Phase I Monitoring of Spatial Surface Data from 3D Printing

Yangyang Zang[1] and Peihua Qiu[2]

[1]Department of Industrial Engineering, Tsinghua University

[2]Department of Biostatistics, University of Florida

In recent years, 3D printing gets more and more popular in manufacturing industries. Quality control of 3D printing products thus becomes an important research problem. However, this problem is challenging due to the facts that (i) the surface of a product from 3D printing can have arbitrary shape, even when the 3D printing process is in-control, (ii) surface observations of the product obtained from a laser scanner may not have regularly spaced locations, and (iii) the overall geometric positions of 3D printing products might be all different, making proper comparison among different products difficult. In this paper, we propose a Phase I control chart for monitoring products from 3D printing that addresses all these challenges. Numerical studies show that it works well in practice.

Keywords: Kernel smoothing; Nonparametric regression; Profile monitoring; Registration; Statistical process control; Surface estimation.

1 Introduction

Statistical process control (SPC) charts have been used widely in manufacturing industries, for monitoring production processes so that they work stably and satisfactorily (Qiu, 2014). The ultimate goal of SPC charts is to distinguish common cause variation, mainly due to noise, from special cause variation, due to systematic problems of the related production process, such as defective raw materials, improper operation of the workers, improperly adjusted machines, and so forth. When a special cause variation is detected, a signal should be given and the production process should be stopped immediately for engineers to find the root causes.

Traditional control charts, including the Shewhart, cumulative sum (CUSUM), exponentially weighted moving average (EWMA) and change-point detection (CPD) charts, are for monitoring one or more quality variables (Shewhart, 1931; Page, 1954; Roberts, 1959; Hawkins et al., 2003). Recently, monitoring of the relationship between two or more variables, called profile monitoring,

finds more and more applications. Early profile monitoring methods are based on linear or parametric modeling (e.g., Jin and Shi, 1999; Kang and Albin, 2000; Kim et al., 2003). More recent methods are more flexible without the parametric assumptions (e.g., Zou et al., 2008; Qiu et al., 2010).

This paper is motivated by the following profile monitoring problem. Figure 1 shows a 3D printer and a product. The top of the product is a curvy surface, while other sides are all flat. For this product, we are mainly concerned about the shape of its top surface. To measure the top surface, the product can be held firmly by a device, and then a laser scanner can scan the surface from its top. Although the scanner usually scans the product from the left side to the right side and from the top to the bottom, the observation locations are not regularly spaced in columns and rows. Instead, they are usually irregularly spaced. Also, different products may not be geometrically matched up well, in the sense that relative positions between the products and the laser scanner could be all different, and the products may have different thickness (i.e., the height from the bottom to the top surface), orientation, etc., although such difference is usually quite small. Therefore, the current profile (i.e., the top surface of the product) monitoring problem has at least the following three challenges: (i) the top surface has a complicated shape and it is difficult to describe using a parametric model, even when the 3D printing process is in-control (IC), (ii) the surface observation locations obtained by a laser scanner are not regularly spaced, and (iii) the overall geometric positions of 3D printing products might be all different, making proper comparison among different products difficult.

Figure 1: A 3D printer (left) and a product (right).

In the literature, we have not found any existing methods for handling the specific profile

monitoring problem described above. One related but different problem was discussed in Colosimo et al. (2014), where cylindrical surfaces were considered and modeled by a Gaussian process model with a constant mean function, homogeneous covariance function and a joint normal distribution of the observations. See also related discussions in Xia et al. (2008); del Castillo et al. (2013). But, the monitoring problem considered there is totally different from the current problem, described below. 1) The true surfaces considered there are cylindrical and can be described by a radial model. Therefore, the difference between the actual radius and the nominal radius can be modeled, and it might be reasonable to assume that the mean difference is a constant (cf., expression (7) in Colosimo et al. (2014)). In the current problem, the surfaces are rather arbitrary and do not have a constant mean. 2) Geometric alignment of different observed surfaces is critically important in the current problem. In Colosimo et al. (2014), because the surfaces are cylindrical, this was achieved by first aligning the centers of different observed surfaces and then maximizing the cross-correlation between roundness profiles of the surfaces. Such a registration approach is appropriate for cylindrical surfaces, but cannot be applied to general surfaces that need to be handled in the current paper. 3) A typical scanned top surface of a 3D product considered in this paper would contain several hundred thousand observations. The Gaussian process model used in Colosimo et al. (2014) may not be feasible to handle data of this size. Another related research problem is about wafer surface monitoring (Wang et al., 2014), where a wafer surface is modeled by a Gaussian Kriging model in which the overall mean surface is assumed to be a linear plan and the spatial observations are assumed to follow a spatially homogeneous Gaussion process. It should be pointed out that the Gaussian Kriging model might be appropriate for describing wafer surfaces. But, from the right panel of Figure 1, it can be seen that the top surface of a 3D printing product is usually nonlinear and the assumption of a spatially homogeneous Gaussion process is difficult to justify either. Also, geometric alignment of different observed surfaces is not considered in that paper.

In this paper, we propose a novel Phase I charting scheme for monitoring products from 3D printing, as described above. More specifically, we focus on monitoring the top surfaces of 3D printing products, as demonstrated in Figure 1, since the shape of the top surfaces is often our major concern in many 3D printing applications. Our proposed method is mainly for cases when these surfaces are continuous, although discontinuous surfaces can also be handled after some proper modifications (cf., a related discussion in Section 2.2 below). Because each typical observed surface would involve a large amount of observations, the Gaussian process method considered in

the literature may not be feasible for this problem. In Phase I SPC, we mainly want to understand the variability in a collected sample of surface data, identify out-of-control (OC) observed surfaces, and so forth. See a recent paper Shi et al. (2016) on a related but different problem. Our proposed method can address all three challenges discussed earlier about the related surface profile monitoring problem. The method is described in detail in Section 2. Its performance is evaluated in Section 3. A real-data example is discussed in Section 4. Some remarks conclude the article in Section 5.

2 Phase I Monitoring of Spatial Surface Data

2.1 Model description

Let (x_{ij}, y_{ij}, z_{ij}) be the jth spatial observation of the ith surface under consideration, for $j = 1, 2, \ldots, N_i$ and $i = 1, 2, \ldots, n$, where (x_{ij}, y_{ij}) is the location of the observation in the (x, y) plane, and z_{ij} is the height of the ith surface at (x_{ij}, y_{ij}), measured by the laser scanner described in Section 1. Then, a nonparametric model for describing the observed data is

$$z_{ij} = g_i(x_{ij}, y_{ij}) + \varepsilon_{ij}, \qquad \text{for } j = 1, 2, \ldots, N_i, \;\; i = 1, 2, \ldots, n, \tag{1}$$

where $g_i(x, y)$ is the true regression function for describing the ith surface, and ε_{ij}'s are random errors with mean 0 and variance σ^2. As described in Section 1, even when the 3D printing system is IC and all products are non-defective, the geometric positions of different products still need to be adjusted, due mainly to the fact that the relative positions between the laser scanner and the top surfaces of the products might be different from product to product, the bottom parts (i.e., foundations) of different products might be (slightly) different in their height and orientation, and so forth. So, the geometric positions of the surfaces of different products are mis-matched in such cases (cf., Figure 2 for a demonstration), and it is critically important to align their positions well before we can check whether they are the same or not in a meaningful way. If we regard each surface as an image, with the height of the surface at a given position as the image intensity, then the above position alignment problem is equivalent to the *image registration* problem in the literature (cf., Bhagalia et al., 2009; Pan et al., 2009; Xing and Qiu, 2011; Qiu and Xing, 2013).

