A Dynamic Sensing-and-Modeling Approach to Three-Dimensional Point- and Area-Sensor Integration[1]

Yunbao Huang

Xiaoping Qian
e-mail: qian@iit.edu

Mechanical, Materials and Aerospace Engineering,
Illinois Institute of Technology,
Chicago, IL 60616

The recent advancement of 3D non-contact laser scanners enables fast measurement of parts by generating a huge amount of coordinate data for a large surface area in a short time. In contrast, traditional tactile probes in the coordinate measurement machines can generate more accurate coordinate data points at a much slower pace. Therefore, the combination of laser scanners and touch probes can potentially lead to more accurate, faster, and denser measurements. In this paper, we develop a dynamic sensing-and-modeling approach for integrating a tactile point sensor and an area laser scanner to improve the measurement speed and quality. A part is first laser scanned to capture its overall shape. It is then probed via a tactile sensor where the probing positions are dynamically determined to reduce the measurement uncertainty based on a novel next-best-point formulation. Technically, we use the Kalman filter to fuse laser-scanned point cloud and tactile points and to incrementally update the surface model based on the dynamically probed points. We solve the next-best-point problem by transforming the B-spline surface's uncertainty distribution into a higher dimensional uncertainty surface so that the convex hull property of the B-spline surface can be utilized to dramatically reduce the search speed and to guarantee the optimality of the resulting point. Three examples in this paper demonstrate that the dynamic sensing-and-modeling effectively integrates the area laser scanner and the point touch probe and leads to a significant amount of measurement time saving (at least several times faster in all three cases). This dynamic approach's further benefits include reducing surface uncertainty due to the maximum uncertainty control through the next-best-point sensing and improving surface accuracy in surface reconstruction through the use of Kalman filter to account various sensor noise. [DOI: 10.1115/1.2714585]

Keywords: dynamic shape sensing, reverse engineering, Kalman filter, B-spline surface, sensor fusion, coordinate metrology

1 Introduction

The recent advancement of 3D non-contact laser sensors enables fast measurement of parts by generating a huge amount of coordinate data for a large surface area. For example, an area laser scanner, a laser-based 3D sensing device that can measure a surface area in one setup, can produce hundreds of thousands of data points in less than 1 min with accuracy on the order of a hundredth of a millimeter. Its rapid data acquisition rate and the continuously improving measurement accuracy have led to the wide adoption of 3D laser sensors in automotive, aerospace, and consumer industries [1,2]. However, despite being a rapid sensing tool, an area laser scanner is vulnerable in shiny surface regions due to the specular reflection, in parts of complex shape due to the occlusion, and in high curvature area due to the large incidence angle and limited sensor resolution. Note that an area scanner is especially susceptible to occlusion due to the large baseline distance between the energy beam source and the detector, which requires the object surface to be in the line-of-sight to both the source and the detector [3]. Figures 1–3 demonstrate the poor data quality or missing data, due, respectively, to high curvature, occlusion, and the shiny surface when an area laser scanner is used to measure two parts.

On the other hand, conventional touch probes in coordinate measuring machines (CMMs) enable more accurate measurement of parts at a much slower speed. The touch probes can only generate coordinate points approximately at the speed of 60 points per minute with accuracy ranging from hundreds of nanometers to several micrometers. Although faster analog probes are also available, they are still much slower than optical area sensors. However, due to the contact mode and the small probe size of tactile sensing, touch probes are not sensitive to surface reflection conditions and able to produce more reliable results on the high curvature areas.

Therefore, due to the complementary characteristics of *area* laser scanners and *point* touch probes in sensing speed, coverage, accuracy, accessibility, surface conditions, and surface geometry, the integration of a laser scanner with a touch probe can potentially dramatically improve the ways that parts are currently measured. An integrated multisensor coordinate measurement system can benefit in measurement accuracy from touch probes, in measurement speed from area laser sensors, and in part versatility from the availability of multiple sensors.

In this paper, we term the integration of multiple sensors with different dimensionalities in terms of sensor spatial coverage, such as the integration of an area sensor with a point sensor, as *dimensionally heterogeneous sensor integration*. This forms a contrast

[1]An earlier version of this paper has been accepted for publication in 2006 ASME International Conference on Manufacturing Science and Engineering, Ann Arbor, MI, USA, Oct 8-11.

Contributed by the Manufacturing Engineering Division of ASME of publication in the JOURNAL OF MANUFACTURING SCIENCE AND ENGINEERING. Manuscript received June 23, 2006; final manuscript received November 3, 2006. Review conducted by Shreyes N. Melkote.

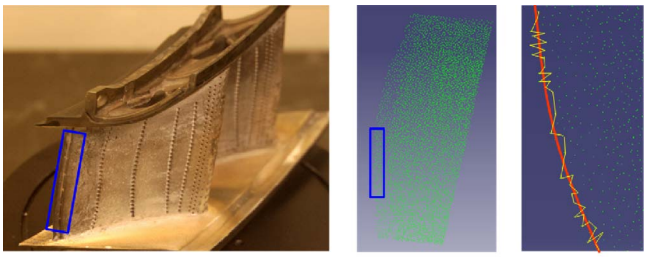


Fig. 1 Inaccurate data on the feature edges by area scanners

to prior multisensor integration wherein the emphasis is on how to integrate multiple dimensionally homogeneous sensors (either point sensors alone or area sensors alone). Even though the integration of area vision sensors and tactile probes has been explored in the past, such integration tends to be *cooperative* where area sensors provide approximate shape to enable the automatic planning for the point sensing. In our approach, we focus on *competitive* integration where the area laser scanned point cloud and point sensed data are fused according to their respective noise to reduce the surface uncertainty. In addition, point sensing is used to improve the coverage loss in area sensing.

More specifically, area scanning is conducted first to construct the base surface. A *next-best-point (NBP)* problem is then formulated to dynamically determine where the best point-sensing location is to reduce the overall surface uncertainty. The newly sensed point is added to the earlier reconstructed surface through the Kalman filter and the next best point-sensing location is again determined based on its uncertainty value in the updated surface. The process continues until the overall surface uncertainty distribution meets the specified value. The overall flowchart of the point- and area-sensor integration is shown in Fig. 4.

Figure 5 shows that, for a set of surfaces with the same rms errors, they are very different in shape due to the missing data. Further, larger surface uncertainty can be observed on the missing data areas, indicating that additional data points are required to improve surface quality in these areas (see Fig. 5(*e*)). The surface uncertainty distribution is thus utilized to determine where to acquire additional points using point sensors in this paper.

In the dimensionally heterogeneous sensor integration process, the amount of data from different sensors is asymmetric due to the difference in their sensing coverage domain. That is, the amount of points scanned by an area laser scanner is much larger than that from a point sensor. Therefore, an effective means to process the

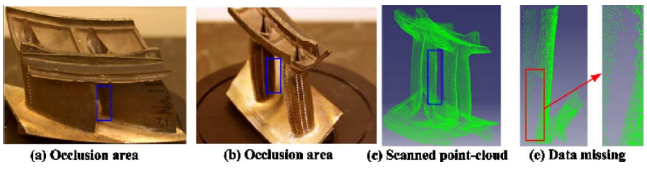


Fig. 2 Incomplete data caused by occlusion in area scanners

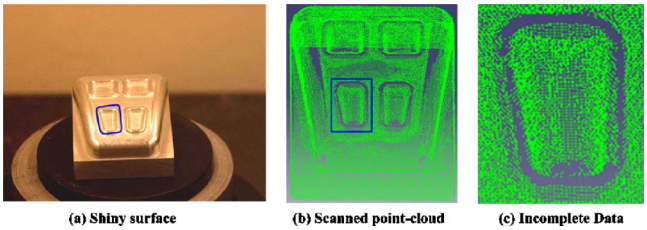


Fig. 3 Incomplete data in shiny surface measurement through area scanners

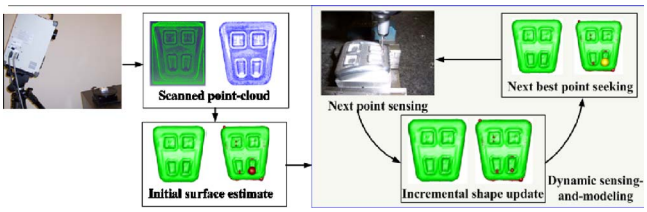


Fig. 4 A dynamic sensing-and-modeling approach for integrating dimensionally heterogeneous sensors

asymmetric amount of heterogeneous data is needed for such sensor integration. In this paper, the Kalman filter is introduced to dynamically update the shape with the point-probed data to avoid reconstructing the whole shape from scratch when each time a new measurement point is added. The technical contribution of this paper includes the following:

- We extend the classical Kalman filter, a set of mathematical equations for estimating the process state, to surface reconstruction. It allows the incorporation of a priori information, along with new measurement information, for the state (surface) updating. Therefore, the large number of laser scanned data points can be reduced to a compact prior surface representation. This enables efficient surface updating by allowing the tactile points to be dynamically and incrementally added into the surface without reference to the laser scanned data.
- We mathematically transform the uncertainty distribution of a B-spline surface into a tensor B-spline form, which enables the rapid and accurate computing of the surface point with the largest uncertainty, hereinafter the *next-best-point (NBP)*. This forms the basis of our dynamic sensing-and-modeling approach.

The remainder of this paper is organized as follows. Section 2 reviews prior work in multisensor coordinate metrology. Section 3 presents the theoretical formulation in using Kalman filter for dynamic surface reconstruction. Section 4 details the technique on determining the NBP. Section 5 describes how to sense the dynamically determined optimal point and how to update the surface with the sensed point. Section 6 presents the experimental results. This paper concludes in Sec. 7.

