# A Feedback Design to a CAD-Guided Area Sensor Planning System for Automated 3D Shape Inspection

Quan Shi[1], Ning Xi[2] and Chris Spagnuluo[3]

[1]Michgain State University, shiquan@egr.msu.edu
[2]Michgain State University, xin@egr.msu.edu
[3]Michgain State University, Spagnu23@msu.edu

**ABSTRACT**

A feedback-based area sensor planning system is introduced in this paper. This system is developed for automated 3D shape inspection. A CAD-guided area sensor planning system had been developed previously. However, physical properties of a part surface are usually difficult be modeled for an area sensor. Shadows, reflections, and image quality are three problems related to surface measurement quality, which cannot be predicted from a CAD model. Feedback from a measured 3D point cloud can be used for dealing with those problems by adding more viewpoints in iterations. In feedback mode, the inspection system generates viewpoints according to point clouds measured in each step. The loop stops when the obtained point clouds satisfy the measurement quality requirements. In this paper, a general framework of such area sensor planning system is introduced. The developed feedback system has been implemented for a robot-aided dimensional inspection system.

**Keywords:** Sensor planning, 3D shape measurement, quality control, feedback.

## 1. INTRODUCTION

An active, triangulation-based area sensor has been widely developed for rapid three-dimensional (3D) shape reconstruction and inspection. Sometime, the scanning process requires moving either the sensor or the part itself to obtain the entire surface from all directions. For surface inspection of a large-scale automotive part, a robot-aided area sensor planning system is often required to move a range sensor without human-involved path programming. The sensor planning problem has been approached with various techniques, e.g., expert system, generate and test, synthesis, fuzzy set theory, and probabilistic methods. A review of planning a range sensor for surface inspection can be found in [1]. Many commercial equipment and systems were developed as well for this task [2]. Among those techniques, a CAD-guided area sensor planning method is more suitable because the CAD model of a part provides the analytical geometry information such that the viewing pose of a range sensor can be accurately determined. And measurement accuracy not only depends on sensor itself but also is affected by sensor's view poses [3].

A CAD-guided robot sensor planning system is illustrated in Fig 1. Given measurement constraints of a range sensor and the CAD model of a part, a robot sensor planner can be developed to find a suitable robot path such that 3D shape of the part surface can be quickly reconstructed [4].

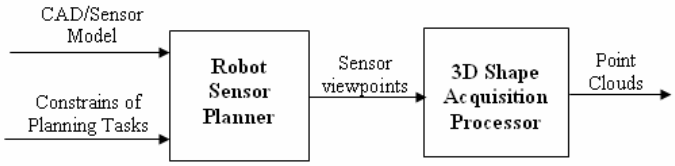



Fig. 1: A CAD-guided robot area sensor planning system for automated 3D shape verification.

However, for such a model-based view planning system, two problems are still remaining for vision based surface inspection: 1) shadow area, caused by self-occlusion of the part, 2) glossy spot, caused by surface specular reflection. Both shadow and reflection can generate "holes" in the measured point cloud, which must be filled for surface inspection. Since estimation of shadow and light reflection is usually not available from the preliminary knowledge, it is

unavoidable to have "holes" in measured point clouds. Therefore, the measured information needs to be fed back for calculating new viewpoints.
Fig 2 shows a feedback-based area sensor planning system. A Model-based area sensor planner is part of this close-loop system that only calculates an initial viewpoint set, on which the measured point clouds will be used for new viewpoints generation. The set of viewpoints iteratively updated till all holes can be filled. Meanwhile, the points in overlap regions between point clouds can also be used to improve the measurement accuracy.

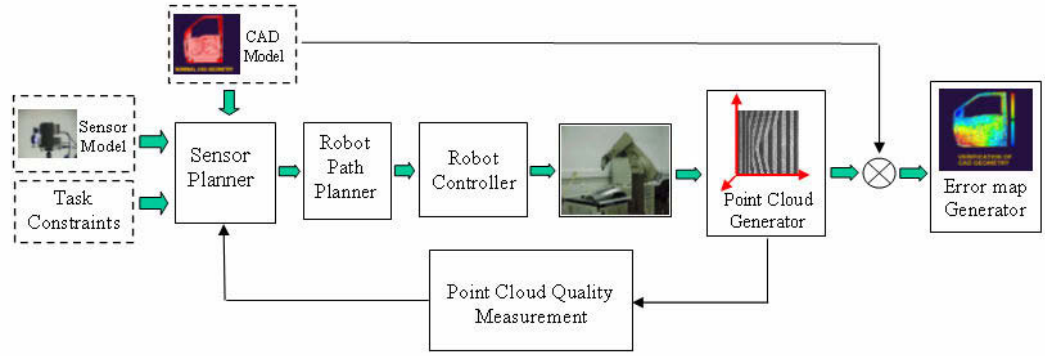



Fig. 2: A feedback-based area sensor planning system.