In the image registration literature, people use both parametric (Denton et al., 1999) and nonparametric (Xing and Qiu, 2011) models for describing the geometric transformation involved. The most popular parametric model is the so-called rigid-body transformation, under which the

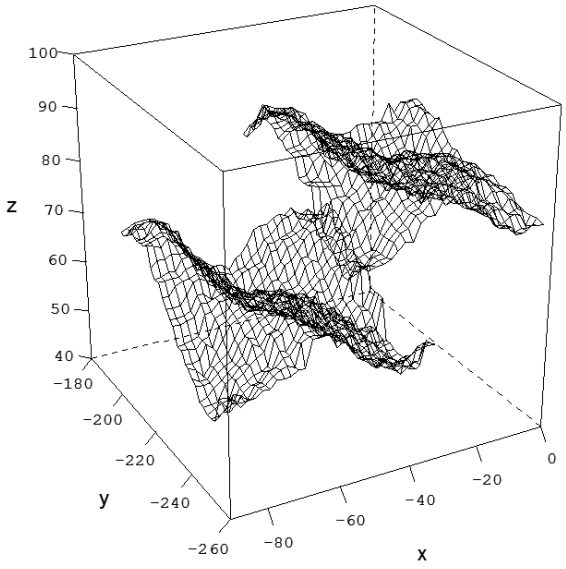

Figure 2: Demonstration of two top surfaces of two products, which need to be geometrically aligned.

Euclidean distance between any two pixels on an image will not change after the transformation. In the current surface alignment problem, we believe that rigid-body transformations are the most relevant ones, because the misalignment among different observed surfaces is mainly due to the relative position move between the laser scanner and the 3D printing products. So, that type of transformations is considered here. Besides the rigid-body transformations, another type of commonly used parametric transformations uses the so-called affine motion models, by which scales in different surfaces are allowed to be different (Qiu and Nguyen, 2008). Nonparametric transformations do not assume specific parametric forms. Therefore, they are more flexible than their parametric counterparts. Commonly used nonparametric image registration methods include those free-form deformation techniques, diffeomorphic image registration methods, and nonparametric local smoothing methods (Tustison et al., 2009; Beg et al., 2005; Qiu and Xing, 2013). These methods will not be considered here because of their complexity and heavy computing burden. Regarding the way to estimate the geometric transformation in image registration, there are two general approaches for that purpose: feature-based and intensity-based methods. By a feature-based method, we need to first select a number of corresponding features in the two images, respectively, and then find a geometrical transformation to best match the two sets of features (Althof et al., 1997; Davis et al., 1997, e.g.,). To this end, landmarks or control points are often the preferred features and they can be selected manually or automatically by a computer. Other commonly used features

include edge lines or curves, which are often detected by gradient-based methods, and regions, centroids or templates, which are usually determined by ways of thresholding and segmentation. For comprehensive surveys, see Brown (1992); Zitova and Flusser (2003). One specific feature-based image registration method is based on the Iterative Closest Point (ICP) algorithm (Besl and McKay, 1992; Yang et al., 2013). This method assumes that the geometric transformation has a parametric form. To use this method, two sets of features need to be extracted from the two related images. Then, the ICP algorithm alternates between estimating the parameters in the geometric transformation and searching for the optimal pointwise correspondence between the two sets of features. However, it is well demonstrated that this method suffers from the following drawbacks: (i) the inconvenience in extracting the related features, (ii) the heavy dependence on the quality of initialization in the algorithm, and (iii) the risk to obtain local minima instead of global minima in parameter estimation. In practice, however, feature-based methods may not be convenient to use. First, feature selection is often a time-consuming and challenging process with much arbitrariness involved. Second, it is often difficult to determine the number of needed features. The more features we select, the higher accuracy we could possibly achieve for estimating the geometrical transformation. But at the same time more effort is required. For these reasons, recent image registration research focuses more on estimating the geometric transformation based directly on the observed image intensities of the two images (i.e., the intensity-based), which is also adopted in this paper.

Without loss of generality, assume that the first surface is the reference surface, and all other surfaces need to be geometrically matched to it. Then, the rigid-body transformations have the following expression:

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \mathbf{R}_i \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} + \mathbf{T}_i, \qquad \text{for } i = 2, 3, \ldots, n, \tag{2}$$

where (x_1, y_1, z_1) is a point on the first surface, (x_i, y_i, z_i) is the corresponding point on the ith surface, $\mathbf{T}_i = (T_{ix}, T_{iy}, T_{iz})$ is the translation in the x, y, and z directions, $\mathbf{R_i}$ is the rotation matrix with the expression

$$\mathbf{R}_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_i & \sin\alpha_i \\ 0 & -\sin\alpha_i & \cos\alpha_i \end{bmatrix} \begin{bmatrix} \cos\beta_i & 0 & -\sin\beta_i \\ 0 & 1 & 0 \\ \sin\beta_i & 0 & \cos\beta_i \end{bmatrix} \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 \\ -\sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

and $\mathbf{A}_i = (\alpha_i, \beta_i, \theta_i)$ are rotating angles along the x, y, and z axes. The transformation comprising the right-hand-side of (2) is denoted as $\mathbf{G}(x_i, y_i, z_i; \mathbf{T}_i, \mathbf{A}_i) = (G_1(x_i, y_i, z_i; \mathbf{T}_i, \mathbf{A}_i), G_2(x_i, y_i, z_i; \mathbf{T}_i, \mathbf{A}_i), G_3(x_i, y_i, z_i; \mathbf{T}_i, \mathbf{A}_i))$.

2.2 Surface estimation and registration

The major idea in our proposed procedure for estimating the rigid-body transformations $\mathbf{G}(x_i, y_i, z_i; \mathbf{T}_i, \mathbf{A}_i)$ is to make the individual surfaces g_i defined in (1) as close to each other as possible after the transformations. Therefore, we need to properly estimate g_i first. To this end, since it is reasonable to assume that g_i's are continuous surfaces in most 3D printing applications, we suggest using the Nadaraya-Watson local constant kernel (LCK) smoothing procedure (Qiu, 2005, sec. 2.3) here. If the surfaces could have jumps or other singularities, then some jump-preserving surface estimation procedures can be considered (Kang and Qiu, 2014; Mukherjee and Qiu, 2011). For a given location (x, y), let us consider its circular neighborhood

$$O(x, y) = \{(u, v) : (u, v) \in \Omega \text{ and } \sqrt{(u - x)^2 + (v - y)^2} \leq h\}$$

where $h > 0$ is a bandwidth parameter that should be predetermined, and Ω is the design space. Then, the LCK estimator of $g_i(x, y)$, donated as $\widehat{g}_i(x, y)$, is defined as

$$\widehat{g}_i(x, y) = \frac{\sum_{(x_{ij}, y_{ij}) \in O(x,y)} z_{ij} K(\frac{x_{ij} - x}{h}, \frac{y_{ij} - y}{h})}{\sum_{(x_{ij}, y_{ij}) \in O(x,y)} K(\frac{x_{ij} - x}{h}, \frac{y_{ij} - y}{h})}, \quad \text{for } i = 1, 2, \ldots, n, \tag{4}$$

where K is a two-dimensional, circularly symmetric, kernel function.