2 Literature Review

Sensor planning for range sensors has been researched in the past. Sensor planning [4], including the next-best-view problem to minimize the number of range views [5,6], and the optimal setup for area laser scanners [7], has been explored. High-level control of data acquisition is demonstrated in [8] based on the uncertainty of a regression model. However, only simple super-ellipsoid model is employed to represent the object. B-spline surface is utilized to modeling the 3D free-form object and uncertainty based viewpoint plan is proposed in [9]. However, in these approaches, only area range sensors are considered.

Inspection planning for tactile probes in CMM has been an active research topic, wherein various methods are proposed to shorten inspection time and to automate the inspection process [10–12]. Among various inspection planning tasks, surface sam-

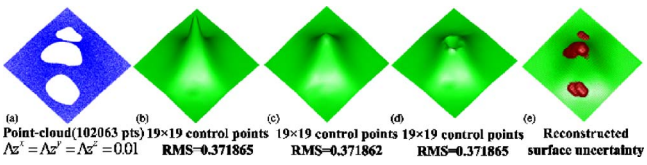


Fig. 5 Reconstructed surface and uncertainty

pling has also been studied to minimize the number of required sampled points and to reduce the resulting uncertainty. In particular, uniform sampling pattern and ten-times ratio between the number of measured data and the number of model parameters are often adopted [13]. A few researchers have tried to optimize the sample size based on user skills [14], and to seek the optimal locations of sample points in [15,16]. Even though the sample size and locations are optimized to improve the reconstructed surface quality and the inspection efficiency using probe sensors in a CMM, all these approaches use single tactile point sensors.

Despite the fact that multisensor sensors have been found coexisting in some advanced CMM systems such as OGP and Mahr systems, information automation is not yet available. Therefore, *cooperative* sensor integration has been proposed to automate the coordinate acquisition, in which vision systems are used to create an approximate shape to guide the tactile sensing [17–19]. Recently, vision and touch probe sensors are integrated to improve coordinate acquisition speed [15,20]. Algorithms were also developed for estimating and reasoning unknown surface areas [21]. In this paper, we use the *competitive* combination of a laser scanner and a touch probe to improve inspection speed and to reduce measurement uncertainty.

In the computer vision community, a probabilistic surface model in combination with the Kalman filter has been proposed to build tensor parametric surface [22–24] and lines and planes [25]. In our approach, we extend the underlying surface representation from single level to multilevel [26]. More importantly, we use the Kalman filter to reduce the large number of scanned data into one surface in an efficient mode so that the surface state can be efficiently updated without reference to the scanned data when additional sensed points are available. Further, we use Kalman filter to dynamically, actively, and incrementally update the surface based on the point-probe data.

3 Dynamic Surface Reconstruction With the Kalman Filter

This section gives the mathematical basis for the dynamic surface reconstruction, including (1) B-spline surface representation, and (2) Kalman filter for surface reconstruction and its properties. Here, the dynamic surface reconstruction refers to a surface reconstruction process in which the surface is dynamically constructed or updated based on prior information and the new measurement data.

3.1 B-spline Surface. The B-spline surface has been widely used in product design and manufacturing. Therefore, it is selected to represent the surface reconstructed from the point-cloud.

A bi-cubic B-spline surface has the form

$$S(u,v) = \sum_{i,j}^{n} N_i(u) N_j(v) P_{ij} \tag{1}$$

where N is B-spline shape function and P_{ij} is the ijth control point. The equation can also be expressed in matrix form

$$S(u,v) = A(u,v)P \tag{2}$$

where $A(u,v)$ is the B-spline shape function matrix, and P means the collection of control points.

To model the surface from the point-cloud, most shape reconstruction methods are based on least-squares fitting, and they do not explicitly consider data noise incurred during the measurement process. It is often assumed the noise corrupting the data is of zero mean. This has two immediate ramifications. First, *reconstructing surfaces from measurement data would lead to inaccurate surfaces if data uncertainty is not properly accounted for*. Numerous studies have shown that least-squares estimates may be completely perturbed when the data contains noise of different variances, or non-Gaussian [27]. Therefore, *such single surface representation without quality description is not sufficient for many subsequent applications*. To account for data uncertainty and to result in more robust surfaces, the Kalman filter is introduced to produce the optimal estimate of the surface and its uncertainty.

3.2 Kalman Filter. The Kalman filter was first proposed by Kalman in 1960 [28]. It is a method to recursively estimate the internal state of a system based on the observation of the system's external behaviors [29]. To do this, a state-space model needs to be first set up to describe the system.

State-Space Model. A mechanical or electrical system can be characterized by several finite variables, which are called the internal state. Usually, the internal state cannot be directly measured but its external behavior can be sensed. The Kalman filter [29] addresses the general problem of trying to recursively estimate the state of a discrete-time controlled process that is governed by the linear *stochastic* difference equation

$$x_l = Cx_{l-1} + Bu_{l-1} + w_{l-1} \tag{3}$$

where x_l and x_{l-1} are the states of the time step l and $(l-1)$, respectively, C is the state transition matrix of the process from previous time step $(l-1)$ to the current step l, and u_{l-1} is the system input of the time step $(l-1)$. Matrix B relates the system input u_{l-1} and the state x_l, and w_{l-1} is the associated independent, white, and Gaussian noise.

Observations on the internal state can be modeled in the form

$$z_l = Hx_l + v_l \tag{4}$$

where z_l is the external measurement of x at the lth time step, H is the measure matrix relating the current state x_l to the external measurement z_l, and v_l is the associated measurement error, which is again assumed to be independent, white, and Gaussian.

Kalman Gain. With the state-space model, Kalman filter optimally estimates the internal state using the Kalman gain.

Let $\hat{x}_l^-$ be the a priori state estimate at the step l with uncertainty Λx_l^-, and $\hat{x}_l$ be the a posteriori state estimate with uncertainty Λx_l. We can estimate $\hat{x}_l$ through a linear combination of $\hat{x}_l^-$ and the residual between the actual measurement z_l and a measurement prediction $H\hat{x}_l^-$ is given by

$$\hat{x}_l = \hat{x}_l^- + K(z_l - H\hat{x}_l^-) \tag{5}$$

where K is the Kalman gain, which is determined by minimizing the a posteriori uncertainty Λx_l [30]. Let Λz_l be the uncertainty of measurement z_l, the optimal Kalman gain K can be obtained by [29]

$$K = \Lambda x_l^- H^T (H \Lambda x_l^- H^T + \Lambda z_l)^{-1} \tag{6}$$

From Eqs. (5) and (6), we see that $\hat{x}_l$ weighs more on $\hat{x}_l^-$ when $\Lambda x_l^- \to 0$. Otherwise, $\hat{x}_l$ weighs more on the residual between $H\hat{x}_l^-$ and z_l. Moreover, measurement uncertainty Λz_l is explicitly accounted for to obtain the optimal state estimate.

Update Equations. The Kalman filter estimates the state by using a form of prediction and correction, which includes predictor equations and corrector equations.

Predictor equations have the following form.

The a priori state prediction equation:

$$\hat{x}_l^- = C\hat{x}_{l-1} + Bu_{l-1} \tag{7}$$

The a priori state uncertainty equation:

$$\Lambda x_l^- = C \Lambda x_{l-1} C^T + Q \tag{8}$$

where Q is the uncertainty covariance of process noise w_{l-1}. When $l=1$, the initial state estimate $\hat{x}_0$ and its uncertainty $\Lambda \hat{x}_0$ can be given by prior information, which is able to integrate a priori estimate of internal state.

Corrector equations have the following form.

$$\text{Kalman gain:} \quad K_l = \Lambda x_l^- H^T (H \Lambda x_l^- H^T + \Lambda z_l)^{-1} \tag{9}$$

The estimate updating equation with measurement z_l:

$$\hat{x}_l = \hat{x}_l^- + K_l (z_l - H\hat{x}_l^-) \tag{10}$$

where K_l is the same as K in Eq. (5).

$$\text{The state uncertainty updating equation:} \Lambda x_l = (I - K_l H)\Lambda x_l^- \tag{11}$$

The predictor and corrector equations form a basis for recursive updating. The predictor equations are to predict the a priori state and its uncertainty based on the previous state estimate. The a priori state estimate is then corrected in the corrector equations by the new measurement to obtain a posteriori state estimate. This recursive nature of Kalman filter is a very appealing feature, which makes it possible to apply newly measured data for state updating without requiring all previous data kept in storage [29]. In addition, The Kalman filter combines all available measurement data, plus prior knowledge about the system and measuring devices, to produce an estimate of the desired internal state in such a manner that the error is minimized statistically [31].

3.3 Kalman Filter for Surface Reconstruction. For one measurement point $z \in Q$ on the surface with parameter (u_z, v_z), we can get

$$z = A(u_z, v_z)P + \varepsilon \tag{12}$$

where ε is measurement noise.

In the terminology of Kalman filter, the B-spline surface represents a linear system between the internal surface state and external observations. The collection of control points P constitutes the internal state of the object shape. The point-cloud measured from surface forms the external observations of B-spline surface. Thus, the measurement z is the measured surface point with uncertainty Λz. $A(u_z, v_z)$ corresponds to the measure matrix H in Eq. (4).

From the state-space model of surface fitting, we have $x=P$, $z=z$, u_{l-1} and $w_{l-1}=0$, and the state transition matrix $C=I$.

The predictor equations of the Kalman filter changes to

$$\hat{x}_l^- = \hat{x}_{l-1} = P_{l-1} \tag{13}$$

$$\Lambda x_l^- = \Lambda x_{l-1} = \Lambda P_{l-1} \tag{14}$$

The corrector equations changes to

$$\hat{x}_l = P_l = P_{l-1} + K_l [z - A(u_z, v_z)P_{l-1}] \tag{15}$$

$$\Lambda P_l = [I - K_l A(u_z, v_z)]\Lambda P_{l-1} \tag{16}$$

where K_l is the Kalman gain, which changes to

$$K_l = \Lambda P_{l-1} A(u_z, v_z)^T [A(u_z, v_z)\Lambda P_{l-1} A(u_z, v_z)^T + \Lambda z]^{-1} \tag{17}$$

Substituting K_l, we get the new form of Eq. (16) ([30])

$$\Lambda P_l^{-1} = \Lambda P_{l-1}^{-1} + A(u_z, v_z)^T (\Lambda z)^{-1} A(u_z, v_z) \tag{18}$$

Incrementally fitting all the measurement points with the Eqs. (15) and (16), we then obtain the uncertainty ΛP of control points P.