To plan a range sensor through real time observations, a positional space (PS) algorithm [5] is introduced, which solves the Next Best View (NBV) problem without any priori CAD information. This PS algorithm mainly focuses on how to estimate viewpoints such that all part surfaces will be visible for measurement. A ray-casting method [6] is developed to incrementally build a meshed model of a part with unknown shape. In [7], a method of clustering unseen directions is introduced, in which the "completeness" (percentage of surface sampled) of scanned surface is optimized through multiple viewpoints. The planning methods to solve self-occlusion problem have been discussed, but methods to deal with light reflection are still not well developed.

Many sensor planning systems focus on minimizing the number of viewpoints and the length of paths, the method introduced in this paper focuses on planning a range sensor such that the measurement quality can be improved. The model-based area sensor planning scheme and the feedback strategy are integrated together to solve the problem. This feedback-based area sensor planning system can: 1) initialize a set of viewpoints using a CAD-guided sensor planner to cover majority of part surface, 2) automatically generate viewpoints according to the measured point cloud and its corresponding CAD model. This paper is organized as follows: Section 2 introduces the system mathematic model; Section 3 shows the implementation of the feedback system. Conclusions are summarized in Section 4.

## 2. MATHEMATIC MODEL OF THE PLANNING SYSTEM

The developed feedback system utilize a variety of techniques to automatically acquire a qualified 3D point cloud: CAD-guided area sensor planning, image processing, 3D points filtering and clustering, point cloud registration, structured light method, and pixel-to-pixel calibration strategy [8]. Fig 3 shows a block diagram of the proposed feedback system, which integrates all methods together to automatically finish the inspection task.

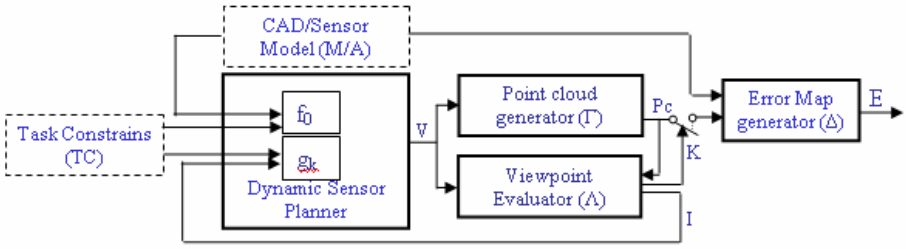



Fig. 3: Diagram of the feedback based area sensor planning system.

The dynamic sensor planner in the diagram has four inputs: CAD model, area sensor model, task constrains, and feedback information. Viewpoints are output of the planner. A part's CAD model is tessellated into triangles and the coordinates of triangle vertexes are input to the system. The norm of each triangle is determined by the order of its vertexes: P1->P2->P3. A vertex adjusting process is developed so that all triangle norms are pointed out of part surfaces. Finally, the triangulated CAD model can be represented by Eqn. (2.1):

$$M = \{T_i \mid T_i = [(X_{i1}, Y_{i1}, Z_{i1}), (X_{i2}, Y_{i2}, Z_{i2}), (X_{i3}, Y_{i3}, Z_3)], i = 1 \ldots n\} \tag{2.1}$$

Task constraint TC represents a set of measurement conditions required for the 3D shape measurement using the developed area sensor:

$$TC = \{fov, S, \rho, \eta, f_d\} \tag{2.2}$$

where *fov* represents the field of view, which is defined by the length L and width W of a rectangle field, S is the standoff distance of a range sensor, and ρ represents the image resolution. The symbol η represents visibility of the area sensor, determined by three vectors: projection vectors PV, camera viewing vector CV, and surface norm vector SV. For a piece of surface with average norm vector SV, it is visible if Eqn. (2.3) is satisfied:

$$\begin{cases} \arccos\left(\dfrac{PV \cdot SV}{\|PV \cdot SV\|}\right) < \theta_{th1} \\ \arccos\left(\dfrac{CV \cdot SV}{\|CV \cdot SV\|}\right) < \theta_{th2} \end{cases} \tag{2.3}$$

where $\theta_{th1}$ ensures that the encoded patterns can be projected on the surface and $\theta_{th2}$ ensures that this piece of surface can be seen by camera. The $f_d$ in Eqn. (2.2) represents the focus distances that contains two values: nearest focus distance $D_1$, and farthest focus distance $D_2$ [12].

