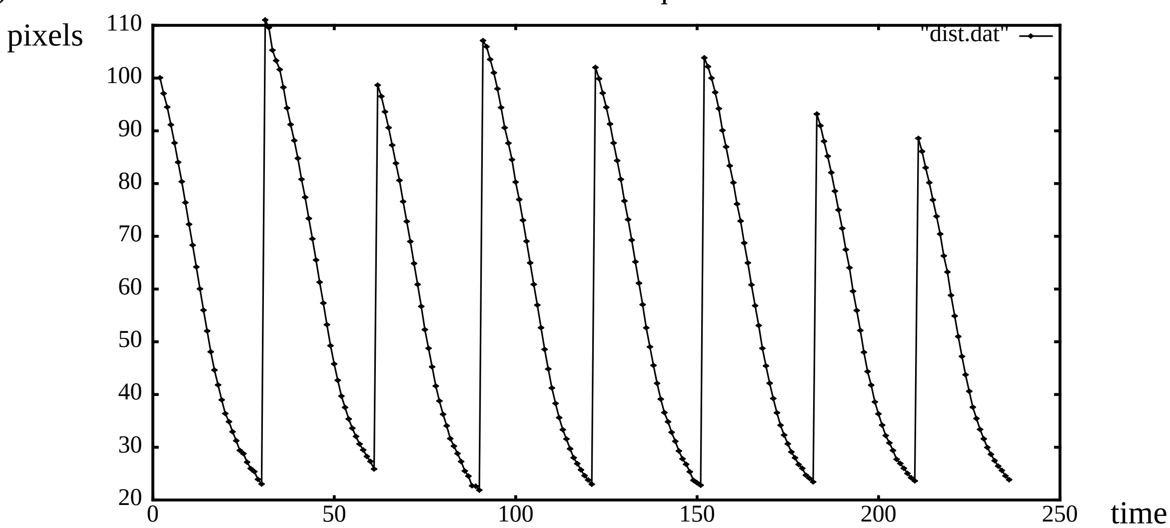

Figure 5. 2-D Plot of Distances to Boundary during Multiple Scans

Figure 6 is a graph of the probe measurements read during a typical scan of the surface. The x axis represents time in seconds; the y axis is the probe measurement in millimeters. 174 probe readings were recorded in 5.2 seconds.