Further, we put forward the following two lemmas in the context of applying the Kalman filter for surface reconstruction. Here we define the surface reconstruction in batch fitting mode as processing all available measured points and a priori surface estimate at once to produce the surface, and define its incremental fitting mode as iteratively updating the estimated surface one by one with measured data.

LEMMA 1. *For a given set of measurements* $\{z_i\}$ *and the corresponding noise* Λz_i, *the reconstructed surface, its control point* P_n *and its uncertainty covariance* ΛP_n, *can be obtained by fitting in a batch mode and it is equal to that obtained by incrementally fitting with Eqs. (15) and (16).*

Proof. If all the measurements are available, the final reconstructed surface can be computed in the batch fitting mode for the Kalman filter by the following equations (the deducing process can be seen in Appendix A)

$$P_n = \left[(\Lambda P_0)^{-1} + \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} A_i \right]^{-1} \left[(\Lambda P_0)^{-1} P_0 + \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} z_i \right] \tag{19}$$

$$\Lambda P_n = \left[(\Lambda P_0)^{-1} + \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} A_i \right]^{-1} \tag{20}$$

where P_0, ΛP_0 is the initial estimate of surface and its uncertainty estimate, A_i is the B-spline shape function matrix corresponding to the measurement z_i, Λz_i is the uncertainty of measurement z_i.

LEMMA 2. P_n *and* ΛP_n *are independent on the measurement sequence of* z_i, z_j, $(i \neq j)$.

Proof. This can be easily seen from the batch processing Eqs. (19) and (20).

For the same measurements and parameterization, we can also reconstruct a B-spline surface with same number of control points by employing the weighted least-squares method. The reconstructed surface can be represented by

$$P_n = \left[\sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} A_i \right]^{-1} \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} z_i \tag{21}$$

Comparing Eqs. (19) and (21), we can see that the two fitting surfaces are actually equivalent when the initial estimate $P_0=0$ and its uncertainty $|\Lambda P_0| \to \infty$ or more likely that P_0 and ΛP_0 are computed directly from the weighted least-squares method. In summary, the Kalman filter for surface reconstruction has the following properties:

- The resulting surface is independent of the sampling sequence.
- The surface can be incrementally updated and the resulting surface is equivalent to the batch reconstructed surface.

These two properties form the basis of our proposed dynamic sensing-and-modeling approach. That is,

- The final surface is independent of the fitting sequence of any two of sampled probing points z_i, z_j, $i \neq j$, which ensures the dynamic and sequential sensing-and-modeling approach would lead to the same final surface and surface uncertainty for the same set of measurement points, regardless of the order of the measurement.
- When sequentially sampling n number of probing points z_i, $i=1, \ldots, n$, the end reconstructed surface using Eqs. (15) and (16) in incremental fashion is equivalent to that obtained from Eqs. (19) and (20) in the batch processing fashion.

Furthermore, the two fitting modes of Kalman filter will facilitate the dimensionally heterogeneous sensor integration. Suppose m is the number of large amount of scanned data points, n is the number of fitting surface's control points. From Eqs. (15) and (16), it can be seen that the computational complexity for the Kalman filter is $O(n^2)$ when fitting each single data point in incrementally fitting mode, and is $O(m \times n^2)$ when fitting all of the point-cloud. On the other hand, only $O(m+n^3)$ is required for the Kalman filter in the batch processing mode to fit the whole point-cloud (see Eqs. (19) and (20)). Therefore, the two modes have the following significance in the dimensionally heterogeneous sensor integration:

- The batch processing capability allows the large amount of the laser-scanned point-cloud $(m \gg n)$ to be processed effi-

ciently at once (*its computational complexity is* $O(m+n^3)$) as opposed to the large number of iterative use of incremental updating (*its computational complexity is* $O(m\times n^2)$).
- The incremental update allows any additional point probing data to be dynamically and efficiently fused to the reconstructed surface model without reference to the large amount of laser data (*its computational complexity is* $O(n^2)$).

Note that even though the Kalman filter has these advantages and can be used to achieve the robust estimate from measured point-cloud if the underlying model is known, the underlying model structure typically cannot be known previously. Inappropriate model selection tends to under-fit or over-fit noisy measurement data. To circumvent the model selection issue, the multilevel Kalman filter is proposed in [26] to determine the initial surface model estimation, and the Kalman filter can be used to fit to the new measurements based on the selected model.

4 Next-Best-Point Determination

Based on the above introduction of Kalman filter, we choose the multilevel Kalman filter [26] to reconstruct the initial surface from available measurement points. Any additional sensing is placed at lowest quality location (the largest uncertainty point) of the initial reconstructed surface to improve the final reconstructed surface quality. This section details how to dynamically determine the NBP.

4.1 Next-Best-Point Problem Formulation. Due to sensor noise and finite measured points of the surfaces, it is not possible to find a reconstructed surface from those measurements to exactly represent the true surface. Since the internal state-control points Φ_k of the reconstructed surface satisfy Gaussian probability distribution, the uncertainty $\Lambda\Phi_k$ constitutes a measure of information contained in the Gaussian probability distribution $Q(\Phi_k)$ describing the parameter errors

$$Q(\Phi_k) = (\Phi_k - \hat{\Phi}_k)^T \Lambda\Phi_k (\Phi_k - \hat{\Phi}_k) \tag{22}$$

Given a confidence level γ, we can find a number $Q(\Phi_r)$ from the distribution. In addition, the true model must lie in the hyper-ellipsoid defined by $Q(\Phi_r)$. From Eq. (22), it can be seen that smaller size of the ellipsoid means the reconstructed surface has higher confidence and approximates the underlying true model better. Thus, reducing the size of hyper-ellipsoid means improving the confidence or the quality of the reconstructed surface. Actually, the size of hyper-ellipsoid is only dependent of the reconstructed surface uncertainty $\Lambda\Phi_k$, and the determinate of the uncertainty covariance matrix $\Lambda\Phi_k$ is proportional to the square of the volume of the hyper-ellipsoid [8,15].

Therefore, the next-best-point (NBP) problem can be formulated as: *finding a next optimal point to maximally reduce the determinant of uncertainty covariance matrix* $\Lambda\tilde{\Phi}_k$ (denoted as $\det(\Lambda\tilde{\Phi}_k)$).

Let the optimal point be p with parameter (u_{z_p}, v_{z_p}), and uncertainty be Λz_p. For the single level Kalman filter

$$\begin{aligned}(\Lambda\tilde{\Phi}_k)^{-1} &= (\Lambda\Phi_k)^{-1} + [A(u_{z_p}, v_{z_p})]^T (\Lambda z_p)^{-1} A(u_{z_p}, v_{z_p}) \\ &= (\Lambda\Phi_k)^{-1}\{I + \Lambda\Phi_k [A(u_{z_p}, v_{z_p})]^T (\Lambda z_p)^{-1} A(u_{z_p}, v_{z_p})\}\end{aligned} \tag{23}$$

where $A(u_{z_p}, v_{z_p})$ is the B-spline shape function matrix. Hence

$$\Lambda\tilde{\Phi}_k = \Lambda\Phi_k \{I + \Lambda\Phi_k [A(u_{z_p}, v_{z_p})]^T (\Lambda z_p)^{-1} A(u_{z_p}, v_{z_p})\}^{-1} \tag{24}$$

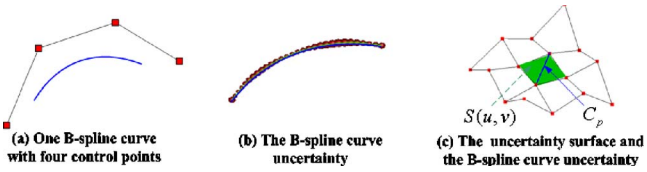


Fig. 6 Geometric interpretation of the uncertainty of a B-spline curve

$$\det(\Lambda\tilde{\Phi}_k) = \det(\Lambda\Phi_k)/\det\{(I + \Lambda\Phi_k [A(u_{z_p}, v_{z_p})]^T (\Lambda z_p)^{-1} A(u_{z_p}, v_{z_p}))\} \tag{25}$$

Because $A(u_{z_p}, v_{z_p})$ is only one row matrix and Λz_p is a value, $I + \Lambda\Phi_k [A(u_{z_p}, v_{z_p})]^T (\Lambda z_p)^{-1} A(u_{z_p}, v_{z_p})$ has only one non-unitary eigenvalue $1 + A(u_{z_p}, v_{z_p}) \Lambda\Phi_k [A(u_{z_p}, v_{z_p})]^T (\Lambda z_p)^{-1}$ [8].

$$\det(\Lambda\tilde{\Phi}_k) = \det(\Lambda\Phi_k)/\{1 + A(u_{z_p}, v_{z_p}) \Lambda\Phi_k [A(u_{z_p}, v_{z_p})]^T (\Lambda z_p)^{-1}\} \tag{26}$$

From Eq. (26), we can see that

$$\min[\det(\Lambda\tilde{\Phi}_k)] \sim \max\{A(u_{z_p}, v_{z_p}) \Lambda\Phi_k [A(u_{z_p}, v_{z_p})]^T\} \tag{27}$$

Thus, the NBP problem can be formulated as *finding an optimal point on the reconstructed B-spline surface with maximal uncertainty*. It is clear that the NBP problem is a global optimization problem.

To our best knowledge, until now there is no efficient deterministic solution to solve such a general problem. The Fedorov exchange method is often used in the random optimization sense to find the optimal point [15,32], but the global optimal point cannot be guaranteed to be found since the point is randomly sampled on the domain.