$$\begin{aligned} D_1 &= \frac{a \cdot f \cdot d + c \cdot f \cdot p}{a \cdot (d - f) - c \cdot f} \\ D_2 &= \frac{a \cdot f \cdot d - c \cdot f \cdot p}{a \cdot (d - f) + c \cdot f} \end{aligned} \tag{2.4}$$

The area sensor model can be described by Eqn. (2.5), in which d represents the sensor baseline distance, S represents the sensor standoff distance, L is the measured distance of a pair of correspondent points on reference plane [9], calibration of this sensor model can be found in [8].

$$h = \frac{L \cdot S}{d + L \cdot S} \tag{2.5}$$

Defect map is a group of 3D points that are reported from the viewpoint evaluator. There are three types of defects: a shadow map IS, a reflection map IR, and an inaccuracy map IA. Shadow map is a set of 3D points extracted around the boundary of a shadow region among a point cloud.

$$IS = \{p_i(X_{is}, Y_{is}, Z_{is}) \mid i = 1,2,\ldots k_s\} \tag{2.6}$$

Similarly, a reflection map is a set of points extracted around the boundary of a glossy spot. Measurement in those bright spots is not available because the projected stripes cannot be seen by the camera as well.

$$IR = \{p_i(X_{ir}, Y_{ir}, Z_{ir}) \mid i = 1,2,\ldots k_r\} \tag{2.7}$$

Symbol V in Fig. 3 represents the set of estimated sensor viewpoints, which has two categories: $V_0$ is used to represent the initial set of viewpoints generated from CAD model, and $V_k$ is used to represent the sets of viewpoints estimated from feedback, while k is the iteration times. Each viewpoint includes a location *p* and a viewing vector *v* as seen in Eqn. (2.8)

$$\begin{aligned} V &= (\bigcup_{k=1}^{n} V_K) \bigcup V_0 \\ V_0 &= \{\Psi_{0i} = (p_i, v_i), p_i \in R^3, v_i \in R^3\} \\ V_k &= \{\Psi_{ki} = (p_{ki}, v_{ki}), p_{ki} \in R^3, v_{ki} \in R^3\} \end{aligned} \tag{2.8}$$

Initial viewpoint configuration is a process to estimate viewpoints based on the given CAD model of a part. As shown in Eqn. (2.9), $f_0$ represents is a function that represents a bounding box algorithm developed to find viewpoints set $V_0$ from a given CAD model M.

$$f_0 : M \mapsto V_0 \tag{2.9}$$

During the process, the tessellated triangle model is first divided into different patches according to the norms of triangles. For each patch, a bounding box is generated for holding all triangles. This bounding box is then evenly

divided into several smaller bounding boxes to hold certain triangles according to the measurement task constraints: *fov*, and $f_d$. Each small bounding box has three norm vectors defined: up, front, and right. An up vector points to the average norm of all triangles in one patch. The direction of a front vector is determined by the direction of the sensor's motion platform, usually an industrial robot. The right vector can then be calculated by the right-hand rule after up and front vector are decided. A viewpoint can then be estimated on the line of the up vector of a bounding box. Fig. 4 illustrates such a bounding box that is going to be measured under one viewpoint.

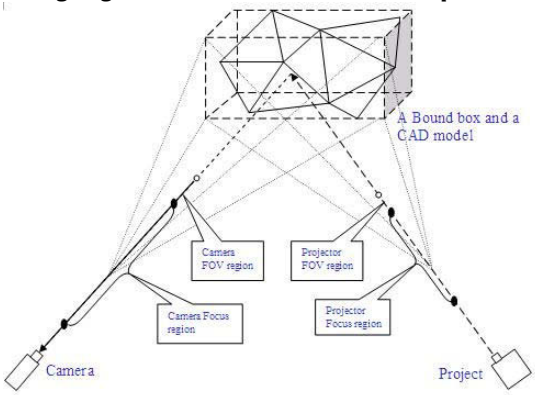



Fig. 4: A bounding box method for CAD-guided area sensor planning.