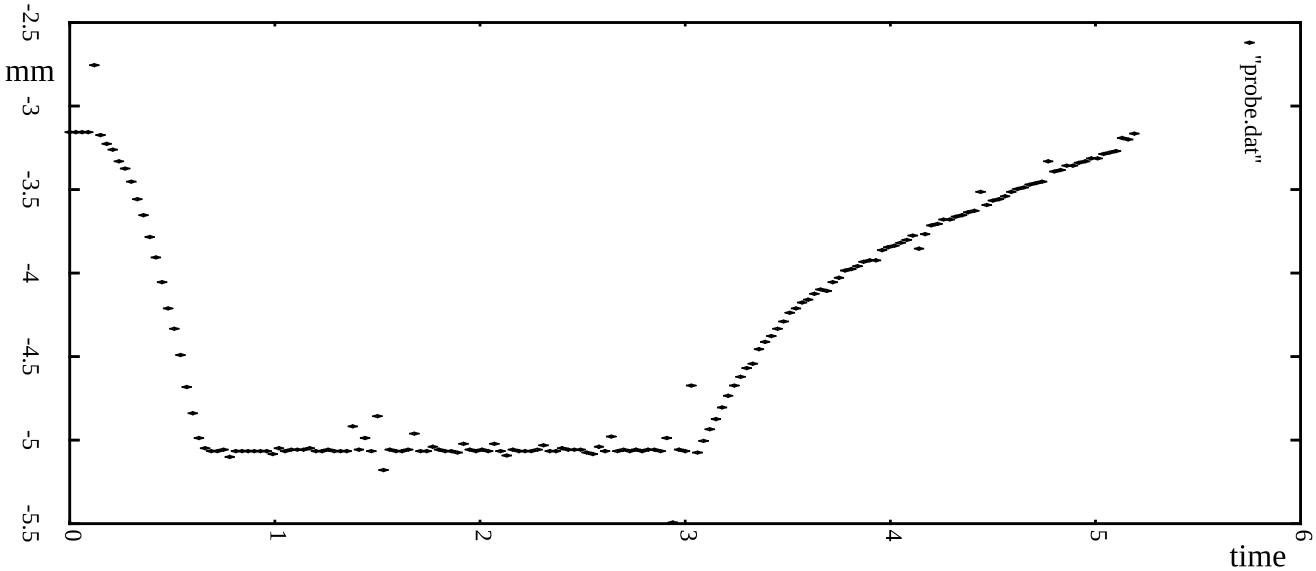

Figure 6. Probe Measurements Read During Single Scan

5.7. Autonomous Data Collection for Calibration

We are also investigating fast, easy to perform, methods for registering image space with CMM space in order to calibrate our camera. Calibration algorithms compute the position and orientation of a a camera in the 3-D world as well as the camera's optical characteristics. Once the camera calibration is known, the position of any object visible in those cameras may be back projected into the 3-D world to determine its 3-D position and orientation. Part locations and the location of the CMM are specified in 3-D. Distances to objects are determined using Cartesian calculations. Under calibrated vision, all manipulation problems requiring vision are transformed from the 2-D domain into 3-D Cartesian-based manipulation problems. This gives us the ability to visually servo the arm to features of interest and to inspect and follow linear and curved contours.

The camera calibration problem is known to be very difficult. In addition to algorithmic complexities, the computed calibration parameters are very sensitive to changes in camera orientation and movement. Because of this, frequent re-calibrations are needed to insure the integrity of the parameters. Collecting sets of corresponding 2-D and 3-D data points, the input required for calibration, is a very error prone and time-consuming operation. To facilitate this operation, we have developed an autonomous data collection system for registering sets of points in the image plane with their corresponding points in world coordinates. The calibration itself is performed using an algorithm designed by Castaño [18]. The automated data acquisition procedure provides an example of the complementary integration of sensor outputs. The information provided by the probe and scales is completely independent of the information provided by the camera, but the union of the outputs provides a representation for understanding the relationship between 2-D and 3-D coordinates.

To perform the data acquisition, the probe is mounted on the CMM and the camera is mounted in a location where it can observe the workspace of the CMM. The visible workspace of the vision/robot/probe system is initially estimated by hand. The CMM is moved under joystick control to many different positions in the workspace to measure the extent of the camera's field of view. The probe is moved closer to the camera and then moved so the projection of the probe lands in the top-left, top-right, bottom-left and bottom-right corners in camera space. This defines a "front plane" to the workspace. We then move the CMM away from the camera and perform the same operation. This operation defines a "rear plane." This frustum-like space is then populated with three dimensional points.

Next, a red light-emitting diode (LED) is attached to the probe tip. LEDs are inexpensive and are easily tracked in images. All external lights are dimmed to simplify the segmentation of the LED from the background. We use simple area statistics to find the LED's center in image coordinates. We assume the illumination profile for the LED is uniform in all directions. The LED's position in future frames is estimated using a filtering/prediction algorithm described in [19].

The CMM is commanded to move the probe to each predetermined 3-D point within the frustum-like space and then stop. The vision system tracks the LED during the entire movement. When the probe stops, the vision system waits for the image to settle and then it records the (x, y, z) position of the probe tip along with its (u, v) image position. The two sets of values are concatenated together to form (x_i, y_i, z_i, u_i, v_i) quintuplets. A list of these quintuplet sets of data is used as input to the calibration algorithm. Figure 7 is the projection of the original 3-D points representing the probe position back into the image plane. The back projection uses the calculated calibration matrix. The results are overlaid on the original LED data points. The coordinates are measured in pixels. The results of the calibration are accurate to within 2 pixels for this data set. The time required for data collection and calibration is approximately 20 minutes.