As a side note, besides the kernel smoothing procedure considered here, there are several other smoothing approaches in the literature, including regression and smoothing splines, and so forth (cf., Qiu 2005, Section 2.5). Smoothing splines may not be appropriate for the current problem because of its extensive computation. As pointed out earlier, a typical observed surface in the current problem has several hundred thousand observations. Its smoothing spline estimator would be difficult to compute in such cases. In order to use a regression spline approach, appropriate knots or basis functions need to be determined in advance. Usually, the knots should be chosen at places of a given surface with large curvature. This task, together with determination of the number of knots, are often difficult to achieve. As a comparison, the LCK estimator defined in (4) is easy to compute. The estimator $\widehat{g}_i(x, y)$ is computed from all observations in the local neighborhood $O(x, y)$ only, the number of which is usually much smaller than the total number of observations

in the entire surface since the bandwidth h is often chosen small. That is the main reason why it is adopted here.

To estimate the six parameters in the transformation $\mathbf{G}(x_i, y_i, z_i; \mathbf{T}_i, \mathbf{A}_i)$, one natural approach is to consider the following minimization procedure:

$$\min_{\mathbf{T}_i, \mathbf{A}_i} \iint_{\Omega} \left[\widehat{g}_i(G_1(x, y, \widehat{g}_i(x, y); \mathbf{T}_i, \mathbf{A}_i), G_2(x, y, \widehat{g}_i(x, y); \mathbf{T}_i, \mathbf{A}_i)) - \widehat{g}_1(x, y)\right]^2 \mathrm{d}x\, \mathrm{d}y, i = 2, 3, \ldots, n. \tag{5}$$

In practice, we need to discretize the related functions for computing the integration in (5). To this end, let $\{(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}), i' = 1, 2, \ldots, \widetilde{n}_x, j' = 1, 2, \ldots, \widetilde{n}_y\}$ be an equally spaced grid that covers the entire design space Ω. Then, we can consider the following discretized version of (5):

$$\min_{\mathbf{T}_i, \mathbf{A}_i} \sum_{i'=1}^{\widetilde{n}_x} \sum_{j'=1}^{\widetilde{n}_y} \left[\widehat{g}_i(G_1(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}, \widehat{g}_i(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}); \mathbf{T}_i, \mathbf{A}_i), G_2(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}, \widehat{g}_i(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}); \mathbf{T}_i, \mathbf{A}_i)) - \widehat{g}_1(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'})\right]^2, \quad \text{for } i = 2, 3, \ldots, n. \tag{6}$$

In (6), the squared difference can be replaced by the absolute difference.

There are 6 parameters in (6). To search for their optimal values simultaneously may not be feasible in practice, due mainly to the heavy computing burden. However, the surface registration problem has the following special property: two well-aligned surfaces should share a same center (e.g., Audette et al., 2000; Colosimo et al., 2014). Based on this property, the three translation parameters can be determined first, by aligning the centers of the two related surfaces. Namely, we define their estimates by

$$\widehat{T}_{ix} = \frac{1}{N_1}\sum_{j=1}^{N_1} x_{1j} - \frac{1}{N_i}\sum_{j=1}^{N_i} x_{ij}, \quad \widehat{T}_{iy} = \frac{1}{N_1}\sum_{j=1}^{N_1} y_{1j} - \frac{1}{N_i}\sum_{j=1}^{N_i} y_{ij}, \quad \widehat{T}_{iz} = \frac{1}{N_1}\sum_{j=1}^{N_1} z_{1j} - \frac{1}{N_i}\sum_{j=1}^{N_i} z_{ij}. \tag{7}$$

Then, $\mathbf{T}_i$ can be estimated by $\widehat{\mathbf{T}}_i = (\widehat{T}_{ix}, \widehat{T}_{iy}, \widehat{T}_{iz})$, and we only need to estimate the three rotation parameters. To this end, we consider using the following iterative searching algorithm.

- Set the initial values for $\mathbf{A}_i = (\alpha_i, \beta_i, \theta_i)$ to be $\mathbf{A}_i^{(0)} = (0, 0, 0)$.
- In the kth iteration, for $k \geq 1$, evaluate the objective function
  $$\sum_{i'=1}^{\widetilde{n}_x} \sum_{j'=1}^{\widetilde{n}_y} \left[\widehat{g}_i(G_1(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}, \widehat{g}_i(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}); \widehat{\mathbf{T}}_i, \mathbf{A}_i), G_2(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}, \widehat{g}_i(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}); \widehat{\mathbf{T}}_i, \mathbf{A}_i)) - \widehat{g}_1(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'})\right]^2$$
  at 27 possible values of $\mathbf{A}_i$ with each component being the value obtained in the previous iteration or that value $\pm d_k$, where d_k is a constant depending on k. The minimizer is denoted as $\mathbf{A}_i^{(k)}$.

- The procedure stops if $\max(|\alpha_i^{(k)} - \alpha_i^{(k-1)}|, |\beta_i^{(k)} - \beta_i^{(k-1)}|, |\theta_i^{(k)} - \theta_i^{(k-1)}|) < \epsilon$, where $\epsilon > 0$ is a given small number.

The resulting estimates are denoted as $\widehat{\mathbf{A}}_i = (\widehat{\alpha}_i, \widehat{\beta}_i, \widehat{\theta}_i)$. From the above algorithm, we can see that it stops when the searched values in the two consecutive iterations are close to each other, controled by the pre-specified constant ϵ. When ϵ is chosen small enough, $\widehat{\mathbf{A}}_i$ should be close to a local optimum of the objective function, since it is obtained from the search among 27 neighboring values of $\mathbf{A}_i$. Because the true values of $\mathbf{A}_i$ are usually small in reality (i.e., the directional difference between the two related surfaces is small) and we set the initial value of $\mathbf{A}_i$ to be $(0, 0, 0)$ in the algorithm, the estimate $\widehat{\mathbf{A}}_i$ should provide a good approximation to the true value of $\mathbf{A}_i$ in most cases. In the above algorithm, there are several parameters to choose, including the bandwidth h in (4), the sequence of parameters $\{d_k\}$ and the constant ϵ. Based on our numerical experience, we suggest choosing $h \in [0.1, 0.2]$, $d_k = d_{k-1}/2$ (i.e., the bisection search), $d_0 = 10^o$, and $\epsilon = 0.001$. In the numerical examples presented in the next section, they are chosen that way. In that setup, the algorithm converges quite fast (within 10 to 20 iterations in all cases considered there).

2.3 Phase I surface monitoring

After surface estimation and registration, we are ready for Phase I surface monitoring. To this end, we need to use n geometrically aligned estimated surfaces, denoted as $\{\widehat{g}_i^*(x, y), i = 1, 2, \ldots, n\}$, where $\widehat{g}_1^*(x, y) = \widehat{g}_1(x, y)$, and for $i = 2, \ldots, n$,

$$\widehat{g}_i^*(x, y) = \widehat{g}_i(G_1(x, y, \widehat{g}_i(x, y); \widehat{\mathbf{T}}_i, \widehat{\mathbf{A}}_i), G_2(x, y, \widehat{g}_i(x, y); \widehat{\mathbf{T}}_i, \widehat{\mathbf{A}}_i)).$$