Fortunately, B-spline has many unique geometric properties such as continuity, local modification, and strong convex hull properties [33], which are very beneficial in B-spline curve/surface boundary box computing, intersection algorithm, and rapid test of geometry interference. We will show in the section below that the uncertainty of B-spline curve/surface can be represented in a tensor B-spline form and therefore a convex hull based NBP searching method is presented.

To help illustrate the basic NBP searching approach, we introduce it in the context of reconstructing a B-spline curve.

4.2 Geometric Interpretations of the Uncertainty of a Reconstructed B-Spline Curve. Assuming a reconstructed B-spline curve with control points P_n and uncertainty covariance matrix ΛP_n. The point p with parameter u on the B-spline curve has the uncertainty

$$\Lambda p = A(u) \Lambda P_n [A(u)]^T \tag{28}$$

where $A(u)$ is the B-spline shape function matrix. Equation (28) can also be written as

$$\Lambda p = A(u) \Lambda P_n [A(v)]^T \quad (v = u) \tag{29}$$

Letting $S(u,v) = A(u) \Lambda P_n [A(v)]^T$, we can see that

- $S(u,v)$ is actually a B-spline surface defined by uncertainty matrix ΛP_n (here we call $S(u,v)$ *an uncertainty surface*).
- The distribution of the uncertainty of the B-spline curve $p(u)$ over u forms a space curve C_p on $S(u,v)$ corresponding to a *pcurve* ($v = u, u \in [0,1]$) in the parametric domain (see Fig. 6). That is, the uncertainty value of curve point $p(u)$ is $\Lambda p = S(u,u)$.

Thus, the problem of finding NBP with maximal uncertainty is equivalent to finding the highest point on the space curve C_p on $S(u,v)$. We can find the highest point in a B-spline surface using

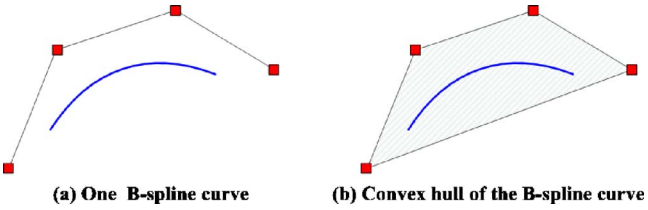


Fig. 7 B-spline curve and its convex hull

the geometry property such as strong convex hull property of B-spline. This new insight forms the basis of our novel geometric algorithm for the NBP determination.

4.3 Convex Hull Property of B-Spline Curve Uncertainty. One important geometric property for B-spline curve/surface is the strong convex hull property, which means that the B-spline curve is completely enclosed in the convex hull defined by its control points.

As shown in Fig. 7, a cubic B-spline curve is defined by four control points, and the B-spline curve is completely enclosed in the convex hull. The B-spline surface has the same strong convex hull property since it is a *tensor product scheme* of B-spline curve (Fig. 8). Thus, the B-spline surface $S(u,v)$ is in the convex hull defined by the elements of uncertainty covariance matrix ΛP_n.

As we know, the B-spline curve's uncertainty is a space curve on the B-spline surface $S(u,v)$. Therefore, the B-spline curve uncertainty is fully contained in the convex hull of the B-spline surface $S(u,v)$, and each B-spline curve segment's uncertainty is in the convex hull of the elements of covariance matrix of defining the curve segment. The strong convex hull property can be employed to compare the uncertainty of the two curve segments.

Thus, we have the following lemma.

LEMMA 3. *Let* $C_{\text{seg }1}$ *and* $C_{\text{seg }2}$ *be the two B-spline curve segments with uncertainty matrix of control points as* $\Lambda P_{C_{\text{seg }1}}$ *and* $\Lambda P_{C_{\text{seg }2}}$. *If* $\max_{ij}(\Lambda P_{C_{\text{seg }1}})_{ij} \leq \min_{ij}(\Lambda P_{C_{\text{seg }2}})_{ij}$, *then the uncertainty of any point on the curve segment* $C_{\text{seg }1}$ *is no greater than that of any point on the curve segment* $C_{\text{seg }2}$.

As shown in Fig. 9, $C_{\text{seg }1}$ and $C_{\text{seg }2}$ are the two B-spline curve segments with respective uncertainties $\Lambda P_{C_{\text{seg }1}}$ and $\Lambda P_{C_{\text{seg }2}}$. If $\max_{ij}(\Lambda P_{C_{\text{seg }1}})_{ij} \leq \min_{ij}(\Lambda P_{C_{\text{seg }2}})_{ij}$, then the uncertainty of any point p_1 on the curve segment $C_{\text{seg }1}$ is no greater than that of any point p_2. The conclusion can be easily obtained from the convex hull property of B-spline curve uncertainty (detailed proof can be seen in Appendix B).

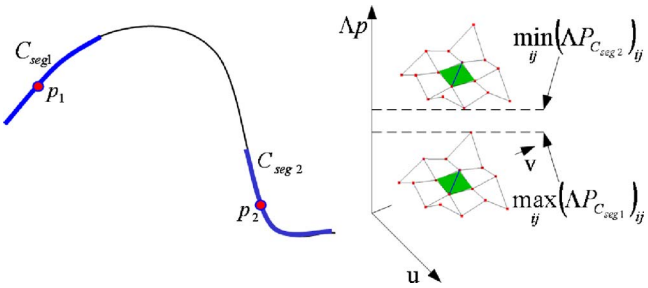


Fig. 9 Uncertainty comparison of two B-spline curve segments

4.4 Convex Hull-Based Next-Best-Point Determination. Based on the analysis of B-spline curve uncertainty property, a convex hull-based NBP determination method is proposed based on a divide-and-conquer strategy. The core idea is (1) to discard (filter) the curve segments of lower uncertainty through the convex hull property of the uncertainty surface, and (2) to further subdivide each B-spline curve segment into more segments where the curve cannot be discarded.

As shown in Fig. 10, the uncertainty surface has only one surface patch. To find the exact point on the B-spline curve with maximal uncertainty, subdivision is performed on the uncertainty surface patch and the corresponding B-spline curve segment. Thus, four subdivided uncertainty surface patches and two B-spline curve segments can be obtained. Among the four patches, only two surface patches and the corresponding subdivided B-spline curve segment uncertainty need to be extracted. If we repeat the subdivision, extraction, and discarding process, the subdivided curve segment finally converges to the optimal point

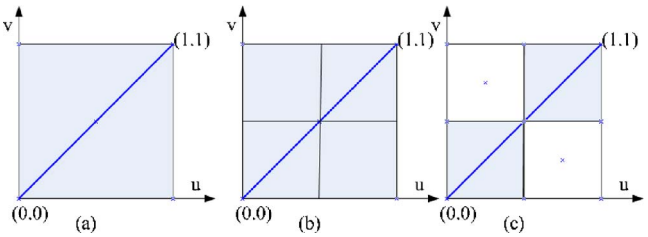


Fig. 10 Uncertainty surface subdivision and extraction

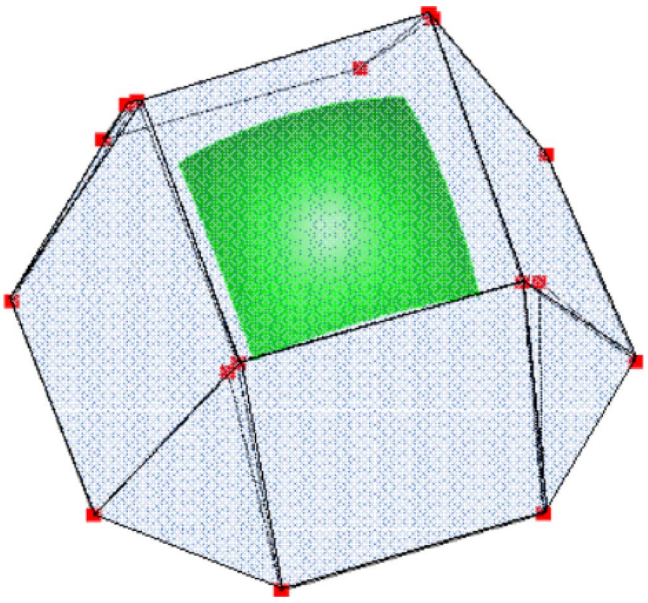


Fig. 8 B-spline surface and its convex hull

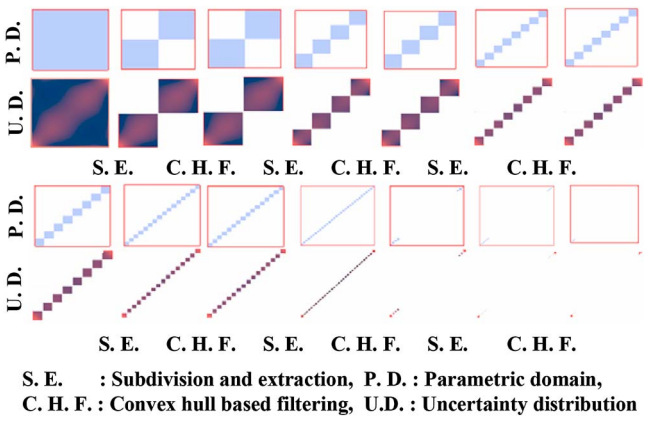


Fig. 11 An example NBP computing process

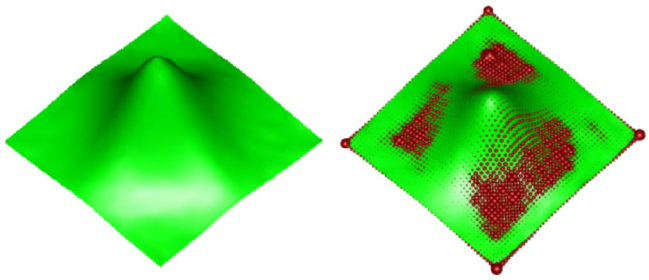


Fig. 12 A reconstructed surface and its uncertainty

within a user-specified tolerance ρ, and thus the corresponding uncertainty surface converges to the uncertainty of optimal point. Figure 11 details such an example NBP computing process on a B-spline curve.

