The International Journal of Advanced Manufacturing Technology (2018) 97:3595–3606
https://doi.org/10.1007/s00170-018-2203-7

ORIGINAL ARTICLE

Fast registration of 3D point clouds with offset surfaces in precision grinding of free-form surfaces

Shanyong Chen[1,2] · Chuanchao Wu[1,2] · Shuai Xue[1,2] · Zhengjian Li[1,2]

Received: 2 January 2018 / Accepted: 14 May 2018 / Published online: 28 May 2018

Abstract
Because of its high sensitivity to misalignment, precision grinding of free-form surfaces with micron accuracy requires accurate registration of the surface measurement point cloud. Registration of point clouds obtained with a coordinate measuring machine (CMM) is generally an iterative process of finding optimal coordinate transformation between the CMM frame and the model frame of the workpiece by minimizing the point-to-surface distances with probe radius compensation. For free-form surfaces, frequent calculation of point-to-surface distances consumes very much time, and a trade-off has to be made between the efficiency and the accuracy. This paper presents a method for fast registration of free-form surface point clouds based on the point-to-triangle distance which involves only Delaunay triangulation of a two-dimensional dataset, and the surface normal is quickly calculated from cross product. Probe radius compensation is realized by registering the probe center points with the offset surface. We prove that it is equivalent to registering the probe contact points with the nominal surface through theoretical analysis. The registration problem is then formulated as sequential linear least-square problems with properly defined ball constraints. To validate the method, numerical simulations are presented to show the accuracy of the point-to-triangle distance. The registration algorithm also shows excellent robustness against misalignment of tens of millimeters/degrees. Finally measurement, registration, and grinding of a free-form optical surface are experimentally demonstrated. The surface error obtained after registration is used for compensatory grinding which reduces it to micron level.

Keywords Free-form surface · Probe radius compensation · Surface registration · Coordinate measurement · Precision grinding

1 Introduction

Free-form surfaces find more and more applications in precision and even ultra-precision systems. A remarkable change for optical free-form surfaces is the transition from illumination systems to high-performance imaging systems working in infrared, visible, or even ultraviolet spectrum [1, 2]. As a result, tolerance on the surface error gets much tighter. It typically requires compensatory grinding of the surface with micron accuracy before entering the lapping and polishing stage, based on measurement of the profile or surface error [3–5]. To get the surface error, the surface is sampled with a coordinate measuring machine (CMM) which gives a three-dimensional (3D) point cloud representing the real surface. And then, the point cloud is registered and compared with the nominal surface. The normal deviations of measurement points are used to assess the surface error.

Because of its high sensitivity to misalignment, registration of free-form surface point clouds is indispensable for error assessment and corrective machining. It is generally an iterative process of finding optimal coordinate transformation between the CMM frame and the model frame of the workpiece by minimizing the point-to-surface distances. This follows the well-established iterative closest point (ICP) algorithm [6]. However, special care must be taken to two major problems in terms of high accuracy and efficiency. One is calculation of the point-to-surface distance in the surface normal direction. It is equivalent to finding the closest point on the surface. The other problem is the probe radius compensation. The CMM readout is basically 3D coordinates of the probe center when touching the test surface. Physically, the trajectory of the

✉ Shanyong Chen
mesychen@163.com

[1] College of Mechatronic Engineering and Automation, National University of Defense Technology, 47 Yanzheng Street, Changsha 410073, Hunan, China

[2] Hunan Key Laboratory of Ultra-precision Machining Technology, 47 Yanzheng Street, Changsha, Hunan, China

probe center lies on the offset surface generated by offsetting the nominal surface in normal directions with an amount of probe radius. Therefore, the probe contact points need to be determined before calculating their deviations from the nominal surface. Obviously, surface normal calculation is the common basis of the above two problems. In reverse engineering practice, nominal description of the test surface is unavailable, and the surface normal is usually estimated through linear [7–9] or quadratic [10] approximation using some local points, or through spline fitting [11] to the measurement profile. It demands sufficiently dense sampling to suppress the bias of normal vector estimation. When there is the prior knowledge of the nominal surface description, the surface normal is also well-defined. However, for free-form surfaces, there is no closed-form solution, and numerical algorithms are usually adopted to solve a system of nonlinear equations. It consumes very much time, and a trade-off has to be made between the efficiency and the accuracy. Usually, similar approximations as in reverse engineering are preferred to reduce the computation time but with reduced accuracy at the same time. In strongly curved surface measurement, the residual error of approximation may introduce considerable change of normal vectors and significant error of point-to-surface distance.

Probe radius compensation strongly relies on the surface normal. A recent review of probe radius compensation methods is presented by Kawalec and Magdziak [12]. To avoid use of incorrect surface normal, the probe center points instead of the contact points were also proposed to be registered with the offset surface. The offset surface is obtained by converting the nominal bi-cubic Bezier surface into bi-cubic Coons surface [13]. Ainsworth et al. [14] obtained the offset surface by non-uniform rational B-spline (NURBS) fitting to the point clouds generated from the nominal surface with normal offset of probe radius. This indirect way of radius compensation does not need to calculate the probe contact points but still consumes too much time for surface fitting. To further improve the efficiency, Xiong and Li [15] proposed to simply subtract the probe radius when calculating the point-to-surface distance in surface registration or workpiece localization. The probe center points are still registered with the nominal surface instead of the offset one, but the contribution of probe radius is subtracted from the normal deviations. Since the probe center points are ideally deviated from the nominal surface by an amount of probe radius (a couple of millimeters), there is a risk of finding incorrect closest points especially when the surface is strongly curved with significant variation of curvatures. To deal with the problem of incorrect surface normal, a stylus tip envelop method was proposed for radius compensation at surface discontinuities or small local features [16, 17]. However, it demands high-density sampling points to ensure that the successive probe tips overlap partially with each other.

The objective of this study is to develop a new mathematical model for registration of free-form surface point clouds with high efficiency and high accuracy. The nominal surface is defined by high-density point cloud which makes the simple point-to-triangle distance sufficiently accurate. In addition, the probe center point cloud is registered with the offset surface point cloud to avoid use of potentially incorrect closest points. Note the point-to-point distance is not preferred for registration of two sets of discrete points despite the availability of numerous algorithms based on data structure and computational geometry, e.g., the k-D tree reducing the computational complexity to $O(N_p \log N_x)$ where N_p and N_x are the point number of two sets [18]. Because the sampling grid of one set probably "slides" away from that of the other set, incorrect correspondence of the closest point can induce considerable error compared with the point-to-surface distance.

This paper is organized as follows. In the next section, the mathematical model is built for calculating the point-to-triangle distance approximating the real point-to-surface distance. The registration problem with offset surface is then formulated following the ICP algorithm with additional ball constraints imposed on the sequential linear least-squares problems (LSPs). Theoretical analysis is presented to show the equivalence to registering probe contact points with the nominal surface. Section 3 gives simulation results validating the accuracy of point-to-triangle distances and also robustness of the registration algorithm against misalignment. In Section 4, we experimentally demonstrate the use of the proposed method for compensatory grinding of a free-form optical surface. The surface error is finally reduced to micron level. We then reach a conclusion in Section 5.

2 Registration model for 3D point clouds with offset surfaces

The nominal surface represented by either analytical equation or discrete data cloud is described in the model frame {M} while the measurement point cloud is described in the CMM frame {W}. The two coordinate frames are generally not coincident and, therefore, have to be aligned before we can assess the surface error. Registration is the process of finding optimal coordinate transformation between the CMM frame and the model frame by minimizing the point-to-surface distances usually in the LS sense. The distances are calculated for measurement points transformed into the model frame by the optimal coordinate transformation.