In the above expression, $(G_1(x, y, \widehat{g}_i(x, y); \widehat{\mathbf{T}}_i, \widehat{\mathbf{A}}_i), G_2(x, y, \widehat{g}_i(x, y); \widehat{\mathbf{T}}_i, \widehat{\mathbf{A}}_i))$ is the point on the surface $\widehat{g}_i(x, y)$ that matches the point (x, y) on the surface $\widehat{g}_1(x, y)$ (cf., the definition of $\mathbf{G}(x_i, y_i, z_i; \mathbf{T}_i, \mathbf{A}_i)$ at the end of Section 2.1). So, $\widehat{g}_i^*(x, y)$ is the ith estimated surface that has been aligned to $\widehat{g}_1(x, y)$. Next, we discuss Phase I surface monitoring based on these geometrically aligned estimated surfaces. First, the IC surface, denoted as $g^{(0)}(x, y)$, can be estimated by $\widehat{g}^{(0)}(x, y) = \frac{1}{n}\sum_{i=1}^n \widehat{g}_i^*(x, y)$. Then, a natural charting statistic can be defined as

$$Q_{ART} = \max_{i=1,2,\ldots,n} \frac{1}{\widetilde{n}_x \widetilde{n}_y} \sum_{i'=1}^{\widetilde{n}_x} \sum_{j'=1}^{\widetilde{n}_y} \left| \widehat{g}_i^*(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}) - \widehat{g}^{(0)}(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}) \right|, \tag{8}$$

where the subscript "A", "R" and "T" denotes the "absolute value" norm used in the above definition, and rotation and translation considered in the surface registration, respectively. The

chart gives a signal when

$$Q_{ART} > c_{ART}, \tag{9}$$

where $c_{ART} > 0$ is a control limit. Note that the construction of Q_{ART} did not take into account the possible spatial correlation of the surface observations. As discussed in Section 1, some authors used different Gaussian process models for describing such spatial correlation for surface monitoring, which is difficult to justify in practice. In the statistical literature, some existing research confirms that statistics which ignore the data correction, such as the one in (8), should still be statistically consistent for estimating the corresponding parameter (the mean absolute difference between the two surfaces $\widehat{g}_i^*$ and $g^{(0)}$ in the case of (8)), under some regularity conditions (Brabanter et al., 2011; Choi et al., 2013, e.g.,), although they would loss some efficiency in such cases. In the next section, we will provide a numerical example to show that the chart based on Q_{ART} has certain power to handle correlated data.

Obviously, when all products are IC, the charting statistic Q_{ART} in (8) does not depend on $g_i(x, y)$, for $i = 1, 2, \ldots, n$, and $g^{(0)}(x, y)$ because $g_i^*(x, y) = g^{(0)}(x, y)$ in such cases and Q_{ART} is based on $g_i^*(x, y) - g^{(0)}(x, y)$, where $g_i^*(x, y)$ denotes the ith aligned surface. So, it is possible to obtain the empirical IC distribution of Q_{ART} based on the following bootstrap procedure (Chatterjee and Qiu, 2009). Randomly select $\widetilde{n}_x\widetilde{n}_y$ elements with replacement from the set $\{\widehat{g}_i^*(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}) - \widehat{g}^{(0)}(\widetilde{x}_{i'j'}, \widetilde{y}_{i'j'}), i' = 1, 2, \ldots, \widetilde{n}_x, j = 1, 2, \ldots, \widetilde{n}_y, i = 1, 2, \ldots, n\}$ and calculate their mean absolute value. Repeat this n times, and the maximum of the n mean absolute values is used as an approximation to Q_{ART}. This entire process is then repeated for B times, and the $(1 - \alpha)$th quantile of the B approximations of Q_{ART} can then be used as the control limit value c_{ART}, where α is a given nominal IC false alarm rate (FAR). This bootstrap procedure is appropriate to use in applications when $g^{(0)}(x, y)$ is unknown but we have an IC sample of n observed surfaces. In simulation studies, $g^{(0)}(x, y)$ is known. So, we can obtain the empirical distribution of Q_{ART} by performing B repeated simulations, in each of which a value of Q_{ART} can be computed based on n simulated IC surfaces. Again, we can use the $(1 - \alpha)$th quantile of the B calculated values of Q_{ART} as the control limit value c_{ART}.

3 Simulation

In this section, we present some numerical results regarding the performance of the proposed method. Besides the chart Q_{ART} defined in (8)-(9), we also consider the following six alternative methods:

- the chart with L_2 norm in (6) and (8) (i.e., the absolute value is replaced by the squared value) and with both rotation and translation in surface registration, denoted as Q_{SRT},
- the chart with the L_2 norm in (6) and (8) and with translation in surface registration (i.e., rotation is ignored), denoted as Q_{ST},
- the chart with the L_2 norm in (6) and (8) and without surface registration, denoted as Q_S,
- the chart with L_1 norm in (6) and (8) and with translation only in surface registration, denoted as Q_{AT},
- the chart with L_1 norm in (6) and (8) and without surface registration, denoted as Q_A,
- and the Gaussian-Kriging method proposed by Wang et al. (2014).

The alternative chart Q_{SRT} is considered here to see whether it makes difference to use L_2 norm in (6) and (8), instead of the L_1 norm. In the alternative charts Q_{ST}, Q_S, Q_{AT}, and Q_A, either the rotation or both the rotation and translation of the geometric transformation are ignored. They are considered here to investigate the consequence when the geometric mislignment exists among the observed surfaces but part or the entire geometric misalignment is ignored. The Gaussian process approach is the most popular one in the surface monitoring literature and it can accommodate certain spatial data correlation. Therefore, the one by Wang et al. (2014) is also considered in this study. In the methods Q_{SRT}, Q_{ST}, Q_S, Q_{ART}, Q_{AT}, and Q_A, we choose the Epanechnikov kernel function $K(u,v) = (1-u^2-v^2)I(u^2+v^2 \leq 1)$ and $h = 0.1$ in the LCK estimation (4). More details about the Gaussing-Kriging method will be given in Section 3.3.

In the simulation, we consider the design space $\Omega = [-10, 10] \times [-10, 10]$, the IC surface

$$g^{(0)}(x,y) = xy \exp\left\{-\left(\frac{1.5x}{10}\right)^2 - \left(\frac{3y}{10}\right)^2\right\} + 5,$$

and the following four OC surfaces

$$
\begin{aligned}
g^{(1)}(x,y) &= g^{(0)}(x,y) + r_1 xy \exp\left\{-\left(\frac{1.5x}{10}\right)^2 - \left(\frac{3y}{10}\right)^2\right\} I(x>0, y>0), \\
g^{(2)}(x,y) &= g^{(0)}(x,y) + r_2 xy \exp\left\{-\left(\frac{1.5x}{10}\right)^2 - \left(\frac{3y}{10}\right)^2\right\}, \\
g^{(3)}(x,y) &= g^{(0)}(x,y) + \eta(x^2+y^2)/10^2, \\
g^{(4)}(x,y) &= g^{(0)}(x,y) + \delta,
\end{aligned}
$$

where $r_1 = 0.12, 0.24$, $r_2 = 0.04, 0.2$, $\eta = 0.3, 1$, and $\delta = 0.5, 2$. The above four types of OC surfaces represent 4 different geometric misalignments with $g^{(0)}(x,y)$. The OC surface $g^{(1)}(x,y)$ is different from the IC surface $g^{(0)}(x,y)$ in the upper-right quadrant only, $g^{(2)}(x,y)$ is different from $g^{(0)}(x,y)$ in the entire design space with the difference small in the central and border regions of the design space, $g^{(3)}(x,y)$ is also different from $g^{(0)}(x,y)$ in the entire design space with the difference large in the border regions of the design space, and $g^{(4)}(x,y)$ is different from $g^{(0)}(x,y)$ by a constant in the design space. We expect that the charts with surface registration will perform well in the first three cases, and perform badly in the last case. The plot of $g^{(0)}(x,y)$ is shown in Figure 3, and the plots of the four OC surfaces when $r_1 = 0.12$, $r_2 = 0.04$, $\eta = 0.3$ and $\delta = 0.5$ are shown in Figure 4.