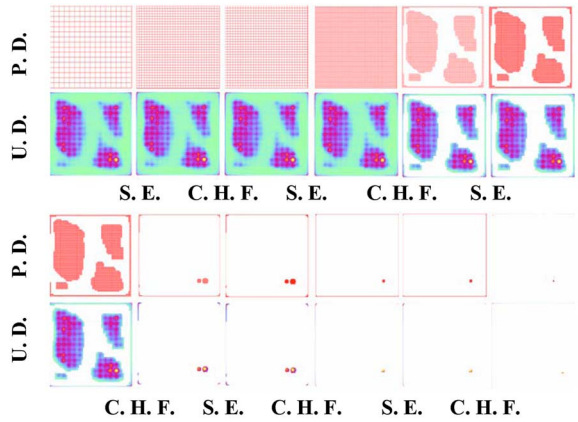
As described, the B-spline surface is a *tensor product scheme* of B-spline curves. The uncertainty of B-spline surface has the same properties such as convex hull and subdivision property. Thus, a convex hull based optimal point location procedure can be easily extended to find the optimal point on the B-spline surface. The uncertainty surface defined by control point's covariance is two dimensional for a B-spline curve. The uncertainty "surface" corresponding to the B-spline surface will be in R^4.

For example, for a given reconstructed surface and its uncertainty (Fig. 12), and the threshold of maximal knot interval length $\theta_{kl}=1.0\times10^{-6}$ (the surface domain is $(0,1)\times(0,1)$). Because of the uncertainty surface for a B-spline surface is defined in R^4, only the surface patches not filtered out are shown in the following optimal point searching process.

From Fig. 13, we can see that the convex hull-based NBP method has the following advantages when compared with the Fedorov exchange method:

- The obtained NBP is guaranteed to be less than ρ (any prior given tolerance) from the true globally optimal point.
- Multiple optimal solutions can be obtained if there are more than one point that has the same uncertainty on the B-spline curve/surface.

These two characteristics enable efficient NBP computing to support the overall dynamic sensing-and-modeling approach.



S. E. : Subdivision and extraction, P. D. : Parametric domain, C. H. F. : Convex hull based filtering, U.D. : Uncertainty distribution

Fig. 13 NBP search process in a B-spline surface

5 Dynamic NBP Sensing and Incremental Surface Update

Given the computed NBP, it can then be sensed, and the reconstructed surface can be updated. Then the next optimal point can be determined, sensed, and the surface can then be updated. These steps constitute a dynamic process to achieve a high quality surface, as shown in Fig. 4. How to sense the next optimal point, how to update the fitting surface with sensed points, and when to terminate the procedure are described in this section.

5.1 Dynamic NBP Sensing. After the next point to be sensed is found through the convex hull-based optimal point determination method, the optimal point needs to be transformed from design coordinate system (DCS) to the measurement coordinate system (MCS) as the next sensing location on the physical part. In DCS, the point-cloud is modeled and used to reconstruct surface. The MCS is the coordinate system where physical part is sensed by the CMM probe.

The transformation can be determined by registrating the key features such as the attached spheres or planes. Then the actual estimate position in MCS can be computed through the transformation. With the transformed position, a touch probe can be driven to the nearest location of the target point and the actual point can be probed on the physical part.

5.2 Incremental Surface Update. With the sensed point(s) and the knowledge on the point sensor uncertainty, the surface needs to be rapidly updated to re-evaluate the surface and to determine the next optimal point. The Kalman filter is employed to update the reconstructed surface in an incremental fashion.

For single level Kalman filter, the updated surface can be computed using Eqs. (15) and (16) directly. The incremental fashion means that the end reconstructed surface can be obtained in real-time without reconstructing the whole surface from scratch. As described in Sec. 3.3, the incremental updating of the Kalman filter is very efficient even in the case of large amount of laser scanner data.

5.3 Terminating Criterion. In the dynamic sensing-and-modeling framework, there is a need to determine *how many points need to be sensed to achieve the desired quality surface*. Traditionally, the number of sampled points is required to be ten times the number of parameters in the model [13]. Since the dynamic sensing-and-modeling procedure works in the iterative fashion, we essentially answer the question as to *when to stop further point sensing*.

In this paper, several terminating criteria are used. Assuming the model has been properly reconstructed, a terminate criterion is suggested based on the largest uncertainty value on the surface. That is, given the maximal surface uncertainty threshold θ_{us}, the terminating criterion of uncertainty can be defined as follows: checking if the maximal surface uncertainty satisfies $\delta\leqslant\theta_{us}$. If $\delta\leqslant\theta_{us}$, the dynamic sensing-and-modeling procedure would be terminated. Otherwise the process repeats until the maximal uncertainty is below the specified uncertainty threshold θ_{us}. Herein (1) the maximal uncertainty can be obtained from the convex hull-based optimal point determination algorithm, (2) uncertainty threshold θ_{us} can be taken as the average uncertainty value of sampled low uncertainty points to make the surface uncertainty uniformly distributed, (3) to compute the average value of low uncertainty points on the surface, we uniformly sample N points and view the uncertainty of each point as intensity, and extract the points with low "intensity" through isodata algorithm developed by Ridler and Calvard [34].

6 Experimental Implementation

Three examples are presented below to validate the dynamic sensing-and-modeling procedure. The first example is a synthetic surface and it shows how the large uncertainty of reconstructed

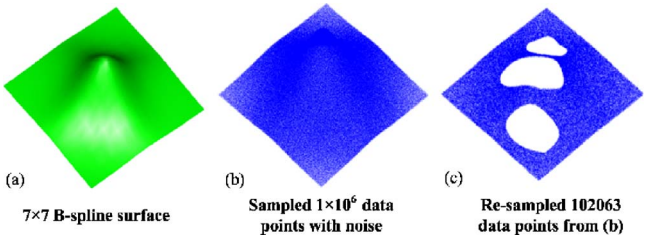


Fig. 14 Sampled incomplete data point-cloud

surface caused by missing data in laser scanning can be reduced to achieve higher quality surface. The other two examples are based on the measurement of two physical parts using Minolta Vivid 910 (Accuracy: $X: \pm 0.22$ mm, $Y: \pm 0.16$ mm, $z: \pm 0.10$ mm; Working distance: 40 to 500 mm; Field of view: 0.6 to 1.2 m; Measurement speed (Fine mode): one scan/2.5 s) and Mitutoyo CMM BH303 (Travel distance: 304.8 mm $\times$ 304.8 mm $\times$ 304.8 mm). They illustrate how dynamic sensing can be used to improve the area measurement of surface impaired by occlusion, shiny reflection, or high curvature areas such as feature edges.

6.1 Examples. In Fig. 14, 1×10^6 data points (Fig. 14(*b*)) are uniformly sampled from a known B-spline surface (7×7 control points) (Fig. 14(*a*)) and are added to the Gaussian noise (variance $\Lambda z^x = \Lambda z^y = \Lambda z^z = 0.01$). In the noisy data, only 102,063 data points (Fig. 14(*c*)) are then selected to simulate the measurement with data missing on the surface.

First, we assume the model is not known. The multilevel Kalman filter is applied to reconstruct the surface model. According to the proposed model selection criteria in [26], the five-level B-spline surface is reconstructed with 19×19 control points as shown in Fig. 15.

Given the surface's maximum uncertainty threshold $\theta_{us} = 0.0012$, which can be the average value of uniformly sampled points' uncertainty, the dynamic sensing-and-modeling procedure is employed to determine the optimal CMM sensing points (assuming Gaussian noise $\Lambda z^x = \Lambda z^y = \Lambda z^z = 0.0001$) to reduce the surface uncertainty.

The two actual examples shown in the introduction section are also used to demonstrate the dynamic sensing-and-modeling approach. Since our current experimental setup does not allow programmable CMM sampling, only limited points are probed at the surface area whereby there is missing information during laser scanning. With those additional measured points, a dynamic procedure is applied to select the optimal points and incrementally update the surface.

To measure the nozzle part (83 mm $\times$ 203 mm $\times$ 108 mm) shown in Figure 2, a 3D laser scanner (Minolta VIVID 910) is used to scan the whole part surface with total 12 range views. One airfoil surface can be completely covered by laser scanning and the other airfoil surface has incomplete coverage from laser scanning due to occlusion (Fig. 2). In Fig. 3, the mechanical part (63.5 mm $\times$ 58.8 mm $\times$ 34.9 mm) surface is shiny, when projecting the laser on the shiny surface, very little diffuse light can be returned, especially in the high curvature area; consequently, there is a lack of laser scanned data in the those areas.

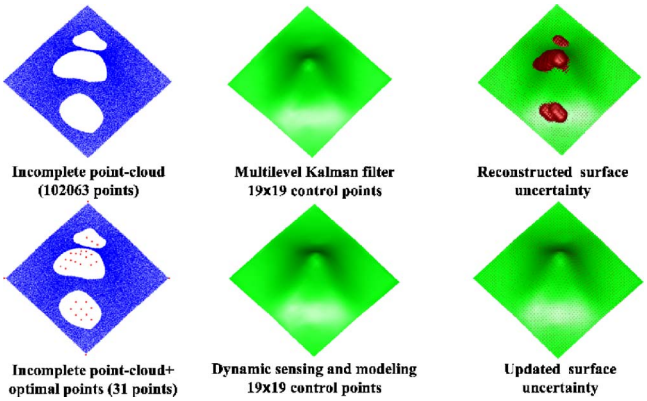


Fig. 15 Reconstructed surface and its uncertainty before and after dynamic update

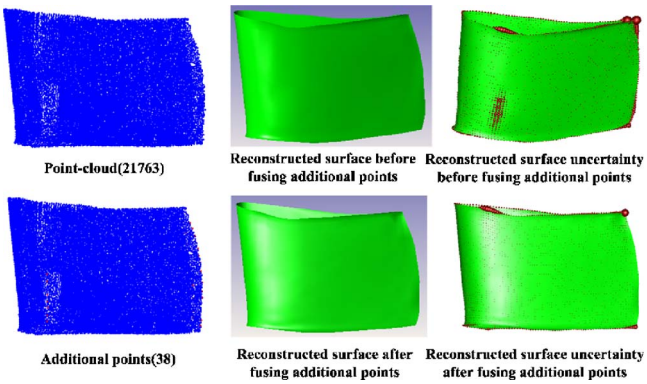


Fig. 16 Reconstructed surface and its uncertainty before and after dynamic update

As processed for the simulation surface in Fig. 15, the initial surface is first reconstructed with the multilevel Kalman filter from scanned point-cloud as the base surface, and the maximum uncertainty threshold in these two parts can be estimated by 0.000434 and 0.000885. The dynamic sensing-and-modeling procedure is then used to modeling the surface based on shape uncertainty as shown in Figs. 16 and 17.