2.1 Point-to-triangle distance

When the nominal surface is described by high-density point cloud $\{S_i | i = 1,2,\ldots,N\}$, it is sufficiently accurate to approximate it by Delaunay triangulation. Then, the nearest triangle is

found, and the point-to-surface distance is well-approximated by the point-to-triangle distance. However, triangulation of a high-density 3D data cloud consumes still too much time and memory. Our solution is to project the measurement points $\{P_j | j = 1,2,\ldots,n\}$ as well as the nominal surface points onto a certain plane, e.g., the XOY plane. The projections $\{T_i\}$ of $\{S_i\}$ are two-dimensional (2D) data set and quickly triangulated in the XOY plane. It is also fast to search for the enclosing Delaunay triangles for projections $\{Q_j\}$ of $\{P_j\}$. The vertices of enclosing triangle are then back-projected onto the nominal surface as illustrated in Fig. 1(a).

It is easy to calculate the normal distance of the measurement point P_j to the plane determined by the triangle $\Delta S_1 S_2 S_3$. As shown in Fig. 1(b), the unit normal vector $\boldsymbol{n}_j$ of the plane is given by the cross product of two edge vectors of the triangle:

$$\boldsymbol{n}_j = \frac{\overrightarrow{S_2S_1} \times \overrightarrow{S_2S_3}}{\left\| \overrightarrow{S_2S_1} \times \overrightarrow{S_2S_3} \right\|} \tag{1}$$

The point-to-triangle distance d_j is then obtained from the inner product of the unit normal vector and the vector determined by the measurement point and one of the vertices, e.g., S_2:

$$d_j = \left\langle \overrightarrow{S_2P_j}, \boldsymbol{n}_j \right\rangle \tag{2}$$

The closest point on the nominal surface is hence approximated by the pedal point C_j on the triangle plane.

The above process involves only Delaunay triangulation of a 2D dataset, and the surface normal is quickly calculated from cross product. It is far more efficient than the conventional algorithm which requires tessellation of 3D dataset and searching for the nearest triangle in 3D space.

2.2 Probe radius compensation using offset surfaces

The CMM basically records the 3D coordinates of the probe center or probe location P_l instead of the probe contact point P_c. The trajectory of the probe center lies in theory on the offset surface, and the offset distance is right at the probe radius which is typically a couple of millimeters. Surface error is evaluated by normal deviations of the real surface from the nominal one. That naturally requires to find the probe contact points which are also the closest points on the real surface corresponding to the probe center points. As suggested by Xiong and Li [15], we can simply subtract the probe radius from the calculated point-to-surface distance when registering the probe center points with the nominal surface. But there is a risk of finding incorrect closest points because the probe center points are ideally deviated by the probe radius. This problem gets much more serious when we use the point-to-triangle distance by projecting all points onto a coordinate plane. The millimeter-scale offset of the probe center is big enough to change the enclosing triangle, which then results in considerable error of normal distance. As shown in Fig. 2, the projections of probe center P_l and probe contact P_c lie in different triangles. This difference exists even for an ideal surface without any surface error. That means the subsequent iterations of surface registration will probably fail to find the right coordinate transformation between the CMM frame and the model frame.

However, the above problem can be avoided by registering the probe center points with the offset surface. It allows that initial deviation exists between the two coordinate frames which may also result in wrong triangle correspondence. Owing to the convergence of the algorithm, the registration process can still find the optimal transformation which leads to all-zero point-to-surface distances for an ideal surface.

When measuring a surface represented by a high-density point cloud $\{S_i = (x_i, y_i, z_i)\}$, the offset surface can also be represented by points $\{S_{ei} = (x_{ei}, y_{ei}, z_{ei})\}$ normally deviated from the nominal surface. Such a point cloud of offset surface is generated offline and keeps the sampling density. The surface normals are hence estimated by fitting the nominal point cloud to high-order surfaces, e.g., bi-cubic fit to ensure high accuracy. The offset point cloud is then obtained by normally shifting the nominal points with an amount of probe radius r:

$$S_{ei} = S_i + r \cdot \boldsymbol{n}_i \tag{3}$$

After generation of the offset point cloud, the measurement points, i.e., the probe center points, are then registered with the offset surface. To show that such an indirect registration is equivalent to registration with the nominal surface, we consider the 2D case as illustrated in Fig. 3. The measurement point is P_l lying on the offset surface of the real surface with machining error. When registering with the nominal surface, the contact point P_c is found. And the point-to-surface distance is calculated as below:

$$d_c = \left\langle \overrightarrow{CP_c}, \boldsymbol{n}_c \right\rangle \tag{4}$$

where $\boldsymbol{n}_c$ is the unit normal vector at the closest point C on the nominal surface. While registering with the offset surface, the point-to-surface distance is calculated as below:

$$d_l = \left\langle \overrightarrow{H_eP_l}, \boldsymbol{n}_h \right\rangle \tag{5}$$

where $\boldsymbol{n}_h$ is the unit normal vector at the closest point H_e on the offset surface. The difference of the two kinds of distances is given by the following equation when the surface error is small enough:

$$d_l - d_c \approx r(\cos\theta - 1) \approx -r\frac{\theta^2}{2} \tag{6}$$

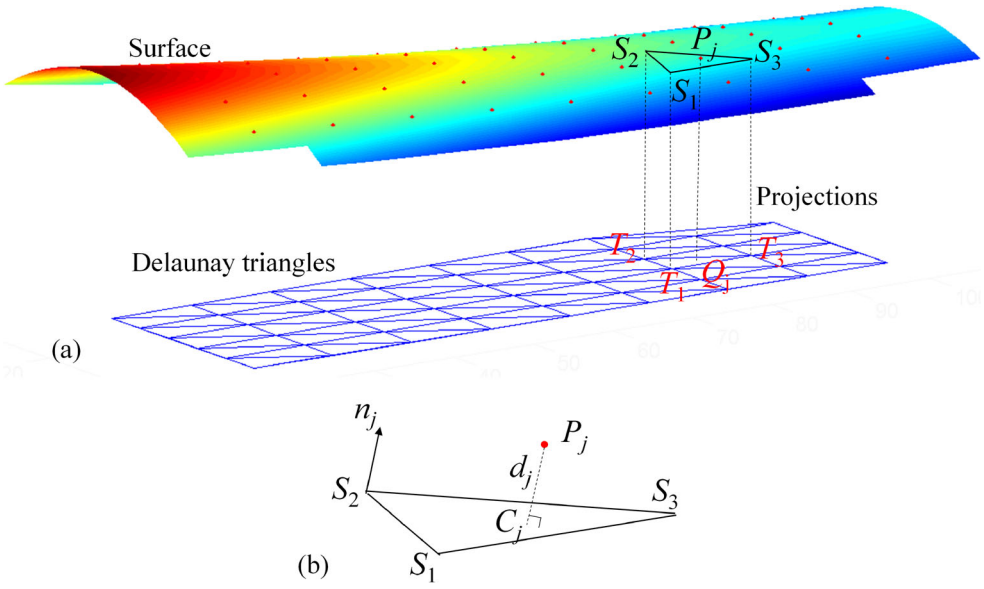


Fig. 1 Schematic diagram of point-to-triangle calculation. (a) Finding the triangle. (b) Calculating the point-to-triangle distance

The approximation is valid as the small surface error indicates infinitesimal angle θ between the nominal surface normal and the real surface normal. This angle is approximated by the surface height change (micron level) divided by the lateral spacing (millimeter scale). Figure 3 is an exaggerated schematic diagram clearly showing the geometrical variation induced by the real surface height. The lateral shift of the closest point from C to H on the nominal surface is first order infinitesimal o(θ) while the vertical change is second order infinitesimal o(θ^2). And the variation of normals at these two points is also second-order infinitesimal. Therefore, the difference described in Eq. (6) is second-order infinitesimal and can be neglected. That is to say, registering the probe center points with the offset surface will potentially introduce only second-order error.