Function $g_k$ in Fig.3 is the feedback controller that estimating viewpoint from the defect map I. $V_k$ in Eqn. (2.8) is a set of viewpoints that is estimated from a defect map I, since the planning process is recursively executed, symbol *k* is used to represent the number of iterations, as shown in Eqn. (2.10),

$$g_k : I \mapsto V_k \tag{2.10}$$

Fig 5(a) illustrates a modified bounding box algorithm. Different with $f_0$, $g_k$ is developed to generate viewpoints for filling "holes". Fig 5(b) shows an example of shadows under a view. Viewpoint can be estimated by clustering 3D points around the unseen regions.

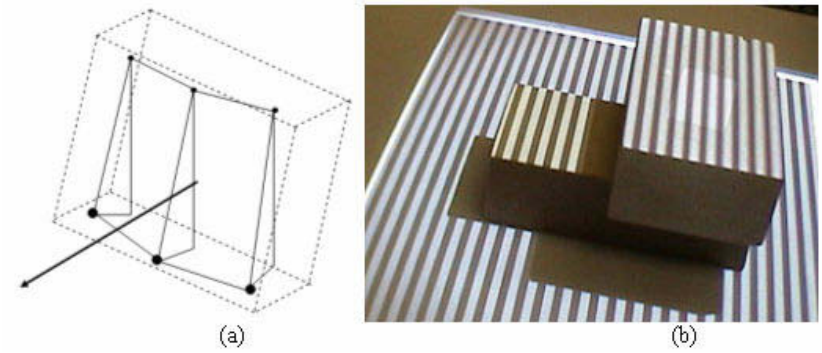



Fig. 5: Estimate viewpoint from a shadow map, (a) Estimate a viewpoint using a bounding box algorithm (b) Shadows in measurement.

For 3D measurement using a white light area sensor, light reflection often generates glossy spots in images so that the projected stripes on part surface are not detectable. Reflection problem is especially sensitive on surfaces of an automotive body part that is made by sheet metal, as can be seen in Fig 6(b). Powder spray can effectively avoid the reflection problem, but it is sometime not feasible for industrial applications. It may need to solve reflection problem without powder spray for automated 3D shape inspection. The Phong model [10] is used to analyze the reflection property: three parts are included in a total reflection $I_r$: uniform diffuse reflection $I_a$, direction diffuse reflection $I_d$, and specular reflection $I_S$, shown in Eqn. (2.11). Fig 6(a) illustrates the concepts of each part.

$$I_r = K_a \cdot I_a + \frac{1}{d^2} \cdot (K_d \cdot (\vec{V} \cdot \vec{N})) \cdot I_d + K_s \cdot I_s \cdot \cos^n(\vec{V} \cdot \vec{R})) \tag{2.11}$$

Knowing the angle between vectors $\vec{V}$ and $\vec{R}$ [10], new viewpoints for covering glossy spots then can be derived based on the Fresnel Law.

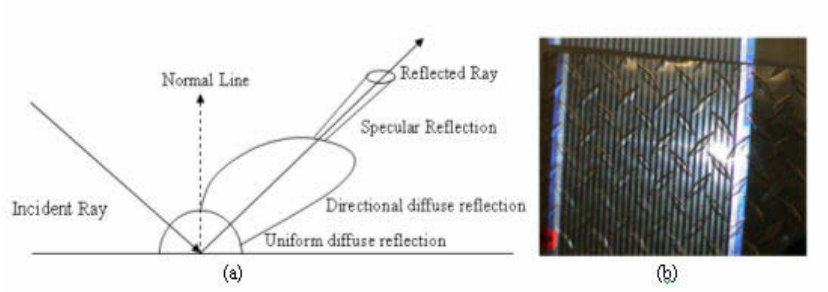



Fig. 6: Reflection analysis (a) Phong reflection model (b) A glossy spot caused by specular reflection.

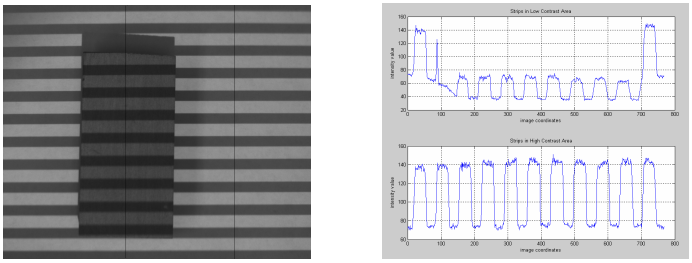



Fig. 7: Analysis of image intensity contrast on part surface (a) intensity profile (b) intensity image.