6.2 Validation Criteria. To further validate the proposed dynamic sensing and modeling approach, we now compare the efficiency and final reconstructed surface accuracy. The measurement time and modeling surface accuracy computation criteria are given as follows.

Measurement Efficiency

Single sensor measurement time. In order to quantify the time saving of multisensor inspection, a two-criteria comparison is made between the multisensor inspection and the single tactile probe inspection since tactile sensing in CMM is still the prevalent coordinate measurement tool. The first criterion is that the

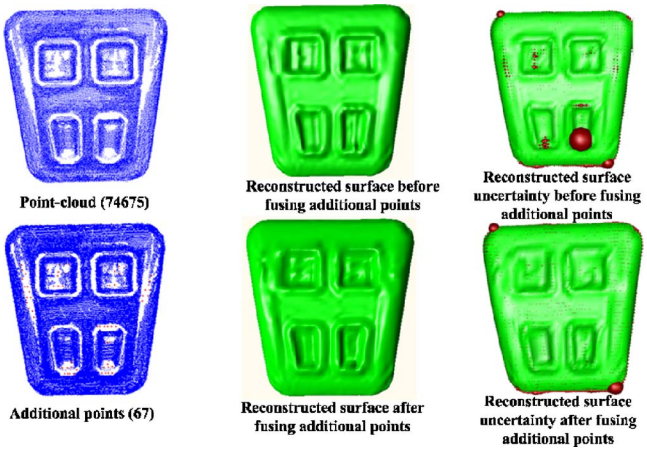


Fig. 17 Reconstructed surface and its uncertainty before and after dynamic update

Table 1 Measurement time comparison

Sensing method		Number of measured points	Estimated time (min)
Simulation surface	3D laser scanner and CMM	102,063+31	0.68
	CMM Criterion I	$19\times19\times10=3610$	60.1
	CMM Criterion II	$102,063/100+31=1052$	17.5
Shiny part	3D laser scanner and CMM	74,675+67	3.20
	CMM Criterion I	$35\times35\times10=12250$	204.166
	CMM Criterion II	$74,675/100+67=814$	13.06
Nozzle part (considering two surfaces)	3D laser scanner and CMM	$21,763\times2+38$	$2.117\times2+0.63=4.867$
	CMM Criterion I	$(35\times19\times10)\times2=13,300$	221.67
	CMM Criterion II	$21,763\times2/100+38=785$	7.888

number of measured points should be ten times the number of parameters in the model [13] (Criterion I). The second criterion is based on the fact the uncertainty is inversely proportional to the number of points N [35] and the surface uncertainty is proportional to the uncertainty of measurement (see the weighted least squares in Appendix A) (Criterion II), which is denoted by

$$\Lambda U \propto \frac{\Lambda z}{N} \tag{30}$$

With these two criteria, the time of single touch probe sensing to produce the required number of measurement points can be compared with the combined time of 3D laser scanning and dynamic CMM probing.

Measurement speed estimate. To estimate the time for the synthetic surface, we assume the laser 3D scanner scanning speed is 10,000 points per second [1], and one point per second [36] for CMM probe.

In the two actual parts measuring process, the actual scanning speed of the whole part is about 2.117 min for the Minolta scanner with six views for each part. To scan enough points for nozzle airfoil to reduce occlude area, an additional six views are used for scanner to scan the reversal airfoil part. Again, we assume touch probe senses the parts at the speed of one point per second.

Surface Accuracy. To validate the final surface quality, we use the root mean square (RMS) error distance to check the reconstructed surface f accuracy

$$\mathrm{RMS} = \sqrt{\frac{\sum_{ij=1}^{M,N} [d(p_{ij}^f, p_{ij}^g)]^2}{MN}} \tag{31}$$

where p_{ij}^f is the ijth uniformly sampled point on the reconstructed surface f, p_{ij}^g is the ijth projected point of p_{ij}^f to the nominal surface g, $d(p_{ij}^f, p_{ij}^g)$ is the distance between the point p_{ij}^f and its projected point p_{ij}^g on the surface g, and M and N are the numbers of sampled points along u,v directions on the parametric domain (here, $M=N=50$).

In practice, the actual surface may not be known. A large number of CMM probing points can be sampled as the reference points on the actual surface. The RMS can than be obtained by

$$\mathrm{RMS} = \sqrt{\frac{\sum_{i=1}^{N} [d(p_i^s, p_i^f)]^2}{N}} \tag{32}$$

where p_i^s is the sampled point using CMM probe, N is the number of those sampled points. p_i^f is the projected point to the reconstructed surface f, and $d(p_i^s, p_i^f)$ is the distance between p_i^s and p_i^f.

For the same data points, we can also employ least-squares to fit a surface and compare RMS error between the reconstructed surface and the nominal surface. This would give us a perspective of the surface accuracy using the Kalman filter and the simple least-squares method.

6.3 Comparison. Based on above criteria, we examine the performance of the dynamic sensing and modeling in terms of measurement time, algorithm efficiency, and final reconstructed surface accuracy.

Measurement Time Comparison for Sensors. The two criterions are used to compare the measurement speed using single tactile probe sensing and using the integration of the two sensors.

From Table 1 , we can see thatdynamic integration of the 3D laser scanner and CMM leads to much faster measurement than single tactile probe sensing, at least several times (some times 100 times) better.

Algorithm Efficiency Comparison. For the three examples, a comparison is also made between the proposed convex hull-based NBP computing approach and the traditional Fedorov exchange method on a computer (Dell™ Dimension DIM4700) with two Intel Pentium® IV 2.8 GHz processors and 512 MB RAM.

In Table 2, in average, the NBP computing takes more time for the Fedorov exchange to search 10,000 random points to find an "optimal" point that is comparable to the convex hull-based NBP computing method. We believe our geometric convex hull-based NBP method can be further optimized to achieve real-time speed,

Table 2 Computational time comparison of NBP algorithms

Part	Fedorov exchange (s)	Convex hull (s)
Simulation surface	4.35545	1.98181
Shiny part	6.92026	6.03249
Nozzle part	5.02865	4.38437

Table 3 Initial surface computing time with the Kalman filter in different modes

Part	Incremental updating mode (s)	Batch fitting mode (s)
Simulation surface (102,063 points)	15,472.75	76.703027
Shiny part(74,675 points)	11,641.68	92.548003
Nozzle part(21,763 points)	1076.72	24.826996

Table 4 Dynamic surface updating time with one single sensing point

Part	Reconstructing surface from scratch with least-squares (s)	Incremental updating with the Kalman filter (s)
Simulation surface	7.717999	0.003032
Shiny part	19.172001	0.025983
Nozzle part	5.702999	0.009895

which will be explored in our future work.

In addition, it is efficient to use the Kalman filter to estimate the initial surface from scanned point cloud in the batch fitting mode and to update the surface with one NBP point in the incremental updating mode.

As compared in Table 3, the initial surface with the Kalman filter in the batch fitting mode leads to much higher efficiency, especially when the number of scanned data points is very large.

Once the initial surface is obtained, it can be easily seen (Table 4) that the Kalman filter in the incrementally updating mode exhibits the significant efficiency in reconstructing surface with only one single sensing point, and can take place in real time.

Accuracy Comparison Between the Kalman Filter and Simple Least-Squares. The simple least-squares (weight is not considered in the fitting process) is a common surface reconstruction method. Here we use the simple least-squares method to validate the reconstructed surface accuracy of dynamic sensing-and-modeling approach.

From Fig. 18, it can be seen that

- *The quality (smoothness and RMS) of reconstructed surface has improved when additional points are added through either Kalman filter or least-squares method.*
- *The reconstructed surface through the Kalman filter is smoother and more accurate than that of a simple least-squares method.*

The two actual physical parts also show the same two characteristics. Due to the actual surface geometry is unknown, additional CMM probing points are sampled to compare the fitting accuracy.

As compared in Figs. 19 and 20, we can see that the reconstructed surface quality characterized by RMS error improves with additional dynamic CMM probing points. The resulting surface is more accurate than that with the simple least squares method.