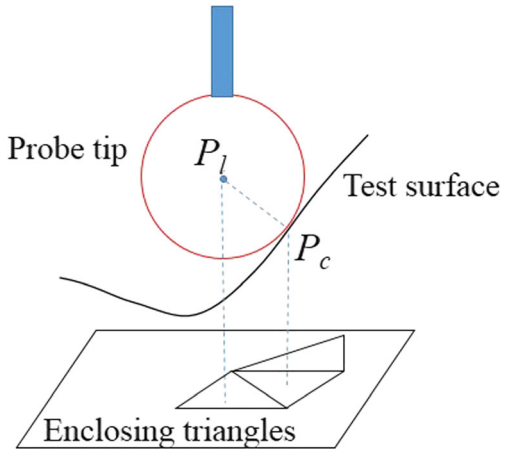


Fig. 2 The probe center offset results in wrong triangle correspondence

2.3 Registration algorithm with ball constraints

Suppose the measurement points, i.e., the probe centers are denoted by $\{Y_j = (x_j, y_j, z_j, 1) | \ j = 1,2,\ldots,n\}$ in homogeneous coordinates in the CMM frame {W}. Now, we have also generated the offset surface points denoted by $\{X_{ei} = (x_{ei}, y_{ei}, z_{ei}, 1) | \ i = 1,2,\ldots,N\}$ in the model frame {M}. Registration problem is then formulated as the nonlinear LSP.

$$\min f(g, X_e) = \sum_{j=1}^{n} d_j^2 = \sum_{j=1}^{n} \left\langle g^{-1} Y_j - X_{ej}, n_j \right\rangle^2 \tag{7}$$

where g is the coordinate transformation between the model frame {M} and the CMM frame {W}, d_j is the point-to-triangle distance defined in Eq. (2), X_{ej} and n_j are the closest point on and unit normal vector of the triangle plane corresponding to the measurement point, respectively.

The coordinate transformation g transforms coordinates in the model frame to those in the CMM frame. It can be represented by an element of the special Euclidean group SE(3), which has the following canonical representation using the exponential map [19]:

$$\mathrm{SE}(3) = \left\{ \exp\left(\sum_{k=1}^{6} m_k \hat{\xi}_k \right) \right\} \tag{8}$$

where

$$\hat{\xi} = \begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix} \tag{9}$$

is referred to as a twist and $\xi = [v, \omega]^{\mathrm{T}}$ as the twist coordinate. $\xi_k \in \mathrm{R}^6$ is a unit vector whose kth element is 1 and otherwise 0

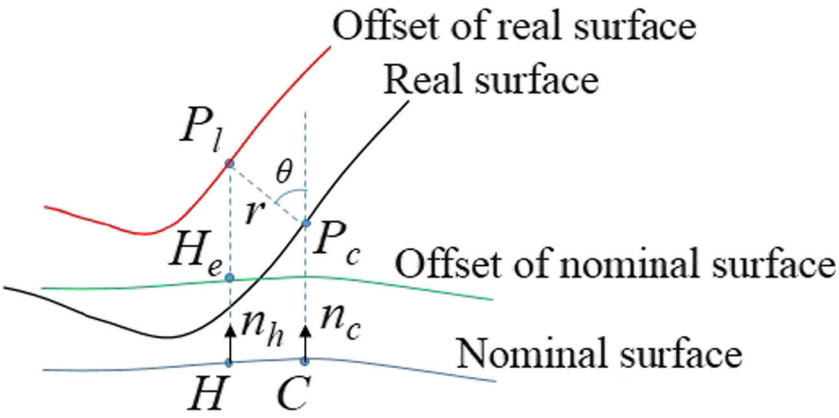


Fig. 3 Second-order error induced by registering the probe center with the offset surface

[20, 21]. The optimization variables m_k represent the six degrees of freedom of a rigid body in 3D space.

The objective function $f(g)$ is a nonlinear function of transformation parameters m_k. The nonlinear LSP described in Eq. (7) is essentially an optimization problem involving two (sets of) variables: transformation parameters m_k and closest points X_e or normal vectors n. Algorithms for this kind of problem are usually classified as simultaneous optimization (SO, e.g., Newton-type algorithms) and alternating optimization (AO, also called alternating variable method). The basic idea of AO is optimizing the individual (set of) variable(s) separately, with the other (set of) variable(s) fixed. Repeat the individual optimization alternatingly, and the iterations will finally converge to a solution of the LSP. Rigorous proof of the convergence of AO-type algorithms is difficult. Therefore, to solve the registration problem, two subproblems are solved alternatingly and iteratively. The first one is calculating the point-to-triangle distance with given coordinate transformation. The other one is optimizing the coordinate transformation with determined closest points and surface normals. The advantage of this AO-type algorithm is that the subproblems can be easily solved. As we mentioned in Section 2.1, the point-to-triangle distance can be calculated very fast based on 2D computational geometry. For the optimization of coordinate transformation, the following linearization of exponential function is used:

$$g^{l+1} = g^l \exp\left(\sum_{k=1}^{6} m_k \hat{\xi}_k\right) \approx g^l \left[I + \sum_{k=1}^{6} m_k \hat{\xi}_k\right] \tag{10}$$

where l is the sequence number of iteration, and I is the identity matrix. Then, the optimization is rewritten as a linearized LSP in the lth iteration which can be solved with well-established algorithms such as singular-value decomposition (SVD):

$$\min f(g, X_e) \approx \sum_{j=1}^{n} \left\langle \left(I - \sum_{k=1}^{6} m_k \hat{\xi}_k\right) \left(g^l\right)^{-1} Y_j - X_{ej}^l, n_j^l \right\rangle^2 \tag{11}$$

It is equivalent to finding the LS solution to the linear equation $\mathbf{A}m = \mathbf{b}$, where the matrix $\mathbf{A}$ and vector $\mathbf{b}$ are defined below:

$$\mathbf{A} = (a_{j,k}) = \left(\left(n_j^l\right)^T \hat{\xi}_k X_{ej}^l\right) = \left(\left(n_j^l\right)^T \hat{\xi}_k \left(g^l\right)^{-1} Y_j\right) \tag{12}$$

$$\mathbf{b} = (b_j) = \left(\left(n_j^l\right)^T \left(\left(g^l\right)^{-1} Y_j - X_{ej}^l\right)\right) \tag{13}$$