Color of part surface will also bring unpredictable effect in vision-based 3D shape measurement. Often, measurement uncertainty will be increased in low contrast images. Considering intensity noises, stripes in a high contrast area is usually more robust than stripes in a low contrast area [9]. Figure 7(a) illustrates an image with high/low contrast regions. The intensity profiles of two lines in Figure 7(a) are displayed in Figure 7(b): the top curve shows the intensity profile of the line through a low contrast area and the bottom curve shows the intensity profile of another line through a high contrast area. Image noises can be seen at the peak and valley of each wave, which affects the accuracy of the edge detection algorithm.

Eqn. (2.12) gives the model of the point cloud generator Γ, which represents a process to obtain point clouds from viewpoints. *Pc* represents a point cloud, which describes the shape of part's free-form surfaces. Because it contains millions of 3D points, an adaptive algorithm is developed to reduce the point density according to the geometry shape of the surfaces. Eqn. (2.13) shows the mathematic representation of a point cloud Pc.

$$\Gamma : V \mapsto Pc \tag{2.12}$$

$$\mathrm{Pc} = \{\mathrm{p_i}(\mathrm{x_i}, \mathrm{y_i}, \mathrm{z_i}) \,|\, \mathrm{i} = 1,2 \ldots \mathrm{m}\} \tag{2.13}$$

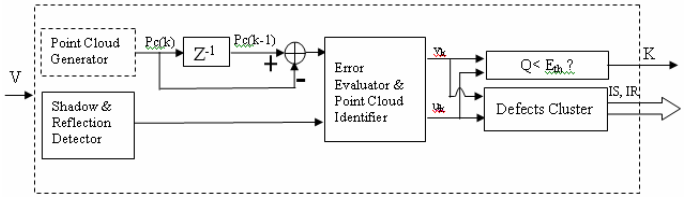



Fig. 8: Diagram of the viewpoint evaluator Λ.

Viewpoint evaluator Λ is a function that makes a judgment whether the measured point cloud Pc satisfies the predetermined conditions. If not, the defect map I will be fed back to update the viewpoint set V. Eqn. (2.14) shows this function and Fig 8 illustrates the structure of the viewpoint evaluator.

$$\Lambda : V \mapsto I \tag{2. 14}$$

Given two viewpoints from set V, two point clouds can be measured sequentially. Differences between those two point clouds are input to the error evaluator. Meanwhile, an image processor is designed to detect shadows and reflections. 3D points at the boundaries of shadows and reflection spots are collected using a point cloud identifier. $u_k$ and $v_k$ are parameters for quantizing the measurement errors. Q is a cost function of $u_k$ and $v_k$, which is used for system stability analysis and will be described next. A logic switch signal K is one output signal of $\Lambda$. If Q is less than a threshold, the switch will be closed so that the iteration process stops and the obtained point cloud will be output for errormap generation.

Eqn. (2.15) is the model of the error map generator $\Delta$. An error map E is the output of this whole feedback system which is a color-coded picture showing the distribution of the shape differences between the CAD model and the real part surface. An error map includes a set of 3D points and their distances D to the closest triangle, shown in Eqn. (2.16)

$$\Delta : (M, Pc) \mapsto E \tag{2.15}$$

$$\mathrm{E} = \{[\mathrm{p_i}, \mathrm{D}(\mathrm{p_i}, \mathrm{M})] \,|\, \mathrm{i} = 1,2...\mathrm{m}\}, \text{ where } \mathrm{D}(\mathrm{p_i}, \mathrm{M}) = \min(\mathrm{d}(\mathrm{p_i}, \mathrm{T_j}), \mathrm{i} = 1,2...\mathrm{m}, \mathrm{j} = 1,2...\mathrm{n}) \tag{2.16}$$

With the above definitions, a state space model of the dynamic area sensor planning process, shown in Eqn. (2.17), is developed for modeling the process of this feedback-based planning system: viewpoint set V and point clouds Pc are state variables. CAD model M and task constraints TC are inputs and error map E is the output of the system. Symbol $\doteq$ represents that V and Pc are accumulated results from iterations.

$$\begin{cases} V \doteq f_0(M, TC) \bigcup g_k(I) \\ Pc \doteq \Gamma(V) \\ E = \Delta(M, Pc) \end{cases} \tag{2.17}$$

As shown in Fig. 3, given CAD model M and a set of task constrains TC, a group of viewpoint $V_0$ will be initialized first using bounding box method. Point clouds Pc is then generated according to $V_0$. At this time, V equals to $V_0$. Two functions will then be executed according to this viewpoint set V: point cloud generation $\Gamma$ and viewpoint evaluation $\Lambda$. If the quality of point clouds are not satisfied by the cost function Q, a group of new viewpoints will be generated through function $g_k$. This measured point cloud is iteratively updated until the viewpoint evaluator reports a stopping signal to close the switch K. Based on this automatic control scheme, the error map will be generated only when the measured point cloud is considered to be qualified, which then compare to its CAD model M for surface quality inspection.