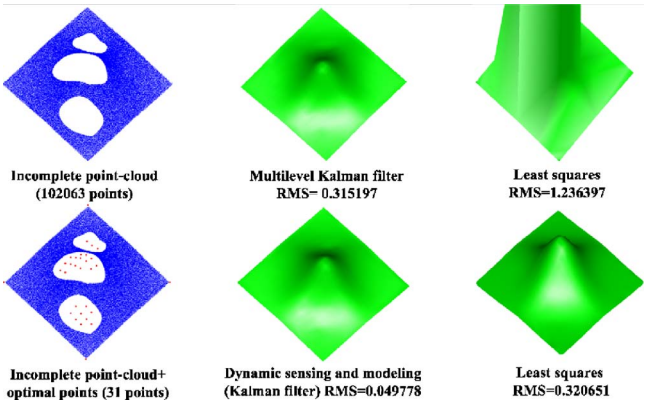


Fig. 18 Accuracy comparison between reconstructed surfaces from the Kalman filter and least-squares

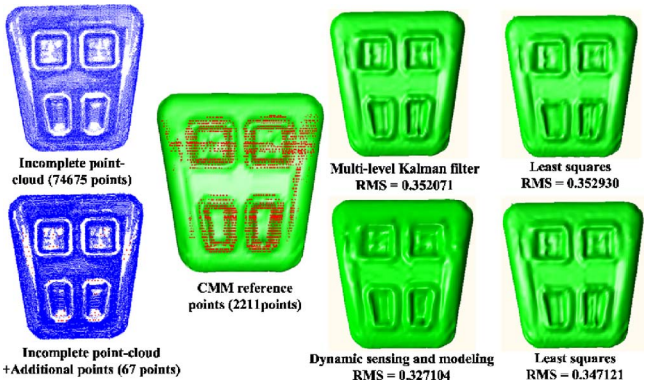


Fig. 19 Accuracy comparison between reconstructed surfaces from the Kalman filter and least-squares

Accuracy Comparison Between Dynamic Sensing-and-Modeling and Statically Pre-planned Sensing and Then Post-sensing Reconstruction Approach. Based on the initial surface and uncertainty estimated from scanned point cloud with area scanner, one way is to statically pre-plan all the measurement points' locations and then sense these points on the physical surface. The surface can them be reconstructed from the initial surface and these additional sensed points. Another way is to dynamically determine sensing locations and update the surface through the dynamic sensing-and-modeling approach as developed in this paper.

In Table 5, It can be seen that *the final surface from dynamic sensing-and-modeling is more accurate than that from statically pre-planned sensing and then post-sensing reconstruction method for the same number of additional probed points, even though the pre-planned points are determined by the exact NBP finding method—Convex hull-based optimal point determination method presented in this paper.*

In fact, due to the fact that the estimated surface cannot completely reflect the true shape in the missing data areas (see Fig. 5), the sensing locations may be away from the true optimal points on the physical surface, which tends to make the reconstructed surface deviate from the true surface. Unlike the statically pre-planned sensing and then post-sensing reconstruction method, dynamic sensing-and-modeling can facilitate the reconstructed surface gradually approximating the actual surface and provide a more accurate estimate of the sensing point, which again make the reconstructed surface approximate better to the physical surface.

7 Conclusion

This paper presents a dynamic sensing-and-modeling approach for integrating an area laser scanner with a tactile *point* probe to improve sensing speed and to reduce measurement uncertainty. The basic idea is to measure the object with area scanning first,

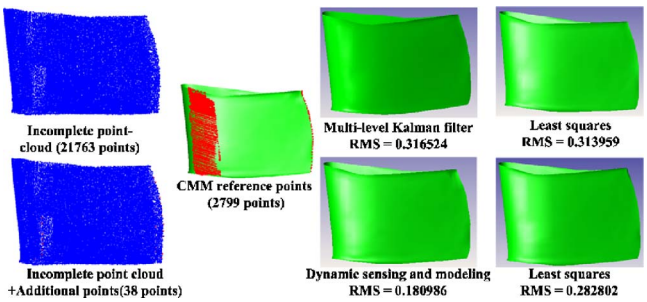


Fig. 20 Accuracy comparison between reconstructed surfaces from the Kalman filter and least-squares

Table 5 Statically pre-planned sensing and then post-sensing reconstruction versus dynamic sensing-and-modeling

Part	Optimal sensing points determining method		Number of additional measure points	RMS
Simulation surface	Statically pre-plan	Fedorov exchange	31	0.059472
		Convex hull-based method	31	0.055860
	Dynamic sensing and modeling		31	0.049778
Shiny part	Statically pre-plan	Fedorov exchange	67	0.342901
		Convex hull-based method	67	0.327779
	Dynamic sensing and modeling		67	0.327104
Nozzle part	Statically pre-plan	Fedorov exchange	38	0.212925
		Convex hull-based method	38	0.184523
	Dynamic sensing and modeling		38	0.180986

followed by point probing at the point of highest uncertainty in the reconstructed surface from scanned data. The surface is dynamically updated with the probed data and the next best probing point is determined based on the updated surface model. In this process, the surface uncertainty is dynamically computed to find the NBP until the shape uncertainty meets certain criteria.

The technical core of this dynamic sensing-and-modeling approach is the use of the Kalman filter for surface reconstruction and a novel efficient method to compute maximal uncertainty in the reconstructed surface. The Kalman filter is used to combine a large number of laser-scanned data and a smaller set of tactile points into one surface. It allows the incremental update of the surface model when additional sensed points are available, as opposed to reconstructing the surface from scratch. We solve the next-best-point problem by transforming the B-spline surface's uncertainty distribution into a higher dimensional uncertainty surface so that the convex hull property of the B-spline surface can be utilized to dramatically reduce the search speed and to guarantee the optimality of the resulting point.

Experimental study further demonstrates the following:

1. The dynamic integration of two sensors is more efficient than single sensor sensing in order to obtain the same quality surface.
2. The different fitting modes of the Kalman filter can be efficiently employed in initial surface estimate from large amount of scanned pointcloud and in the dynamic sensing-and-modeling procedure.
3. The reconstructed surface quality is improved due to the additional optimally and dynamically sensed points, and the reconstructed surface is more accurate than that of simple least-squares.
4. The reconstructed surface through dynamic sensing and modeling approximates better to the actual surface than that with statically pre-planned then post-sensing and reconstruction method.

Acknowledgment

This work was supported in part by the U.S. National Science Foundation Award No. 0529165.

Appendix A

In this appendix, the Kalman filter and the weighted least-squares method are compared in surface reconstruction.

Let n be the number of measured points, P_0 be the initial estimate, K_1 be the Kalman gain when fitting the point z_1 into the B-spline surface, and A_1 be B-spline shape function matrix.

The internal state estimate can be obtained by

$$P_1 = P_0 + K_1(z_1 - A_1 P_0) \tag{A1}$$

The equation can also be written as

$$P_1 = (I - K_1 A_1)P_0 + K_1 z_1 \tag{A2}$$

From Eq. (A2), we can further infer P_2 as

$$P_2 = (I - K_2 A_2)(I - K_1 A_1)P_0 + (I - K_2 A_2)K_1 z_1 + K_2 z_2 \tag{A3}$$

After all the measurements are fused, the end state estimate P_n can be denoted by

$$P_n = \prod_{i=1}^{n} (I - K_i A_i)P_0 + \prod_{i=2}^{n} (I - K_i A_i)K_1 z_1 + \cdots + K_n z_n \tag{A4}$$

From the uncertainty update equation, we can get

$$(I - K_r A_r)\Lambda P_{r-1} = \Lambda P_r \tag{A5}$$

and

$$\prod_{i=r}^{n} (I - K_r A_r)\Lambda P_{r-1} = \Lambda P_n \tag{A6}$$

Multiplying Eq. (A6) with the matrix $(\Lambda P_{r-1})^{-1}$, we can get

$$\prod_{i=r}^{n} (I - K_r A_r) = \Lambda P_n (\Lambda P_{r-1})^{-1} \tag{A7}$$

Substituting Eq. (A7) into Eq. (A4), we get the end state estimate P_n as

$$P_n = \Lambda P_n (\Lambda P_0)^{-1} P_0 + \Lambda P_n (\Lambda P_1)^{-1} K_1 z_1 + \cdots + \Lambda P_n (\Lambda P_n)^{-1} K_n z_n \tag{A8}$$

From the uncertainty updating Eq. (18) of the Kalman filter, we can obtain the state uncertainty covariance by

$$(\Lambda P_r)^{-1} = (\Lambda P_{r-1})^{-1} + A_r^T (\Lambda z_r)^{-1} A_r \tag{A9}$$

Multiplying the matrix $K_r z_r$, we can get

$$(\Lambda P_r)^{-1} K_r z_r = [(\Lambda P_{r-1})^{-1} + A_r^T (\Lambda z_r)^{-1} A_r]\Lambda P_{r-1} A_r^T (\Lambda z_r + A_r \Lambda P_{r-1} A_r^T)^{-1} z_r \tag{A10}$$

$$(\Lambda P_r)^{-1} K_r z_r = A_r^T (\Lambda z_r)^{-1} (\Lambda z_r + A_r \Lambda P_{r-1} A_r^T)(\Lambda z_r + A_r \Lambda P_{r-1} A_r^T)^{-1} z_r \tag{A11}$$

$$(\Lambda P_r)^{-1} K_r z_r = A_r^T (\Lambda z_r)^{-1} z_r \tag{A12}$$

Substituting Eq. (A12) into Eq. (A8), we get the end state estimate P_n as

$$P_n = \Lambda P_n \left[(\Lambda P_0)^{-1} P_0 + \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} z_i \right] \tag{A13}$$

From Eq. (A9), we can calculate the end state uncertainty by

$$(\Lambda P_n)^{-1} = (\Lambda P_0)^{-1} + \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} A_i \tag{A14}$$

Substituting Eq. (A14) into Eq. (A13), we can get the end state estimate P_n as

$$P_n = \left[(\Lambda P_0)^{-1} + \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} A_i \right]^{-1} \left[(\Lambda P_0)^{-1} P_0 + \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} z_i \right] \tag{A15}$$

From Eq. (A15), we can conclude that (1) the end state estimate of the Kalman filter is *independent of the fusion sequence of the point* z_i, *and* z_j and (2) the final state estimate of the Kalman filter can be obtained in the *batch processing mode* with Eq. (A15).