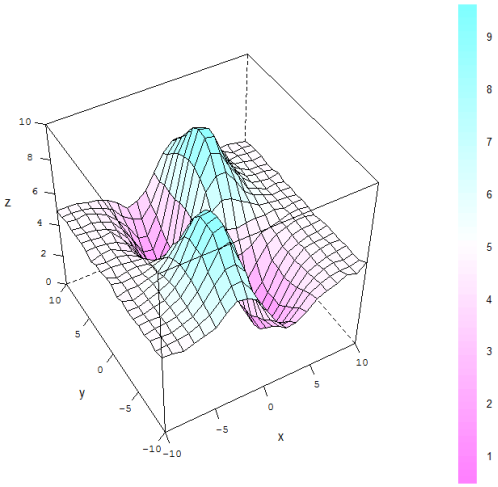

Figure 3: The IC surface $g^{(0)}(x,y)$.

The observed surfaces are generated from model (1) with the random errors ε_{ij} generated i.i.d. from the $N(0, 0.15^2)$ distribution. The observation locations $\{(x_{ij}, y_{ij})\}$ for the n surfaces

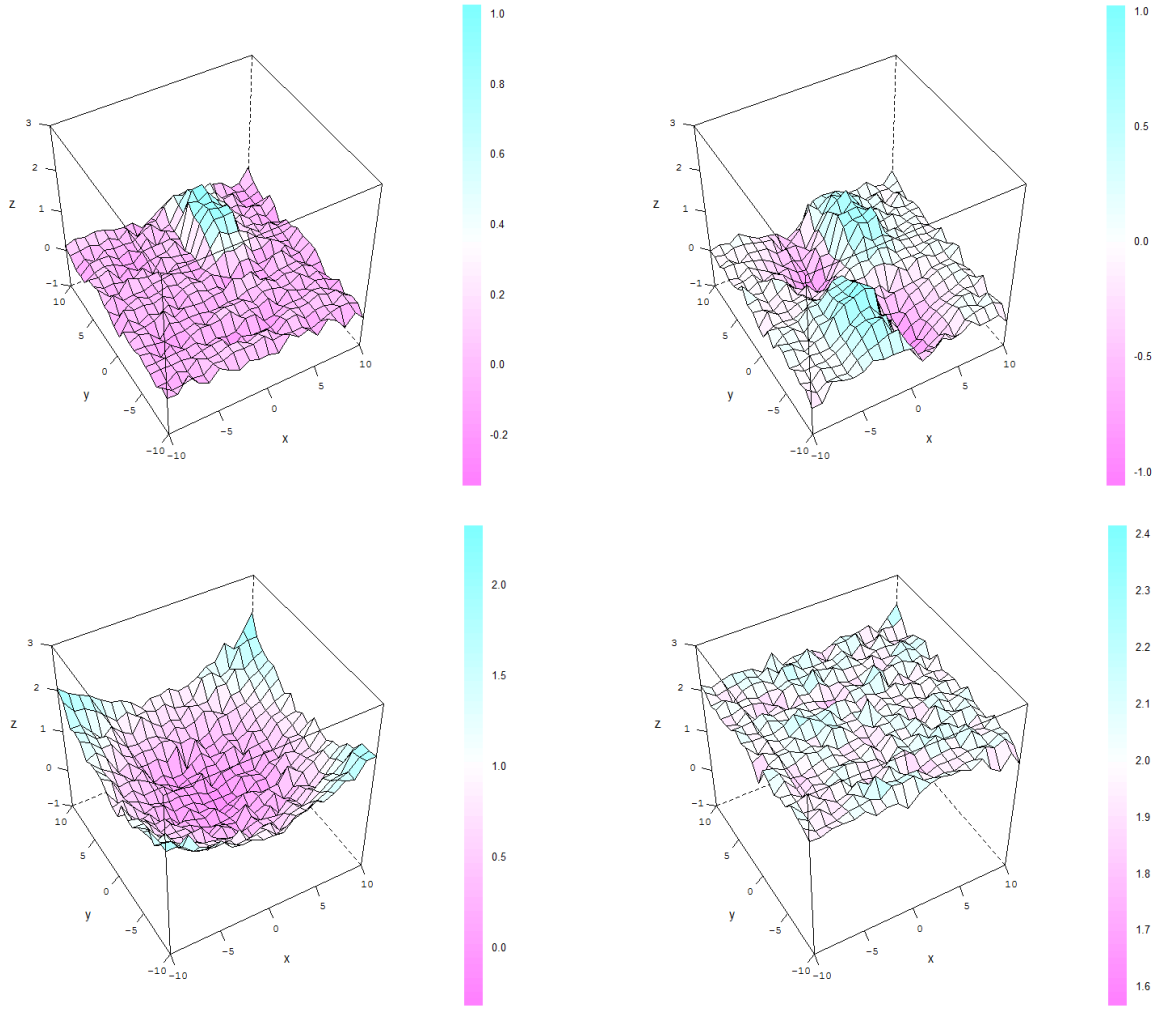

Figure 4: The four OC surfaces $g^{(1)}(x, y)$ (upper-left panel), $g^{(2)}(x, y)$ (upper-rightt panel), $g^{(3)}(x, y)$ (lower-left panel), and $g^{(4)}(x, y)$ (lower-right panel), when $r_1 = 0.12$, $r_2 = 0.04$, $\eta = 0.3$ and $\delta = 0.5$.

are generated in two steps as follows. First, an equally spaced grid is generated in Ω with the distance between two consecutive horizontal or vertical lines to be 0.1. Each grid point is then moved horizontally by s_h and vertically by s_v, where s_h and s_v are two independent random numbers generated from the $N(0, 0.02^2)$ distribution. The set of the resulting disturbed grid points is called *a disturbed grid*. Second, for the ith observed surface, generate an integer number N_i from the $U(15000, 16000)$ distribution as the number of observations of that surface. Then, the observation locations $\{(x_{ij}, y_{ij}), j = 1, 2, \ldots, N_i\}$ are obtained by randomly selecting N_i points without replacement from the disturbed grid. After the n observed surfaces are obtained, the first one is centralized as the reference surface. Namely, the observations of the reference surface are

$$\left(x_{1j} - \sum_{j=1}^{N_i} x_{1j}, \;\; y_{1j} - \sum_{j=1}^{N_i} y_{1j}, \;\; z_{1j} - \sum_{j=1}^{N_i} z_{1j} \right).$$

Each of the remaining $n-1$ observed surfaces is rotated α, β and θ degrees in the x, y, and z axes, respectively, where (α, β, θ) are independent random numbers generated from the $U[-3, 3]$ distribution, and moved by T_x, T_y and T_z along the three axes, where (T_x, T_y, T_z) are independent random numbers generated from the $U[-1, 1]$ distribution.

3.1 IC performance of the six procedures based on (6) and (8)

In this part, we compare the IC performance of the six procedures Q_{SRT}, Q_{ST}, Q_S, Q_{ART}, Q_{AT}, and Q_A that are based on (6) and (8). By the procedure described at the end of Section 2, the control limit values of the six related charts considered in this part are computed in the setup described above, based on 10,000 repeated simulations for the nominal IC FAR $\alpha = 0.05$ and $n = 10$ or 30. They are presented in Table 1. Note that the values of n considered here are relatively small, compared to those in the conventional SPC literature, because the number of 3D printing products of the same type is usually small. From the table, we can see that control limits increase slightly when n increases from 10 to 30.