Therefore, the iterations comprise a series of point-to-triangle distance calculation problems and sequential linear LSPs. The program starts with an initial guess of coordinate transformation g^0, calculates the point-to-triangle distance, and then finds and updates the coordinate transformation g^1 by minimizing the objective function. The two subproblems are iteratively solved in such an alternating way until the program converges to an acceptable tolerance. Readers are referred to ref. [20] for a proof of the local convergence. An important observation is that it is helpful to yield a sequence of $\{g^l, X_e^l\}$, which guarantees the monotone decreasing property of the objective function, i.e.,

$$f\left(g^l, X_e^{l-1}\right) \le f\left(g^l, X_e^l\right) \le f\left(g^{l+1}, X_e^l\right) \tag{14}$$

However, care must be taken when solving the sequential linear LSPs since they are only valid in a very small local region due to the linearization. In other words, solutions to the linear LSP are not always minima of the original transformation optimization subproblem. It may result in an increasing objective, which leads to incorrect convergence or divergence. Therefore, we propose to impose a ball constraint on the vector m of transformation parameters, and this yields a ball-constrained linear LSP [21, 22]:

$$\begin{aligned} &\mathbf{A}m = \mathbf{b} \\ &\text{s.t.} \|m\| \le \alpha \end{aligned} \tag{15}$$

where the positive scalar α is a ball constraint ensuring the local validity of linearization. And "s.t." stands for "subject to." Ball-constrained LSP is typical by constraining the optimization variables in a ball of certain radius, i.e., the norm of

the vector m is equal to or less than the magnitude of the ball radius α.

Algorithms for the ball-constrained LSP are referred to ref. [23]. SVD is still applicable but with additional computation to solve a non-linear equation. The ball constraint can be determined from prior knowledge or changes adaptively with the converging performance. Generally, it starts with quite a big value and then decreases as the iteration number increases, in view of the fact that AO-type algorithms converge very fast in the first few iterations but then quickly slow down. In practice, when detecting an increasing objective function value, the algorithm should back off to the last iteration, and the ball constraint is reduced by a half or even by an order.

The whole algorithm is summarized as below.

Input: measurement points (probe centers) $\{Y_j = (x_j, y_j, z_j,1)| j = 1,2,\ldots,n\}$ in homogeneous coordinates; nominal surface points $\{X_i = (x_i, y_i, z_i,1)| i = 1,2,\ldots,N\}$ in homogeneous coordinates; initial values of the ball constraint α and coordinate transformation g^0; terminating condition ε.

Preliminary: generate the offset surface points $\{X_{ei} = (x_{ei}, y_{ei}, z_{ei},1)| i = 1,2,\ldots,N\}$ keeping high density according to Eq. (3).

Output: surface error described with point-to-triangle distances $\{d_j\}$ defined in Eqs. (2) and (7); optimal coordinate transformation g^*.

Step 0: (a) Set $l = 0$, compute the point-to-triangle distance $\{d_j^0\}$ with an initial transformation g^0 and yields the closest points $\{X_{ej}^0\}$;

(b) Compute the objective function f^0 defined in Eq. (7).

Step 1: (a) Solve the ball-constrained linear LSP defined in Eq. (15) and yields the new transformation parameters m^{l+1};

(b) Update the transformation g^{l+1} with parameters m^{l+1};

(c) Compute the distance $\{d_j^{l+1}\}$ with transformation g^{l+1} and updates the closest points $\{X_{ej}^{l+1}\}$;

(d) Compute the objective function f^{l+1} defined in Eq. (7);

(e) If $(1 - f^{l+1}/f^l) > \varepsilon$, then set $l = l + 1$ and return to step 1(a); otherwise, exit and report the results.

Be careful that the approximation at the right side of Eq. (10) should not be used for updating the transformation g^{l+1} from g^l. The left side should be used instead.

3 Numerical simulations

A free-form surface defined by a high-density point cloud is used for simulation and experimental verifications. The nominal surface point cloud is shown in Fig. 4 along with the offset surface point cloud generated from Eq. (3) with the surface normals estimated by bi-cubic fit to ensure high accuracy. The offset distance is the probe radius $r = 1.5$ mm. The number of points is $N = 451{,}401$.

Figure 5a, b shows the surface slope in degree and the Gaussian curvature of the surface, respectively. It can be seen that the surface shape changes with remarkable variation of slope and curvature. The slope s is defined as the root of sum square of partial derivatives in X and Y directions:

$$s = \sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2} \tag{16}$$

3.1 Validity of point-to-triangle distances

The point-to-triangle distance is an approximation of the point-to-surface distance on condition that the surface is represented by a high-density point cloud. It is not intuitional to determine how dense the point cloud should be. But we can check it in a straightforward way.

The surface point cloud is first resampled on a grid of 1021×1241 points with spacing of 0.1 and 0.05 mm in X and Y directions, respectively. The test surface is then generated with some random error added in the surface normal direction. The surface error as the nominal point-to-surface distance has a Gaussian height distribution function and Gaussian autocovariance functions in both X and Y directions. The correlation length is 30 mm, and the peak-to-valley (PV) surface error is 6.657 μm, as shown in Fig. 6a. It simulates the real surface with machining error.

The point-to-triangle distance is then calculated and compared with the simulated surface error. The residual error indicates that the difference between the point-to-triangle distance and the nominal point-to-surface distance is sufficiently small, as shown in Fig. 6b. The PV difference is 5 nm, which is smaller than 1/1000 surface error.

The surface curvature plays a very important role in such a projection-based algorithm. However, its influence is mainly focused on calculation of the point-to-triangle distance since we still follow the ICP algorithm. Inherently, it relates the calculated distance to the sampling density of the surface since the point-to-triangle distance is linear approximation of the real point-to-surface distance. More sampling points are required at the local area of smaller radius of curvature for a given tolerance, which is general and similar to the well-known adaptive sampling strategy for free-form surface based on the curvature change. It is currently difficult for us to give a rigorous analysis on the influence. But we recommend estimating the accuracy of distance calculation in advance since we have been given the nominal surface point cloud and the sampling density.

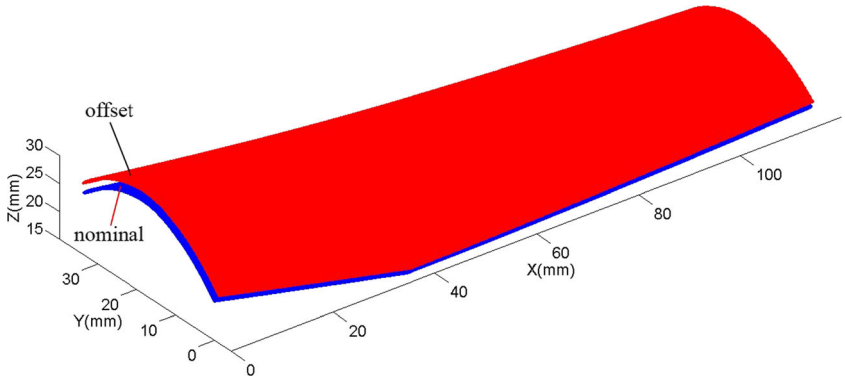


Fig. 4 Nominal surface (bottom level) and offset surface point cloud (top level)

The location of surface in relation to the plane of projection also has influence on the distance calculation. A trivial case is the hyper hemisphere which is a surface portion on a sphere with a solid angle exceeding 2π. Projections on the equator plane of both points on the southern hemisphere and on the northern one coincide with each other. Hence, the projection-based calculation of distance fails but we can use the latitude and longitude coordinates instead [24]. For general free-form surfaces, we can also choose an appropriate projection plane, e.g., the curvilinear coordinate plane of parameters (u, v) for tensor product surfaces. It indicates that for an extremely complex surface, we may need to divide it into several patches which are then measured and registered in different orientations.