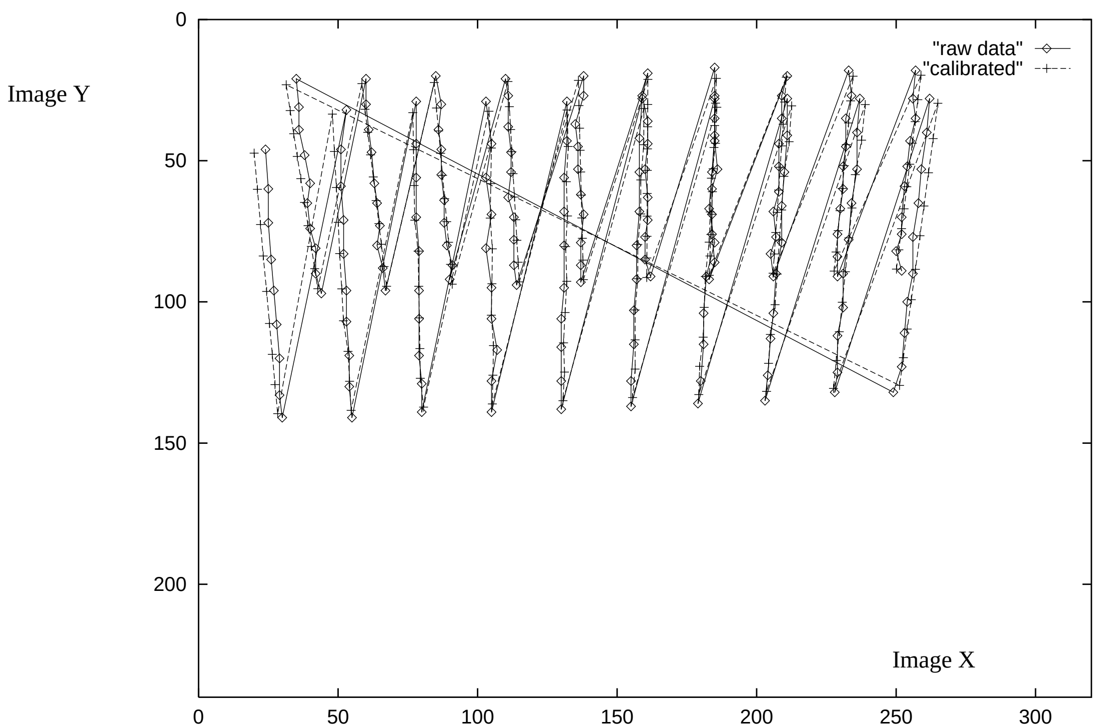

Figure 7. Projection of original 3D points onto the image plane using the calculated calibration matrix. Results are overlaid on original LED data points. The coordinates are in pixels.

6. Conclusion and Future Research

Our system demonstrates that vision can be used to guide the acquisition of CMM measurements. The first experiment described does not use a calibrated camera . We have demonstrated that an uncalibrated vision system adds flexibility to a measurement task by eliminating the need for exact placement of the part being inspected. It also reduces the need for preprogramming all CMM motions when visual data are used as feedback to the CMM controller.

For applications that require calibration of image space with 3-D space, we have developed a fast technique for registering sets of data points that are used as input to a calibration algorithm. The unique feature of this technique is the autonomous collection of registered 2-D and 3-D points.

In order to overcome the problems caused by geometric and radiometric constraints (see Section 2), we plan to introduce multiple cameras with gaze control mechanisms into the testbed. This will give us the ability to analyze parts and features from multiple viewpoints as well as giving us the ability to adjust the field of view of an individual camera. We are continuing to develop new sensor-servoed scanning techniques and to experiment with strategies for using vision systems to support CMM tasks. Our ultimate goal is to demonstrate that the integration of vision and touch probes is a more effective tool than either sensor by itself, and to transfer new techniques using multiple sensors to CMM users.