Table 1: Control limits of the six charts when $\alpha = 0.05$ and $n = 10$ or 30.

n	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A
10	0.0279	0.1235	0.1769	0.3401	2.0463	1.2223
30	0.0327	0.1328	0.1889	0.3504	2.2082	1.2421

The IC performance of a control chart is evaluated by the actual IC FAR, defined to be the

proportion of the repeated simulations in which at least one signal is obtained, in the case when all surfaces are IC in each simulation. The actual IC FAR values calculated based on 1,000 simulations are presented in Table 2, when the nominal FAR is $\alpha = 0.05$ and $n = 10$ or 30. From the table, it can be seen that i) the charts Q_{ART}, Q_{AT} and Q_S are the most reliable ones among all six charts since their actual IC FAR values are the closest to the nominal IC FAR value 0.05, and ii) the charts Q_{SRT} and Q_A are reasonably good although the chart Q_{SRT} seems to be a bit conservative.

Table 2: Actual IC FAR values of the six charts based on (6) and (8) when $\alpha = 0.05$ and $n = 10$ or 30.

n	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A
10	0.040	0.052	0.047	0.051	0.050	0.056
30	0.033	0.047	0.055	0.047	0.048	0.041

3.2 OC performance of the six procedures based on (6) and (8)

Next, we evaluate the OC performance of the six charts Q_{SRT}, Q_{ST}, Q_S, Q_{ART}, Q_{AT}, and Q_A in terms of the following three criteria. The first criterion, called alarm probability (AP), is the proportion of the repeated simulations in which at least one signal is obtained, in the case when one or more surfaces are OC in each simulation (cf., Yeh et al., 2009). The second criterion, called fraction correctly classified (FCC), is defined to be the proportion of surfaces used in all repeated simulations that are correctly classified as IC or OC by a chart (cf., Chen et al., 2015). For instance, if there are 10 IC surfaces and 5 OC surfaces used in a numerical study, and a specific chart gives signals to 2 IC surfaces and 4 OC surfaces, then its FCC is $(8+4)/(10+5) = 0.8$. The last criterion, called false positive proportion (FPP), is the proportion of false signals among all signals given by a chart. In the above toy example, the FPP is $2/6 = 0.333$. Obviously, the larger the first two criteria or the smaller the third criterion, the better.

The simulation was performed in cases when $n = 10$ or 30. But, the conclusions are similar in these two cases. So, only the results when $n = 30$ are presented here. In such cases, the number of OC surfaces, denoted as n_1, is assumed to be 1, 2, 3, 6, and 9, and they are the 2nd, 4th, 6th, ..., $(2n_1)$th surfaces in a sample.

In cases when the OC surface is $g^{(1)}(x, y)$, the computed OC performance criteria based on

1000 replicated simulations are presented in Table 3. From the table, it can be seen that when the surface shift is in a quadrant of the design space only, (i) Q_{ART} performs the best among all methods in terms of all three criteria, (ii) with Q_{SRT} is slightly worse than Q_{ART}, (iii) Q_{ST} and Q_{AT} perform better than Q_S and Q_A, and (iii) Q_S and Q_A perform the worst. This example demonstrates the importance of surface registration before surface monitoring. It can be seen from the table that AP can increase quite significantly and FPP can decrease considerably as well when surface registration is considered, although the improvement in terms of FCC is less dramatic.

Table 3: Computed values of the OC performance criteria when the OC surface is $g^{(1)}(x, y)$ and $n = 30$. Numbers in bold denote the best performance.

Criteria	n_1	r_1	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A
AP	1	0.12	0.089	**0.159**	0.143	0.120	0.050	0.057
		0.24	**1.000**	**1.000**	0.226	0.200	0.052	0.059
	2	0.12	0.100	**0.198**	0.132	0.126	0.047	0.044
		0.24	**1.000**	**1.000**	0.278	0.243	0.049	0.044
	3	0.12	0.117	**0.219**	0.169	0.153	0.045	0.050
		0.24	**1.000**	**1.000**	0.343	0.284	0.046	0.041
	6	0.12	0.130	**0.258**	0.162	0.169	0.057	0.057
		0.24	**1.000**	**1.000**	0.443	0.392	0.068	0.066
	9	0.12	0.116	**0.213**	0.182	0.169	0.047	0.047
		0.24	**1.000**	**1.000**	0.428	0.395	0.066	0.070
FCC	1	0.12	0.968	**0.970**	0.963	0.963	0.965	0.965
		0.24	**0.999**	0.998	0.966	0.965	0.965	0.965
	2	0.12	0.935	**0.938**	0.930	0.930	0.932	0.932
		0.24	**0.999**	**0.999**	0.935	0.934	0.932	0.932
	3	0.12	0.902	**0.906**	0.897	0.897	0.899	0.899
		0.24	**0.999**	**0.999**	0.904	0.902	0.899	0.899
	6	0.12	0.803	**0.807**	0.798	0.798	0.799	0.799
		0.24	**0.999**	**0.999**	0.809	0.806	0.799	0.799
	9	0.12	0.702	**0.704**	0.700	0.700	0.699	0.699
		0.24	**0.999**	**0.999**	0.708	0.705	0.700	0.699
FPP	1	0.12	0.293	**0.232**	0.892	0.905	0.922	0.969
		0.24	**0.034**	0.054	0.562	0.609	0.893	0.887
	2	0.12	0.306	**0.206**	0.805	0.821	0.915	0.944
		0.24	**0.014**	0.022	0.427	0.489	0.800	0.780
	3	0.12	0.233	**0.154**	0.752	0.713	0.854	0.852
		0.24	**0.010**	0.015	0.373	0.425	0.792	0.784
	6	0.12	0.220	**0.156**	0.646	0.638	0.678	0.781
		0.24	**0.004**	0.007	0.281	0.333	0.600	0.699
	9	0.12	0.246	**0.215**	0.514	0.503	0.660	0.772
		0.24	**0.003**	0.004	0.294	0.352	0.479	0.628

The corresponding results when the OC surface is $g^{(2)}(x,y)$, $g^{(3)}(x,y)$, or $g^{(4)}(x,y)$ are presented in Tables 4-6, respectively. From the tables, it can be seen that the surface registration is generally helpful in cases with $g^{(2)}(x,y)$ and $g^{(3)}(x,y)$, and it is not helpful at all in cases when $g(x,y) = g^{(4)}(x,y)$. Remember that $g^{(4)}(x,y)$ has a constant shift from the IC surface $g^{(0)}(x,y)$ and their shapes are exactly the same. Such a constant shift would be removed in the surface registration stage. Consequently, surface monitoring after surface registration would perform worse in such cases, compared to surface monitoring without surface registration. Therefore, these results are all reasonable.

3.3 Comparison with the Gaussing-Kriging method and cases with correlated data

In this part, we compare the numerical performance of the six procedures Q_{SRT}, Q_{ST}, Q_S, Q_{ART}, Q_{AT}, and Q_A with the Gaussian-Kriging method proposed by Wang et al. (2014). For the Gaussian-Kriging method, we consider two different versions: one without surface registration and the other with surface registration, denoted as GK and GK_R, respectively. In this example, we consider two scenarios. The first scenario (denoted as S1) is that the IC and OC surfaces are $g^{(0)}(x,y)$ and $g^{(1)}(x,y)$ considered in Sections 3.1 and 3.2, and the observed surface data are independent. In the second scenario (denoted as S2), surface observations are generated from the following model:

$$z_{ij} = g_i(x_{ij}, y_{ij}) + Z(x_{ij}, y_{ij}) + \varepsilon_{ij}, \qquad \text{for } j = 1, 2, \ldots, N_i, \;\; i = 1, 2, \ldots, n,$$

where z_{ij}, $g_i(x_{ij}, y_{ij})$ and ε_{ij} are defined as in model (1) and $Z(x,y)$ is a Gaussian process satisfying the conditions that

1) $E(Z(x,y)) = 0$,

2) For any two points (x_1, y_1) and (x_2, y_2) in the design space,

$$Cov(Z(x_1,y_1), Z(x_2,y_2)) = 0.1^2 \exp\left\{-\frac{(x_1-x_2)^2}{0.02} - \frac{(y_1-y_2)^2}{0.02}\right\},$$

3) For any n points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, $(Z(x_1,y_1), Z(x_2,y_2), \cdots, Z(x_n,y_n))$ has a multivariate normal distribution.