3.2 Robustness of registration against misalignment

Readers may worry about the potential wrong triangle correspondence shown in Fig. 2 when the measurement point is deviated from the offset surface by a rigid body transformation. Usually, the CMM measurement frame is deviated from the model frame, and the transformation or misalignment is to be recognized by registration. The following simulations show that the registration algorithm can tolerate considerably large amount of misalignment.

To simulate the deviated measurement points, the nominal surface point cloud is undersampled (16× spacing) and then applied with different misalignment. These deviated points are registered with the nominal surface to check whether we can obtain zero surface error. The initial coordinate transformation g^0 is the identity matrix I since we have no prior knowledge of the real misalignment. Figure 7 shows the initial measurement points deviated from the nominal surface by shifting in X, Y, and Z with 20, 10, and 20 mm, respectively, and rotating around X, Y, and Z with 20°, 10°, and 20°, respectively. The measurement points are significantly deviated but the registration algorithm succeeds to bring them best matching the nominal surface.

Figure 8 is a semilog plot showing the iteration sequence of registration with different misalignment. All final objective values are confirmed to be very close to zero (mm^2), though different number of iterations is required with different choice of the ball constraint. The PV values of surface error obtained with the optimal coordinate transformation are all smaller than 10^{-6} nm, which is definitely negligible. It shows that the registration algorithm has excellent robustness against misalignment of tens of millimeters/degrees, which is an important advance compared with the ± 1 mm and ± 1° tolerance of the method proposed by Ainsworth et al. [14]. The robustness can be ascribed to the performance of AO-type algorithms and ball constraints imposed to ensure the convergence. Although the free-form surface is generally very sensitive to

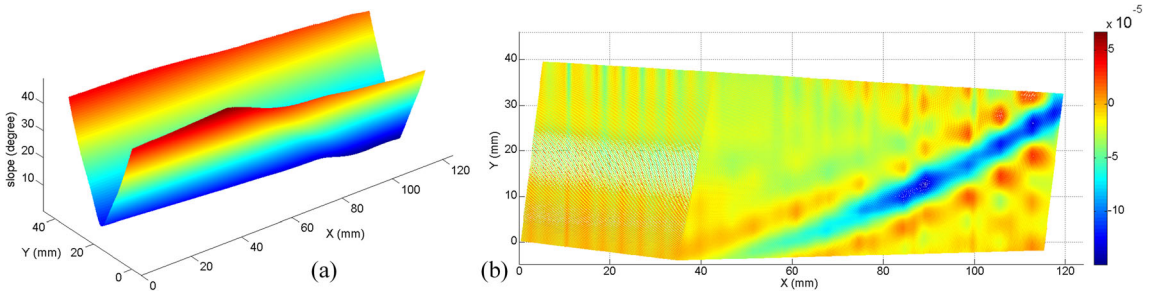
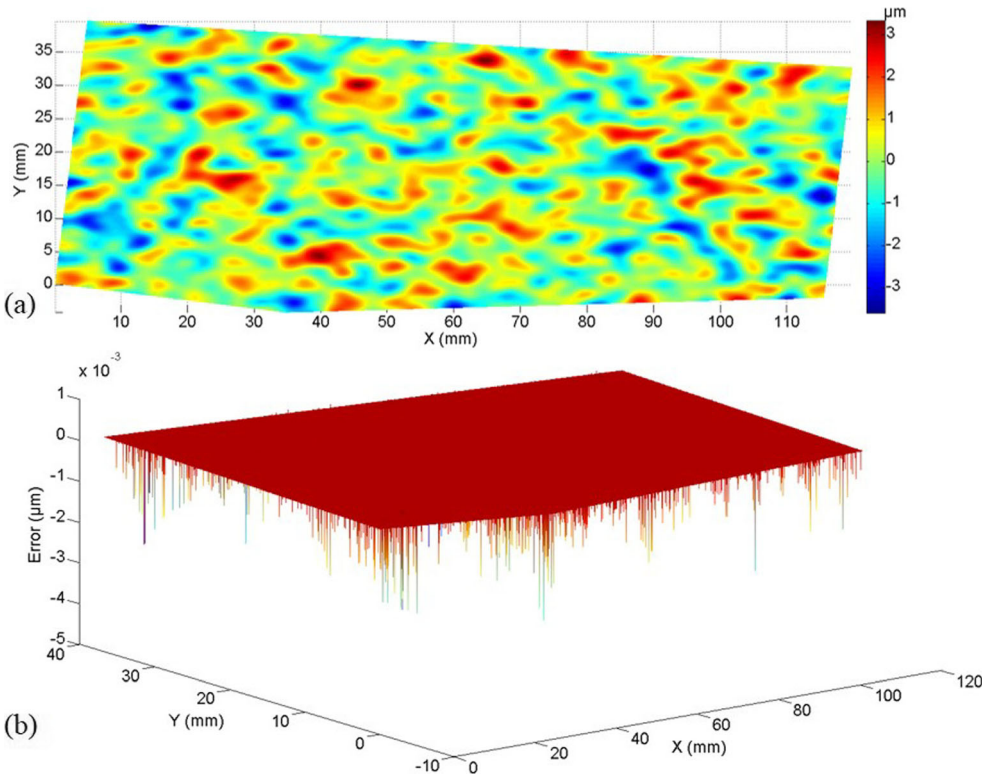


Fig. 5 Basic differential geometry of the surface. **a** Surface slope. **b** Gaussian curvature