The proposed feedback system is stable if the iteration process converges. A cost function Q is defined by Eqn. (2.18). In which, *k* represents the $k^{th}$ iteration, $u_k$ represents the total area of holes, and $v_k$ represents the measurement error. Then, Q is going to be optimized by adding more viewpoints such that both $u_k$ and $v_k$ are minimized. The cost function is formed on a complex plane where $u_k$ and $v_k$ represents the real and imaginary part respectively. A weight number *w* is defined to scale the ratio between $u_k$ and $v_k$.

$$Q = \left\| w \cdot u_k + j v_k \right\| \tag{2.18}$$

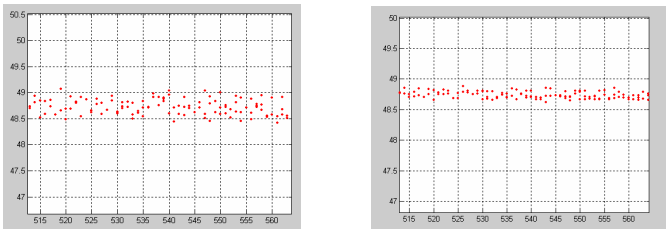
This recursive planning process stops when Q is minimized into a tolerant value. Ideally, Q is expected to be 0, which means the present point cloud has no holes and is exactly matched to the point cloud measured in the previous step. Because new viewpoints will always bring more 3D points to update the point cloud, holes can be filled by continuously adding viewpoints through iterations. In another aspect, accuracy of the point clouds can also be improved by redundant points. Therefore, the system will be stable in finite iterations *k*.

## 3. EXPERIMENTAL IMPLEMENTATION AND RESULTS

An accuracy test is conducted first on flat surface with difference intensity contrast property. Fig. 9 shows the deviations of measurements. Fig. 9(a) displays the result in low contrast region and Fig.9 (b) displays the result in high contrast region. The horizontal axis represents the image coordinates and the unit is pixel, the vertical axis shows the magnitude of the uncertainties, the units is mm. It can be seen that the deviation in high contrast region of measurements in the high contrast region is smaller than it in the low contrast region.

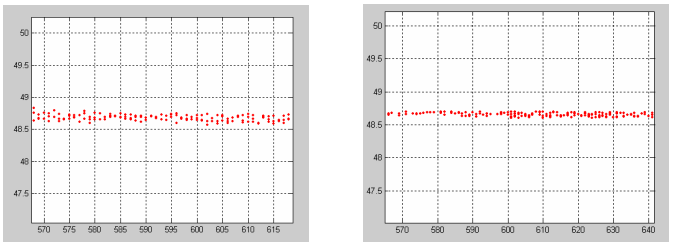
A method to solve this problem is to move the range sensor close to the low contrast surface region and take multiple measurements on that viewpoint. Collecting and averaging the point cloud data from multiple measurements can reduce measurement errors. Testing results on a flat surface are shown in Fig. 10, where (a) shows the deviation of the measured surface from a single shot and (b) shows the average height calculated from multiple shots. Meanwhile, compare Fig. 10(a) to Fig. 9(a), it can be seen that the deviation becomes smaller when the sensor is closer to the part

surface. The trade off of this method is, more calculations have to be executed for measuring this surface, such as the identification of low contrast region, planning new viewpoints, and data processing of 3D shape measurements.



(a) (b)

Fig. 9: Deviation of measurement errors (a) Results in a low contrast region (b) Results in a high contrast region.



(a) (b)

Fig. 10: Reduce measurement uncertainty in a low contrast area by averaging data from multiple measurements (a) Results from a single shot (b) Averaged results from multiple shots.