The above covariance function is the same as that used in Wang et al. (2014). In this scenario, the IC and OC surfaces are still assumed to be $g^{(0)}(x,y)$ and $g^{(1)}(x,y)$.

Table 4: Computed values of the OC performance criteria when the OC surface is $g^{(2)}(x, y)$ and $n = 30$. Numbers in bold denote the best performance.

Criteria	n_1	r_2	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A
AP	1	0.04	0.054	0.102	**0.162**	0.142	0.037	0.033
		0.2	**1.000**	**1.000**	0.830	0.766	0.065	0.052
	2	0.04	0.060	0.122	**0.124**	0.116	0.057	0.049
		0.2	**1.000**	**1.000**	0.936	0.901	0.075	0.065
	3	0.04	0.070	0.129	**0.130**	0.119	0.059	0.061
		0.2	**1.000**	**1.000**	0.971	0.948	0.078	0.063
	6	0.04	0.083	**0.162**	0.146	0.115	0.056	0.057
		0.2	**1.000**	**1.000**	0.975	0.961	0.127	0.082
	9	0.04	0.097	**0.170**	0.140	0.131	0.055	0.049
		0.2	**1.000**	**1.000**	0.960	0.949	0.151	0.117
FCC	1	0.04	0.966	**0.967**	0.962	0.962	0.965	0.966
		0.2	**0.999**	0.998	0.988	0.986	0.965	0.966
	2	0.04	0.933	**0.934**	0.930	0.930	0.932	0.932
		0.2	**0.999**	**0.999**	0.981	0.976	0.932	0.932
	3	0.04	0.900	**0.901**	0.896	0.897	0.898	0.898
		0.2	**0.999**	**0.999**	0.969	0.960	0.900	0.899
	6	0.04	0.801	**0.803**	0.798	0.798	0.799	0.799
		0.2	**0.999**	**0.999**	0.917	0.902	0.801	0.800
	9	0.04	0.701	**0.703**	0.699	0.699	0.699	0.699
		0.2	**0.999**	0.998	0.808	0.793	0.702	0.701
FPP	1	0.04	0.691	**0.490**	0.912	0.902	0.974	0.943
		0.2	**0.037**	0.046	0.154	0.180	0.783	0.789
	2	0.04	0.516	**0.040**	0.886	0.891	0.847	0.857
		0.2	**0.014**	0.022	0.080	0.079	0.659	0.764
	3	0.04	0.520	**0.391**	0.863	0.860	0.873	0.904
		0.2	**0.010**	0.014	0.064	0.068	0.518	0.640
	6	0.04	0.322	**0.228**	0.719	0.734	0.726	0.768
		0.2	**0.005**	0.007	0.036	0.041	0.414	0.489
	9	0.04	0.371	**0.293**	0.639	0.638	0.633	0.731
		0.2	**0.002**	0.008	0.066	0.070	0.358	0.400

Table 5: Computed values of the OC performance criteria when the OC surface is $g^{(3)}(x, y)$ and $n = 30$. Numbers in bold denote the best performance.

Criteria	n_1	η	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A
AP	1	0.3	0.222	**0.639**	0.130	0.123	0.044	0.047
		1	**1.000**	**1.000**	0.670	0.746	0.116	0.166
	2	0.3	0.294	**0.766**	0.144	0.131	0.049	0.052
		1	**1.000**	**1.000**	0.839	0.897	0.179	0.272
	3	0.3	0.324	**0.795**	0.123	0.129	0.074	0.085
		1	**1.000**	**1.000**	0.860	0.913	0.227	0.340
	6	0.3	0.247	**0.677**	0.132	0.132	0.068	0.084
		1	**1.000**	**1.000**	0.866	0.919	0.319	0.509
	9	0.3	0.181	**0.478**	0.150	0.165	0.068	0.088
		1	**1.000**	**1.000**	0.755	0.842	0.344	0.582
FCC	1	0.3	0.972	**0.986**	0.963	0.963	0.965	0.965
		1	**0.999**	**0.999**	0.983	0.987	0.968	0.969
	2	0.3	0.942	**0.969**	0.929	0.930	0.932	0.932
		1	**0.999**	0.998	0.969	0.975	0.937	0.940
	3	0.3	0.911	**0.946**	0.897	0.897	0.899	0.899
		1	**0.999**	0.998	0.949	0.958	0.905	0.909
	6	0.3	0.807	**0.843**	0.797	0.798	0.800	0.800
		1	**0.999**	**0.999**	0.862	0.880	0.809	0.818
	9	0.3	0.704	**0.720**	0.698	0.699	0.700	0.700
		1	**0.999**	**0.999**	0.744	0.761	0.710	0.716
FPP	1	0.3	0.155	**0.078**	0.922	0.920	0.915	0.912
		1	**0.034**	0.043	0.175	0.140	0.392	0.278
	2	0.3	0.104	**0.042**	0.890	0.866	0.840	0.717
		1	**0.017**	0.022	0.093	0.074	0.208	0.193
	3	0.3	0.062	**0.029**	0.824	0.752	0.747	0.689
		1	**0.010**	0.016	0.080	0.063	0.186	0.157
	6	0.3	0.167	**0.044**	0.759	0.699	0.561	0.525
		1	**0.003**	0.005	0.072	0.058	0.144	0.141
	9	0.3	0.175	**0.080**	0.662	0.617	0.438	0.476
		1	**0.002**	0.003	0.113	0.091	0.170	0.235

Table 6: Computed values of the OC performance criteria when the OC surface is $g^{(4)}(x, y)$ and $n = 30$. Numbers in bold denote the best performance.

Criteria	n_1	δ	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A
AP	1	0.5	0.039	0.052	0.114	0.098	0.116	**0.177**
		2	0.027	0.044	0.123	0.119	0.811	**0.873**
	2	0.5	0.033	0.046	0.111	0.107	0.168	**0.267**
		2	0.031	0.048	0.118	0.111	0.947	**0.974**
	3	0.5	0.038	0.055	0.130	0.127	0.173	**0.311**
		2	0.037	0.057	0.142	0.124	0.987	**0.994**
	6	0.5	0.039	0.058	0.113	0.109	0.291	**0.493**
		2	0.036	0.061	0.129	0.113	0.998	**1.000**
	9	0.5	0.033	0.044	0.143	0.121	0.311	**0.551**
		2	0.020	0.033	0.121	0.118	**1.000**	**1.000**
FCC	1	0.5	0.965	0.965	0.963	0.963	0.968	**0.969**
		2	0.966	0.965	0.962	0.962	0.992	**0.994**
	2	0.5	0.932	0.932	0.930	0.930	0.936	**0.940**
		2	0.932	0.932	0.930	0.930	0.985	**0.989**
	3	0.5	0.899	0.898	0.897	0.897	0.903	**0.909**
		2	0.899	0.898	0.896	0.896	0.980	**0.985**
	6	0.5	0.799	0.799	0.798	0.798	0.808	**0.816**
		2	0.799	0.799	0.798	0.798	0.955	**0.968**
	9	0.5	0.700	0.700	0.698	0.698	0.708	**0.716**
		2	0.700	0.699	0.698	0.698	0.914	**0.936**
FPP	1	0.5	0.950	0.963	0.977	0.972	0.388	**0.337**
		2	0.963	0.932	0.993	0.992	**0.043**	0.048
	2	0.5	0.939	0.957	0.914	0.926	0.254	**0.196**
		2	0.938	0.942	0.892	0.872	**0.032**	0.036
	3	0.5	0.921	0.909	0.824	0.863	0.237	**0.143**
		2	1.000	0.933	0.886	0.899	**0.018**	0.025
	6	0.5	0.821	0.793	0.744	0.758	**0.170**	**0.170**
		2	0.784	0.823	0.711	0.703	**0.017**	0.021
	9	0.5	0.600	0.587	0.646	0.674	**0.211**	0.225
		2	0.800	0.758	0.708	0.714	**0.027**	0.036