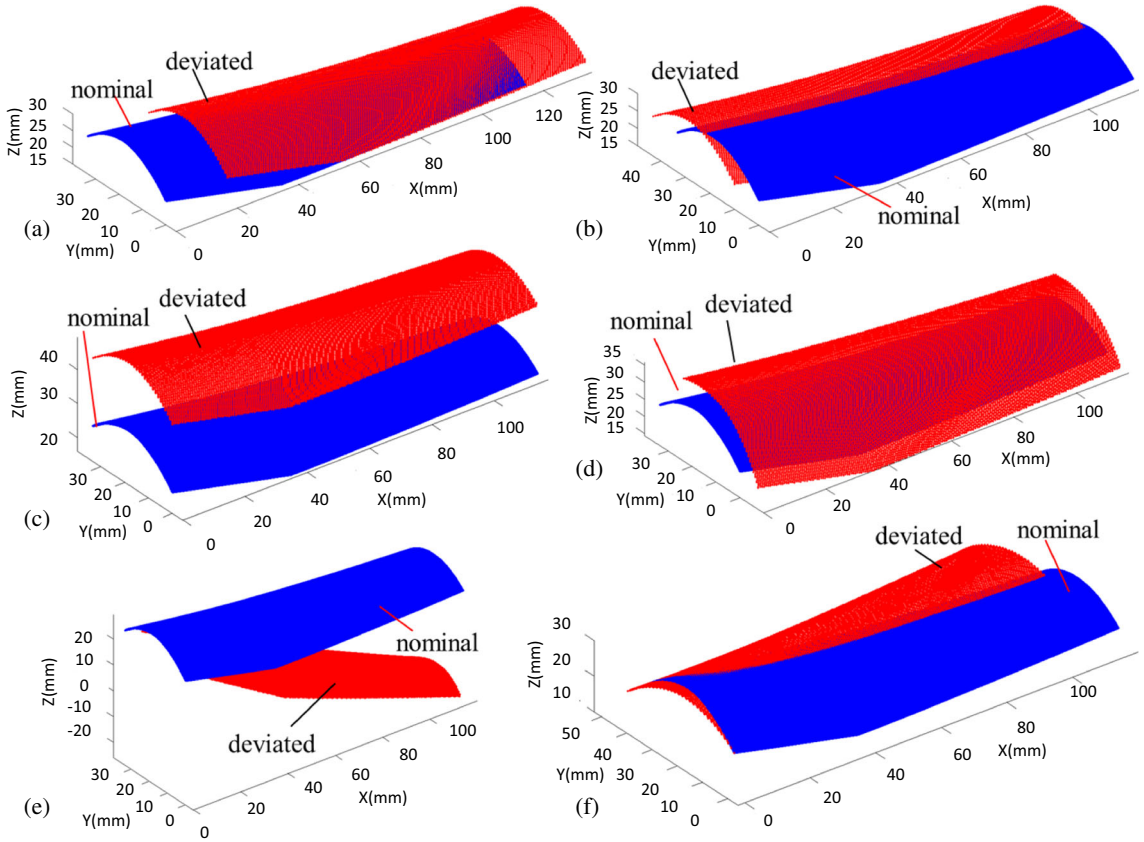


Fig. 6 Validity of point-to-triangle distances. **a** Nominal surface error. **b** Residual of point-to-triangle distances

Fig. 7 Measurement points deviated from the nominal surface. **a** Shift in *X* with 20 mm. **b** Shift in *Y* with 10 mm. **c** Shift in *Z* with 20 mm. **d** Rotation around *X* with 20°. **e** Rotation around *Y* with 20°. **f** Rotation around *Z* with 10°

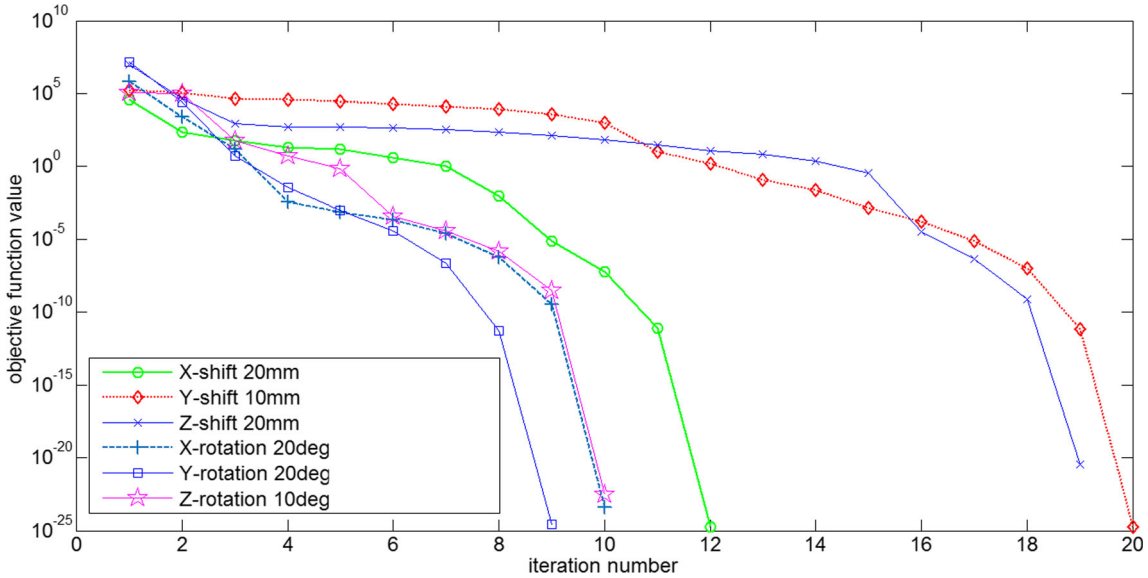


Fig. 8 Iteration sequence of registration with different misalignment

misalignment, the registration algorithm facilitates the alignment of the workpiece on CMM to a great extent.

4 Experimental verifications

Submicron accuracy is required for optical free-form surfaces applied in short-wave imaging systems. It is typically achieved by finishing processes such as lapping and polishing. However, the surface has to be shaped to micron accuracy during the grinding process, which is very important to accelerate the subsequent finishing process [25]. To realize high precision grinding, the surface after grinding is measured and registered to obtain the surface error without the influence of coordinate frame misalignment. The surface error is then used as a feedback for corrective grinding based on error compensation of the geometric error and wear of the wheel [26]. The iterative machining process includes repeated grinding, measurement, and registration, as shown in Fig. 9. Note again the measurement points are actually probe centers and should

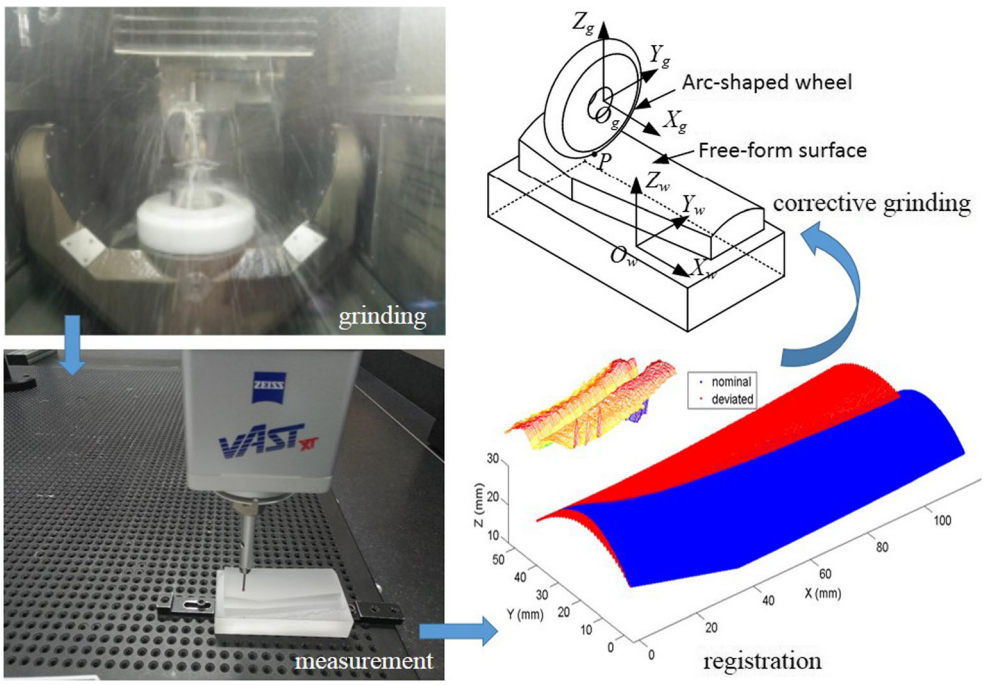


Fig. 9 Corrective machining with repeated grinding, measurement, and registration

be registered with the offset surface. The cubic fit of surface normal is utilized to generate the offset surface points as shown in Fig. 4.

The optical surface was ground with the Opto Tech MCG 250 and then measured with the Zeiss Accura CMM. Because the workpiece model frame is well-established on the CMM with the help of datum, a good guess of initial transformation is available. Figure 10 shows the measurement points applied with the initial guess of coordinate transformation which brings them close to the nominal surface point cloud.