The iteration process of this test is conducted ten times, the standard deviation of points all over the point cloud are calculated. The variation of the obtained point clouds becomes smaller and smaller, and the feedback process is going to stop when the cost function Q reaches to a predetermined set point 0.025. Table 1 shows the data calculated during this process. And Fig. 11 illustrates the converge process of the deviation σ. It is noticed from Tab. 1 that the cost function Q is also converged with σ.

| Iterations $k$ | Deviation, σ | Time, (sec) | No. of Points, | Cost, Q |
|---|---|---|---|---|
| $k$ =1 | 0.1934 | 41.2 | 243,185 | 0.1718 |
| $k$ =2 | 0.1387 | 72.0 | 363,348 | 0.1021 |
| $k$ =3 | 0.1113 | 101.0 | 356,734 | 0.0602 |
| $k$ =4 | 0.1011 | 130.4 | 365,343 | 0.0450 |
| $k$ =5 | 0.0975 | 159.4 | 375,239 | 0.0378 |
| $k$ =6 | 0.0951 | 178.6 | 364,347 | 0.0329 |
| $k$ =7 | 0.0944 | 207.9 | 353,283 | 0.0280 |
| $k$ =8 | 0.0931 | 238.1 | 345,342 | 0.0265 |
| $k$ =9 | 0.0928 | 267.7 | 364,812 | 0.0253 |
| $k$ =10 | 0.0921 | 295.3 | 345,453 | 0.0249 |

Tab. 1: Improve the measurement accuracy from the recursive process.

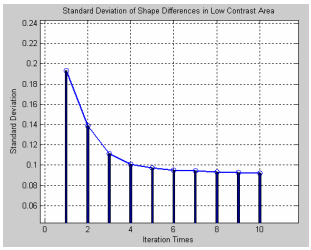



Fig. 11: Stability analysis of the feedback system.

The developed feedback-based automated 3D shape inspection system has been implemented on a PUMA 560 robot. An automotive body part, pillar (part number m32510), is used for testing the developed system. Viewpoints can be tested and visualized using NuGraf™. Description of system setup and viewpoint simulation has been introduced in a previous publication [9].

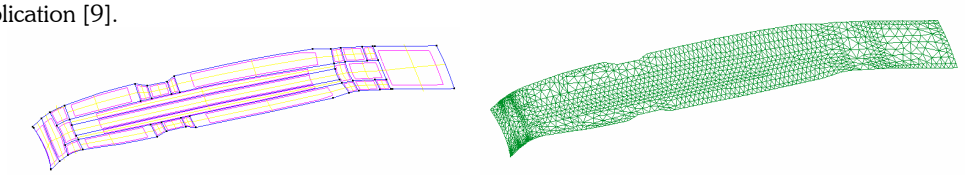


Fig. 12: CAD model of an automotive part, pillar (m32510).

Fig. 12 displays a tessellated CAD model of this pillar. There are totally 2,716 triangles used to represent the geometry shape of this part. Fig. 13 shows a single point cloud measured on one viewpoint. The measured point clouds are shown in Fig. 14. The measured point clouds can be registered together to form a point cloud, which represents the geometry shape of the real part.

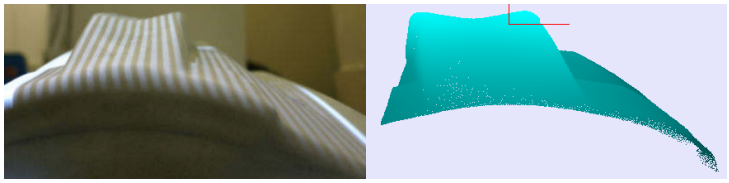


Fig. 13: Projected strips and measured point clouds.

Fig 14 illustrates the point clouds measured on four viewpoints. Software for displaying the point clouds are developed using C++ and OpenGL. Fig. 15 shows two point clouds correspond to two viewpoints, Fig. 15(a) shows a point cloud measured on a viewpoint from set $V_0$, which is a viewpoint generated from CAD model. Fig. 15 (b) displays a point cloud generated in system feedback to fill the holes.

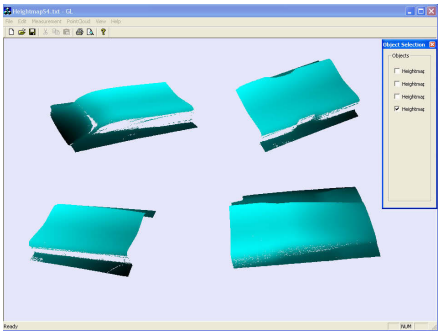


Fig. 14: Updated views derived from the proposed online area sensor planning system.

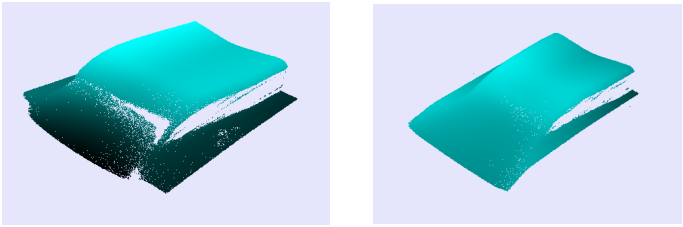


Fig. 15: Filling holes by placing the area sensor to another viewpoint.