References

[1] K. Tarabanis, R. Y. Tsai, "Sensor Planning for Robotic Vision: A Review", Robotics Review II, Ed. O. Khatib, J. Craig, Lozano-Perez, MIT Press, 1992, pp. 113-136.

[2] P. K. Allen, Robotic Object Recognition Using Vision and Touch, Kluwer Academic Publishing, Boston, MA. , 1987.

[3] J. Albus, A. M. Meystel, "A Reference Model Architecture for Design and Implementation of Intelligent Control in Large and Complex Systems", International Journal of Intelligent Control and Systems, Vol. 1, No. 1, pp. 15-30, 1996.

[4] J. Fiala, "Note on NASREM Implementation", NIST Internal Report 89-4215, Gaithersburg, MD. 1989.

[5] F. Proctor. , "Intelligent Controller Systems Development", http://isd. cme. nist. gov/brochure/SoftwareSystems. html, ISD Electronic Brochure, NIST, 1997.

[6] R. Daum, K. Harding, "The Machine Vision Lighting Testbed", Applied Machine Vision Conference, June 1992.

[7] L. B. Wolff, "Using Polarization To Separate Reflection Components", CVPR, 1989, pp 363-369.

[8] M. Nashman, W. Rippey, T. Hong, M. Herman, "An Integrated Vision Touch-Probe System for Dimensional Inspection Tasks", NISTIR 5678, June 1995.

[9] A. Rosenfeld and A. Kak, Digital Picture Processing, Vol. 1, Second Edition, Academic Press, 1982.

[10] M. Sonka, V. Hlavac, R. Boyle, Image Processing, Analysis and Machine Vision, pp 129 - 134, Chapman & Hall, London, 1993.

[11] R. Haralick, L. G. Shapiro, Computer and Robot Vision, Addison-Wesley Publishing Co. , 1992.

[12] Coombs, D. , Herman, M. , Hong, T. , Nashman, M. , “Real-Time Obstacle Avoidance Using Central Flow Divergence and Peripheral Flow”, Proc. Fifth International Conference on Computer Vision, Cambridge, MA, June 1995.

[13] H. Schneiderman, M. Nashman, “A Discriminating Feature Tracker for Vision Based Autonomous Driving”, IEEE Transactions on Robotics and Automation, Vol. 10, 1994.

[14] A. J. Wavering, R. Lumia, “Predictive Visual Tracking”, Proceedings of SPIE Intelligent Robots and Computer Vision 7, Boston, MA,1993.

[15] J. Krejci , “Application Software”, in Coordinate Measuring Machines and Systems, ed. Bosch, J. , Marcel Dekker, Inc. , 270 Madison Avenue, NYC, NY, 10016, pp. 103-122, 1995

[16] S. Sarkar, K. L. Boyer, “Perceptual Organization in Computer Vision: a Review and a Proposal for a Classification Structure”, IEEE Transactions on Systems, Man and Cybernetics, Vol. 23, No. 2, March/April, 1993.

[17] M. Hu, “Visual Pattern recognition By Moment Invariants”, IRE Transactions on Information Theory, 1961.

[18] A. Castaño. , S. Hutchinson. , “Visual Compliance: Task-Directed Visual Servo”, IEEE Transactions on Robotics and Automation, Vol. 10, No. 3, pp. 334-342, June 1994.

[19] M. Nashman. , T. Hong. , W. Rippey. , M. Herman. , “An Integrated Vision Touch-Probe System for Dimensional Inspection Tasks”, Proceedings of the SME Applied Machine Vision ‘96 Conference, Cincinnati, OH, June 3-6, 1996.

[20] B. Yoshimi, T. Hong, M. Herman, M. Nashman, W. Rippey, “Integrated Vision and Touch Sensing for CMMs”, SME Applied Machine Vision Conference, Charlotte, N. C. , June, 1997.