The weighted least-squares method is also often used to reconstruct the surface from the measured points in the statistical optimal sense. The optimal object function $f(P_n)$ can be defined as

$$\min f(P_n) = \sum_{i=1}^{n} \frac{(z_i - A_i P_n)^2}{\Lambda z_i} \tag{A16}$$

Differentiating $f(P_n)$ against P_n, we can get

$$\frac{\partial f}{\partial P_n} = \sum_{i=1}^{n} \frac{A_i^T (z_i - A_i P_n)}{\Lambda z_i} = \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} z_i - \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} A_i P_n \tag{A17}$$

Letting $\partial f / \partial P_n = 0$, the optimal value can be calculated by

$$P_n = \left(\sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} A_i \right)^{-1} \sum_{i=1}^{n} A_i^T (\Lambda z_i)^{-1} z_i \tag{A18}$$

If all the measure points have the same uncertainty $\Lambda z_i = \Lambda z_j$, $i \neq j$, the end estimate of surface control points P_n has the same equation of general least-squares

$$P_n = \left(\sum_{i=1}^{n} A_i^T A_i \right)^{-1} \sum_{i=1}^{n} A_i^T z_i = (A^T A)^{-1} A Z \tag{A19}$$

where A is the matrix form of A_i, $i = 1, \ldots, n$, Z is the matrix form of z_i, $i = 1, \ldots n$. A and Z can be written as

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_n \end{bmatrix} \quad Z = \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix} \tag{A20}$$

Comparing the end estimate of the Kalman filter (see Eq. (A15)) and the weighted least-squares (see Eq. (A18)), we can see that the Kalman filter equals to the weighted least-squares when the initial estimate $P_0 = 0$ and its uncertainty $|\Lambda P_0| \to \infty$ or when P_0 and ΛP_0 are computed directly from measured points using the weighted least-squares method.

Appendix B

LEMMA. *Let* C_{seg1} *and* C_{seg2} *be the two B-spline curve segments with uncertainty* $\Lambda P_{C_{\text{seg1}}}$ *and* $\Lambda P_{C_{\text{seg2}}}$. *If* $\max_{ij}(\Lambda P_{C_{\text{seg1}}})_{ij} \leqslant \min_{ij}(\Lambda P_{C_{\text{seg2}}})_{ij}$, *the uncertainty of any point on the curve segment* C_{seg1} *is no greater than that of any point on the curve segment* C_{seg2}.

Proof. Assume Λp_1 is the uncertainty of any one point p_1 on C_{seg1}, and Λp_2 is the uncertainty of any one point p_2 on C_{seg2}. According to the convex hull property of B-spline curve uncertainty, we can get

$$\min_{ij}(\Lambda P_{C_{\text{seg1}}})_{ij} \leqslant \Lambda p_1 \leqslant \max_{ij}(\Lambda P_{C_{\text{seg1}}})_{ij} \tag{B1}$$

$$\min_{ij}(\Lambda P_{C_{\text{seg2}}})_{ij} \leqslant \Lambda p_2 \leqslant \max_{ij}(\Lambda P_{C_{\text{seg2}}})_{ij} \tag{B2}$$

Since $\max_{ij}(\Lambda P_{C_{\text{seg1}}})_{ij} \leqslant \min_{ij}(\Lambda P_{C_{\text{seg2}}})_{ij}$, then

$$\Lambda p_1 \leqslant \Lambda p_2 \tag{B3}$$

Hence, the uncertainty of any point on the curve segment C_{seg1} is no greater than that of any point on the curve segment C_{seg2}.

References

[1] NVision's model maker3D scanner provides worldwide savings for NISSAN, See http://www.nvision3d.com/CaseStudies/Nissan.html
[2] Metris solutions, http://www.metris.com/en/html/applications/applications.htm
[3] Qian, X., and Harding, K. G., 2003, "Partitioning Positional and Normal Space for Fast Occlusion Detection," ASME Design Engineering Technical Conferences, Chicago, IL, September.
[4] Tarabanis, K. A., Allen, P. K., and Tasi, R. Y., 1995, "A Survey of Sensor Planning in Computer Vision," IEEE Trans. Rob. Autom., **(11)**(1), pp. 86–104.
[5] Maver, J., and Bajcsy, R., 1993, "Occlusions as a Guide for Planning the Next View," IEEE Trans. Pattern Anal. Mach. Intell., **15**(5): 417–433.
[6] Pito, R, and Bajcsy, R, 1995, "A Solution to the Next Best View Problem for Automated CAD Model Acquisition of Free-Form Objects Using Range Cameras," Proc. SPIE, **2598**, pp. 78–89.
[7] Qian, X., and Harding, K. G., 2003, "A Computational Approach for Optimal Sensor Setup," J. Soc. Photo-Opt. Instrum. Eng., **42**(5), pp. 1238–1248.
[8] Whaite, P., and Ferrie, F. P., 1997, "Autonomous Exploration: Driven by Uncertainty," IEEE Trans. Pattern Anal. Mach. Intell., **19**(3), pp. 193–205.
[9] Li, Y. F., and Liu, Z. G., 2003, "Uncertainty-Driven Viewpoint Planning for 3D Object Measurements," *Proceedings of the 2003 IEEE International Conference on Robotics and Automation, Taipei, Taiwan*, September 14–19.
[10] Elmaraghy, H., and Gu, P., 1987, "Expert System for Inspection Planning," CIRP Ann., **36**, pp. 85–89.
[11] Spyridi, A. J., 1994, "Automatic Generation of High Level Inspection Plans for Coordinate Measuring Machines," Ph.D. Dissertation, University of Southern California.
[12] Vafaeesefat, A., and Elmaraghy, H., 2000, "Automated Accessibility Analysis and Measurement Clustering for CMMs," Int. J. Prod. Res., **38**, pp. 2215–2231.
[13] Kurfess, T. R., and Banks, D. L., 1995, "Statistical Verification of Conformance to Geometric Tolerance," CAD, **27**(5), pp. 353–361.
[14] Menq, C., Yau, H., and Lai, G., 1992, "An Intelligent Planning Environment for Automated Dimensional Inspection Using Coordinate Measuring Machines," ASME J. Eng. Ind., **114**, pp. 222–230.
[15] Li, Y. F., and Liu, Z. G., 2003, "Method for Determining the Probing Points for Efficient Measurement and Reconstruction of Freeform Surfaces," Meas. Sci. Technol., **12**, pp. 1280–1288.
[16] Elkott, D. F., Elmaraghy, H. A., and Elmaraghy, W. H., 2002, "Automatic Sampling for CMM Inspection Planning of Free-Form Surfaces," Int. J. Prod. Res., **40**(11), pp. 2653–2676.
[17] Chen, L. C., and Lin, G. C., 1997, "A Vision-Aided Reverse Engineering Approach to Reconstructing Free-Form Surfaces," Rob. Comput.-Integr. Manufact., **13**(4), pp. 323–336.
[18] Motavalli, S., Suharitdamrong, V., and Alradhdan, A., 1998, "Design Model Generation for Reverse Engineering Using Multisensors," IIE Trans., **30**, pp. 357–366.
[19] Nashman, M., 1993, "Vision and Touch Sensors for Dimensional Inspection," Manuf. Rev., **6**(2), pp. 155–162.
[20] Shen, T. S., Huang, J., and Menq, C. H., 2000, "Multiple-Sensor Integration for Rapid and High-Precision Coordinate Metrology," IEEE/ASME Trans. Mechatron., **5**(2), pp. 110–121.
[21] Shen, T., Huang, J., and Menq, C. H., 2001, "Multiple Sensor Planning and Information Integration for Automated Coordinate Metrology," ASME J. Comput. Inf. Sci. Eng., **1**(2), pp. 167–179.
[22] Szeliski, R., 1991, "Probabilistic Modeling of Surfaces," Proc. SPIE, **1570**, pp. 154–165.
[23] Wang, Y. F., and Wang, J. F., 1994, "On 3D Model Construction by Fusing Heterogeneous Sensor Data," CVGIP: Graph. Models Image Process., **60**(2), pp. 210–229.
[24] Wang, Y. F., 1991, "A New Method for Sensor Data Fusion in Machine Vision," Proc. SPIE, **1570**, pp. 31–42.
[25] Van Pabst, J. V. L., and Kreke, P. F. C., 1993, "Multi-Sensor Data Fusion of Points, Line Segments and Surface Segments in 3D Space," Proc. SPIE, **2059**, pp. 190–201.
[26] Huang, Y., and Qian, X., 2006, "A Stochastic Approach to Surface Reconstruction," *Proceedings of IDETC/CIE 2006, ASME 2006 International Design Engineering Technical Conference & Computers and Information in Engineering Conference, Philadelphia, PA, September 10–13*.
[27] Zhang, Z., 1995, "Parameter Estimation Techniques: A Tutorial with Application to Conic Fitting," Research Report No. 2676, INRIA.
[28] Kalman, R. E., 1960, "A New Approach to Linear Filtering and Prediction Problems," ASME J. Basic Eng., **82**, pp. 35–45.
[29] Welch, G., and Bishop, G., 2004, "An Introduction to the Kalman Filter,"

http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf
[30] Thacher, N. A., and Lacey, A. J., 1996, "Tutorial: The Kalman Filter," http://www.tina-vision.net/docs/memos/1996-002.pdf
[31] Maybeck, P., 1979, *Stochastic Models, Estimation, and Control*, Academic Press, Inc, New York, Vol. 1.
[32] Wang, M. Y., and Pelinescu, D., 2001, "Optimizing Fixture Layout in a Point Set Domain," IEEE Trans. Rob. Autom., **17**(3), pp. 312–323.
[33] Piegl, L., and Tiller, W. 1995, *The NURBS Book*, Springer, New York.
[34] Ridler, T. W., and Calvard, S., 1978, "Picture Thresholding Using an Iterative Selection Method," IEEE Trans. Syst. Man Cybern., **8**, pp. 630–632.
[35] Yau, H. T., 1998, "Uncertainty Analysis in Geometric Best Fit," Int. J. Mach. Tools Manuf., **38**, pp. 1323–1342.
[36] Li, Y. D., and Gu, P. H., 2004, "Free-Form Surface Inspection Techniques State of the Art Review," CAD, **36**(13), pp. 1395–1417.