The surface error after the first run of machining is shown in Fig. 11a. After compensatory grinding, the surface error was significantly reduced from PV 27.3 to 6.9 μm. And the root-mean-square (RMS) error is 1.0 μm as shown in Fig. 11b. The surface error reduction indirectly validates the registration algorithm. Two iterations are enough for the registration to reduce the objective function value from 3.2383 to 0.0566. The optimal coordinate transformation is obtained as below:

$$g^* = \begin{bmatrix} 0.99999935 & 0.00016112 & 0.00112800 & -0.06885510 \\ -0.00017015 & 0.99996794 & 0.00800538 & 0.06576194 \\ -0.00112668 & -0.00800557 & 0.99996732 & 0.19546377 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{17}$$

where the upper left 3 × 3 block is the rotation matrix, and the upper right 3 × 1 vector is the translation.

To further check the equivalence of registration with offset surfaces to that with nominal surface experimentally, the measurement points after registration are used to generate the estimated probe contact points by adding the probe radius in the minus normal direction. The surface normal vectors are also given by the registration algorithm as defined in Eq. (1), corresponding to the point-to-triangle distance with optimal coordinate transformation applied. The estimated probe contact points are then registered with the nominal surface. As expected, they are already best aligned with the nominal surface, and the optimal transformation is very close to identity matrix:

$$g = \begin{bmatrix} 1 & -0.00000023 & 0.00000007 & 0.00003690 \\ 0.00000023 & 1 & 0.00000234 & 0.00000294 \\ -0.00000007 & -0.00000234 & 1 & 0.00003522 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{18}$$

It shows that registration of probe center points with the offset surface is equivalent to registration of probe contact points with the nominal surface. Because we do not know exactly the surface normal at the measurement point especially in the case of misalignment, we prefer to register the probe centers with the offset surface which can be exactly generated in advance. Figure 12a shows the result of registration of probe contact points with the nominal surface. It is almost identical with Fig. 10(b).

Finally, as another comparison, we also try the registration with probe radius compensation suggested by Xiong and Li [15]. The measurement points are registered with the nominal surface but the probe radius is subtracted when calculating the point-to-surface distance. As we pointed out in Section 2.2, there is a risk of finding incorrect closest points. Hence, the surface error obtained is incorrect as shown in Fig. 12b.

5 Conclusion

The point-to-triangle distance is proposed to approximate the point-to-surface distance when the nominal surface is defined by a high-density point cloud. It involves tessellation and searching for the nearest triangle in 2D space instead of 3D space, which reduces the computation time and memory significantly. To avoid finding incorrect correspondence of the closest point, measurement points are registered with the offset surface instead of the nominal surface. Equivalence to registration of probe contact points with the nominal surface is then proved through both theoretical analysis and

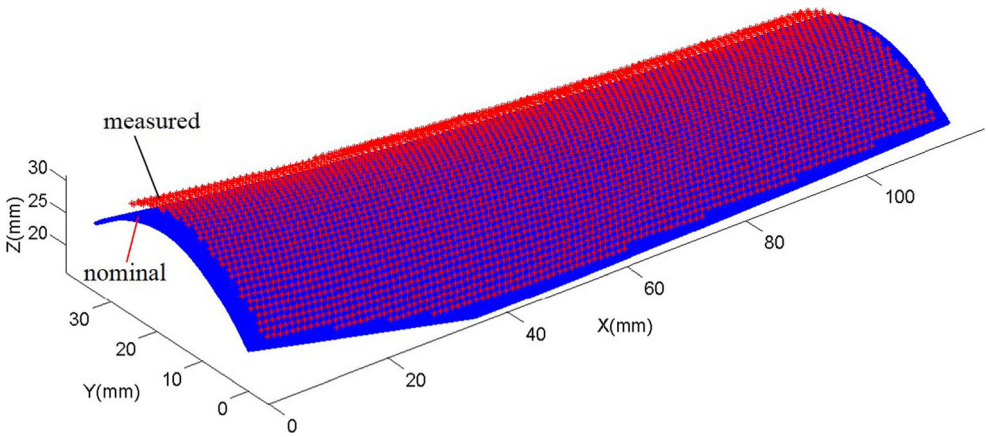


Fig. 10 Measurement points (top level with asterisk marks) applied with initial guess of coordinate transformation

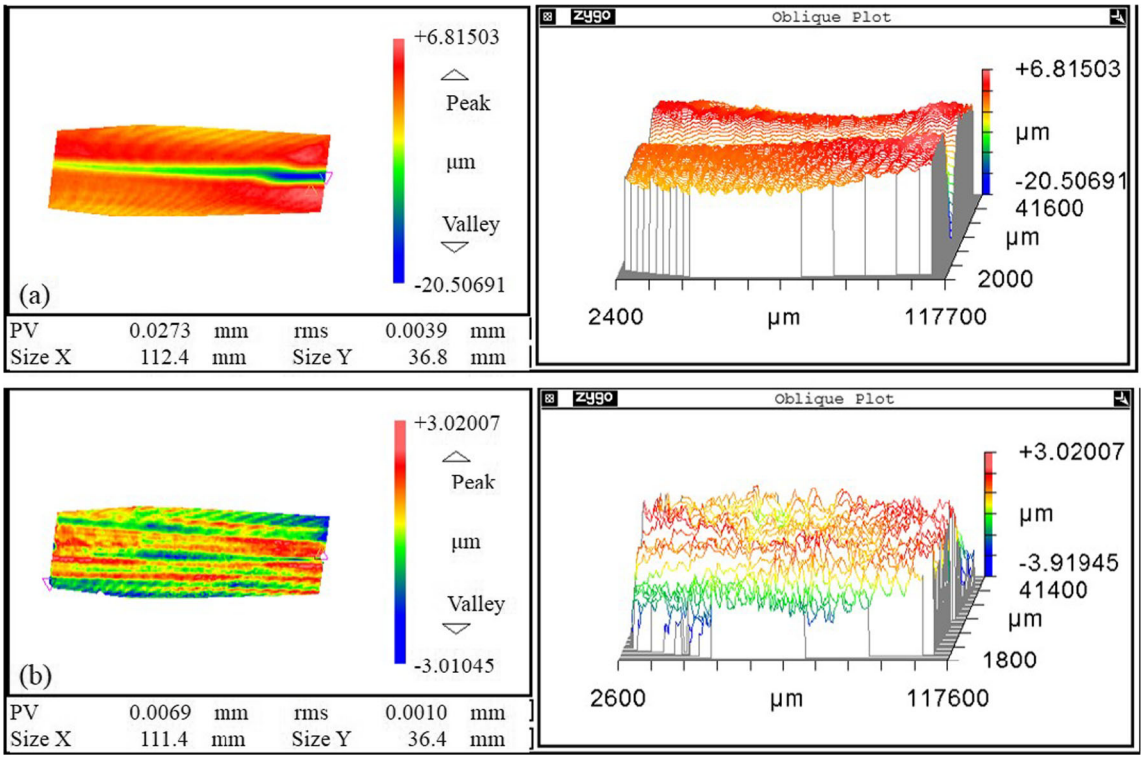


Fig. 11 Surface error reduction. **a** Before corrective machining. **b** After corrective machining

experimental verification. Simulations are also presented to show that the point-to-triangle distance can achieve nanometer accuracy for micron surface error. And the proposed registration algorithm possesses the advantage of excellent robustness