The measured point clouds can be registered to its CAD model according to the base coordinate frame of the PUMA robot [11]. Shape difference between the CAD model and the registered point cloud can be illustrated by a color-coded error map. As shown in Fig. 16, at the left side of part, the surface is higher than the designed dimension. The reason is that this part is punched with pressures, since the curvature is not smooth as the right side, forces are not only from the top but also from the left, which squeeze the surface up than it should be. This part of area need to be checked again to ensure it meets quality requirements.

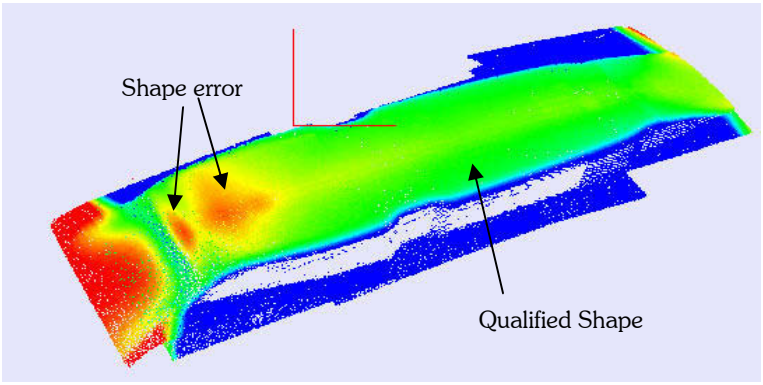



Fig. 16: The error map of a pillar (m32510), developed by the feedback-based 3D shape inspection system.

## 4. CONCLUSIONS

The general framework of a feedback-based robot area sensor planning system is introduced. This system can recursively add viewpoints to improve the quality of a point cloud. A CAD-guide sensor planner is used to find an initial set of viewpoints. However, image noises and poor surface illumination properties cannot be predicted from CAD model, which are often related to measurement uncertainties. New viewpoints estimated from the measured point clouds can be added to solve those problems. A feedback-based planning process is developed to recursively add more viewpoints for measurement till the quality of the entire point cloud is satisfied. Point clouds obtained from this feedback system can be output directly for shape verification. Mathematical models of this system are developed. Experiments on an automotive body part show that the feedback-based planning system is effective in developing an error map for quality evaluation.

## 5. ACKOWNLEDGEMENT

The work described in this paper is supported under NSF Grant IIS-9796300, IIS-9796287, EIA-9911077, DMI-0115355, and FORD Motor Company university research project.

## 6. REFERENCES

[1] Scott, W. R.; Roth, G.: View planning for automated Three-Dimensional object reconstruction and inspection, ACM Computing Surveys, 35(1), 2003, 64-96.

[2] MotionNet.com, http://www.motionnet.com/cgi-bin/search.exe?a=cat&no=3787

[5] Pito, R.: A Solution to the Next Best View Problem for Automated Surface acquisition, IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(10), 1999, 1016-1030.
[6] Reed, M. K.; Allen, P. K.; Stamos, I.: Automated Model Acquisition from Range Images with View Planning, IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition, 1997, 72-77.
[7] Impoco, G.; Cignoni, P.; Scopogno R.: Closing Gaps by Clustering Unseen Directions, International Conference on Shape Modeling and Applications, 2004, 307-316.
[8] Shi, Q.; Xi, N.; Chen, H.; Chen Y.: Calibration of Robotic Area Sensing System for Dimensional Measurement of Automotive Part Surface, IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005, 3898-3903.
[9] Shi, Q.; Xi, N.; Chen, H.; Chen Y.: Integrated Process for Measurement of Free-Form Automotive Part Surface Using a Digital Area Sensor, IEEE International Conference on Robotics and Automation, 2005, 580-585.
[10] Phong, B. T.: Illumination for Computer Generated Pictures, Communications of the ACM, 18(6), 1975, 311-317.
[11] Shi, Q.; Xi, N.; Chen, Y.; Sheng, W.: Registration of Point Clouds for 3D shape inspection, IEEE International Conference on Intelligent Robots and Systems, Oct. 2006.
[12] Sheng, W: CAD-Based Robot Motion Planning For Inspection in Manufacturing, Ph.D. Dissertation, 2002.