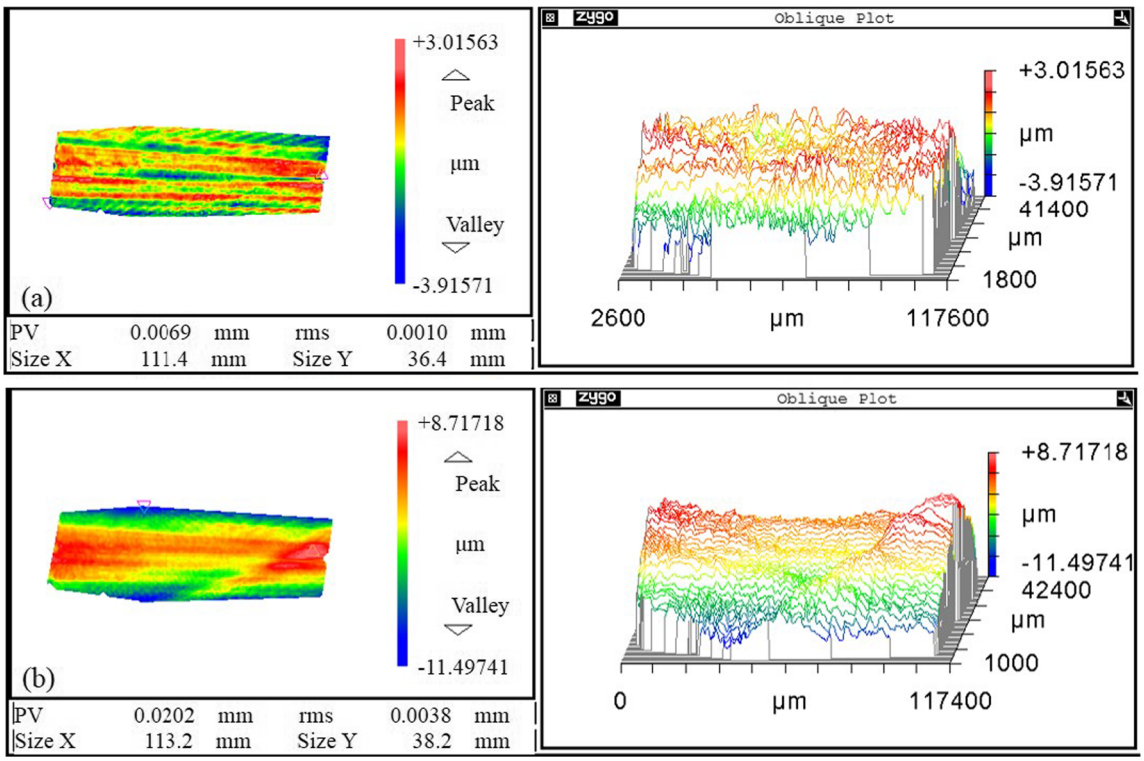


Fig. 12 Surface error comparison. **a** Registration of estimated probe contact points with the nominal surface gives identical result. **b** Registration of probe center points with the nominal surface by subtracting probe radius gives incorrect result

against misalignment of tens of millimeters/degrees, which is an important advance compared with current algorithms. It was successfully applied to grinding practice, providing an accurate surface error feedback to corrective machining which reduces the surface error to micron level.

Acknowledgements This project is supported by Science Challenge Program of China (TZ2018006) and Hunan Provincial Natural Science Foundation of China (2016JJ1003).

References

1. Fang FZ, Zhang XD, Weckenmann A, Zhang GX, Evans C (2013) Manufacturing and measurement of freeform optics. Ann CIRP 62: 823–846
2. Yang T, Jin GF, Zhu J (2017) Automated design of freeform imaging systems. Light: Sci Appl 6:1–10
3. Peng Y, Dai Y, Song C, Shi F (2016) Tool deflection model and profile error control in helix path contour grinding. Int J Mach Tools Manuf 111:1–8
4. Dai Y, Chen S, Kang N, Li S (2010) Error calculation for corrective machining with allowance requirements. Int J Adv Manuf Technol 49:635–641
5. Wang X, Xian J, Yang Y, Zhang Y, Fu X, Kang M (2017) Use of coordinate measuring machine to measure circular aperture complex optical surface. Measurement 100:1–6
6. Besl PJ, McKay HD (1992) A method for registration of 3-D shapes. IEEE T Pattern Anal 14:239–256
7. Suh SH, Lee SK, Lee JJ (1996) Compensating probe radius in free surface modeling with CMM: simulation and experiment. Int J Prod Res 34:507–523
8. Lin YC, Sun WI (2003) Probe radius compensated by the multi-cross product method in freeform surface measurement with touch trigger probe CMM. Int J Adv Manuf Technol 21:902–909
9. Ravishankar S, Dutt HNV, Gurumoorthy B (2010) Automated inspection of aircraft parts using a modified ICP algorithm. Int J Adv Manuf Technol 46:227–236
10. Song CK, Kim SW (1997) Reverse engineering: autonomous digitization of free-formed surfaces on a CNC coordinate measuring machine. Int J Mach Tools Manuf 37:1041–1051
11. Chiang YM, Chen FL (1999) Sculptured surface reconstruction from CMM measurement data by a software iterative approach. Int J Prod Res 37:1679–1695
12. Kawalec A, Magdziak M (2017) The selection of radius correction method in the case of coordinate measurements applicable for turbine blades. Precis Eng 49:243–252
13. Kim KI, Kim K (1996) A new measuring strategy for sculptured surfaces using offset surfaces. ASME J Manuf Sci Eng 118:646–651
14. Ainsworth I, Ristic M, Brujic D (2000) CAD-based measurement path planning for free-form shapes using contact probes. Int J Adv Manuf Technol 16:23–31
15. Xiong Z, Li Z (2003) Probe radius compensation of workpiece localization. ASME J Manuf Sci Eng 125:100–104
16. Woźniak A, Mayer JRR, Bałaziński M (2009) Stylus tip envelop method: corrected measured point determination in high definition coordinate metrology. Int J Adv Manuf Technol 42:505–514
17. Wozniak A, Mayer JRR (2012) A robust method for probe tip radius correction in coordinate metrology. Meas Sci Technol 23: 1–8
18. Bentley JL (1975) Multidimensional binary search trees used for associative searching. Commun ACM 18:509–517
19. Murray R, Li ZX, and Sastry SS (1994) A mathematical introduction to robotic manipulation. CRC Press, New York
20. Li ZX, Gou JB, Chu YX (1998) Geometric algorithm for workpiece localization. IEEE Trans Rob Autom 14:864–878
21. Chen S, Li S, Dai Y (2005) Iterative algorithm for subaperture stitching interferometry for general surfaces. J Opt Soc Am A 22: 1929–1936
22. Chen S, Xue S, Dai Y, Li S (2015) Subaperture stitching test of large steep convex spheres. Opt Express 23:29047–29058
23. Gloub GH, Van Loan CF (1996) Matrix computations, 3rd. Edition. The Johns Hopkins University, London
24. Chen S, Liao W, Dai Y, Li S (2012) Self-calibrated subaperture stitching test of hyper-hemispheres using latitude and longitude coordinates. Appl Opt 51(17):3817–3825
25. Li S, Dai Y (2017) Large and middle-scale aperture aspheric surfaces: lapping, polishing and measurement. John Wiley & Sons, Inc., Beijing
26. Peng Y, Dai Y, Song C, Chen S (2015) Error analysis and compensation of line contact spherical grinding with cup-shaped wheel. Int J Adv Manuf Technol 83:293–299