Control limit values of all related charts considered in this part are computed in the above setup based on 2,500 repeated simulations for the nominal IC FAR of $\alpha = 0.05$. Because similar conclusions can be made in cases when $n = 10$ and $n = 30$, we only present the results when $n = 30$. The actual IC FAR values calculated based on 1,000 simulations are presented in Table 7. From the table, it can be seen that: i) in S1, the charts Q_{ART}, Q_{AT} and Q_S are the most reliable ones among all eight charts, since their actual IC FAR values are the closest to the nominal level of 0.05, and the charts Q_{SRT}, GK and GK_R seem the most unreliable ones; and ii) in S2, the charts Q_{SRT}

and Q_{AT} are quite reliable, while the charts Q_{ART}, Q_A and GK_R seem a little unreliable.

Table 7: Actual IC FAR values of the eight charts when $\alpha = 0.05$ and $n = 30$.

Scenario	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A	GK	GK_R
S1	0.033	0.047	0.055	0.047	0.048	0.041	0.067	0.028
S2	0.050	0.059	0.054	0.047	0.054	0.045	0.046	0.072

Table 8 present the results regarding the OC performance of the eight charts in cases S1 abd S2 when there is only one OC surface among n surfaces and when $r_1 = 0.12$ or 0.24 in $g^{(1)}(x, y)$. From the table, we can see that: i) Q_{SRT}, Q_{ART} and GK_R perform reasonably well in S1 in all three measures, ii) GK does not work well, as expected, because it did not take into account surface registration, and iii) in S2 when the surface data are correlated, it seems that most charts perform worse, compared to their performance in S1, and the charts Q_{SRT} and Q_{ART} perform relatively well, especially when $r_1 = 0.24$.

Table 8: Computed values of the OC performance criteria when the OC surface is $g^{(1)}(x, y)$ and $n = 30$.

Scenario	Criteria	r_1	Q_{SRT}	Q_{ART}	Q_{ST}	Q_{AT}	Q_S	Q_A	GK	GK_R
S1	AP	0.12	0.089	0.159	0.143	0.120	0.050	0.057	0.038	1.000
		0.24	1.000	1.000	0.226	0.200	0.052	0.059	0.043	1.000
	FCC	0.12	0.968	0.970	0.963	0.963	0.965	0.965	0.965	0.996
		0.24	0.999	0.998	0.966	0.965	0.965	0.965	0.965	0.993
	FPP	0.12	0.293	0.232	0.892	0.905	0.922	0.969	1.000	0.097
		0.24	0.034	0.054	0.562	0.609	0.893	0.887	1.000	0.066
S2	AP	0.12	0.044	0.037	0.070	0.063	0.048	0.043	0.045	0.054
		0.24	0.114	0.133	0.103	0.096	0.041	0.037	0.048	0.048
	FCC	0.12	0.965	0.966	0.965	0.965	0.965	0.965	0.965	0.965
		0.24	0.968	0.969	0.967	0.966	0.966	0.965	0.965	0.965
	FPP	0.12	0.878	0.846	0.836	0.838	0.940	0.958	1.000	1.000
		0.24	0.342	0.299	0.472	0.545	0.930	0.925	1.000	1.000

4 A Real-Data Example

In this section, we illustrate the application of the proposed surface monitoring chart using a real 3D printer and its products. The 3D printer used here is the Makerbot Replicator 2 that is owned by the Marston Science Library of the University of Florida, and its photo was shown in the left panel

of Figure 1. A NextEngine 3D scanner is also used for generating surface observations, as described in Section 1. Before the 3D printer can print products, we need to provide the specification of the shape of an product. In this example, the IC product looks like the one shown in the right panel of Figure 1, with the design space to be $[-30, 30] \times [-30, 30]$ (in mm^2) and the z values (i.e., heights at different (x, y) positions) in the range $[10, 30]$ mm. The top surface of the designed IC product is shown in Figure 5(b). We then let the 3D printer print 18 different IC products. For each product, the number of observations generated by the laser scanner ranges between 141,000 and 158,500. A small portion of the (x, y) positions of the observations of one product is shown in Figure 5(a). It can be seen that these positions are indeed irregularly spaced, as discussed in Section 1.

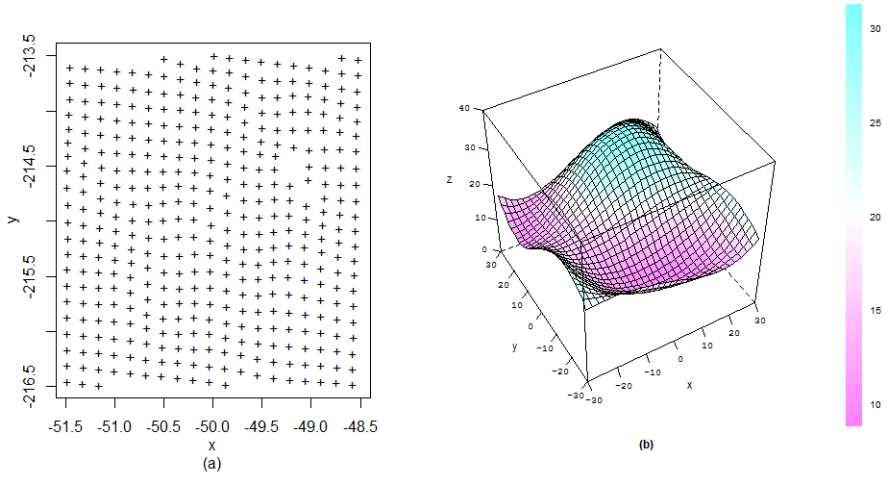

Figure 5: (a) A portion of all observation locations of a typical 3D printing product. (b) The top surface of the designed IC product for the 3D printer.

Next, we let the 3D printer print 6 OC products with different types of shifts. Their top surfaces are shown in Figure 6, together with the top surfaces of 3 IC products for comparison purposes. The first OC product just has a constant shift, as $g^{(4)}(x, y)$ used in the simulation examples discussed in Section 3. The remaining 5 OC products have shifts with different shapes and sizes, with the second, fourth, and sixth ones having relatively large shifts, and the third and fifth ones having relatively small shifts.

We then use the six control charts Q_{SRT}, Q_{ART}, Q_{ST}, Q_{AT}, Q_S and Q_A for monitoring these products. For each of them, their control limits are obtained by a bootstrap re-sampling procedure

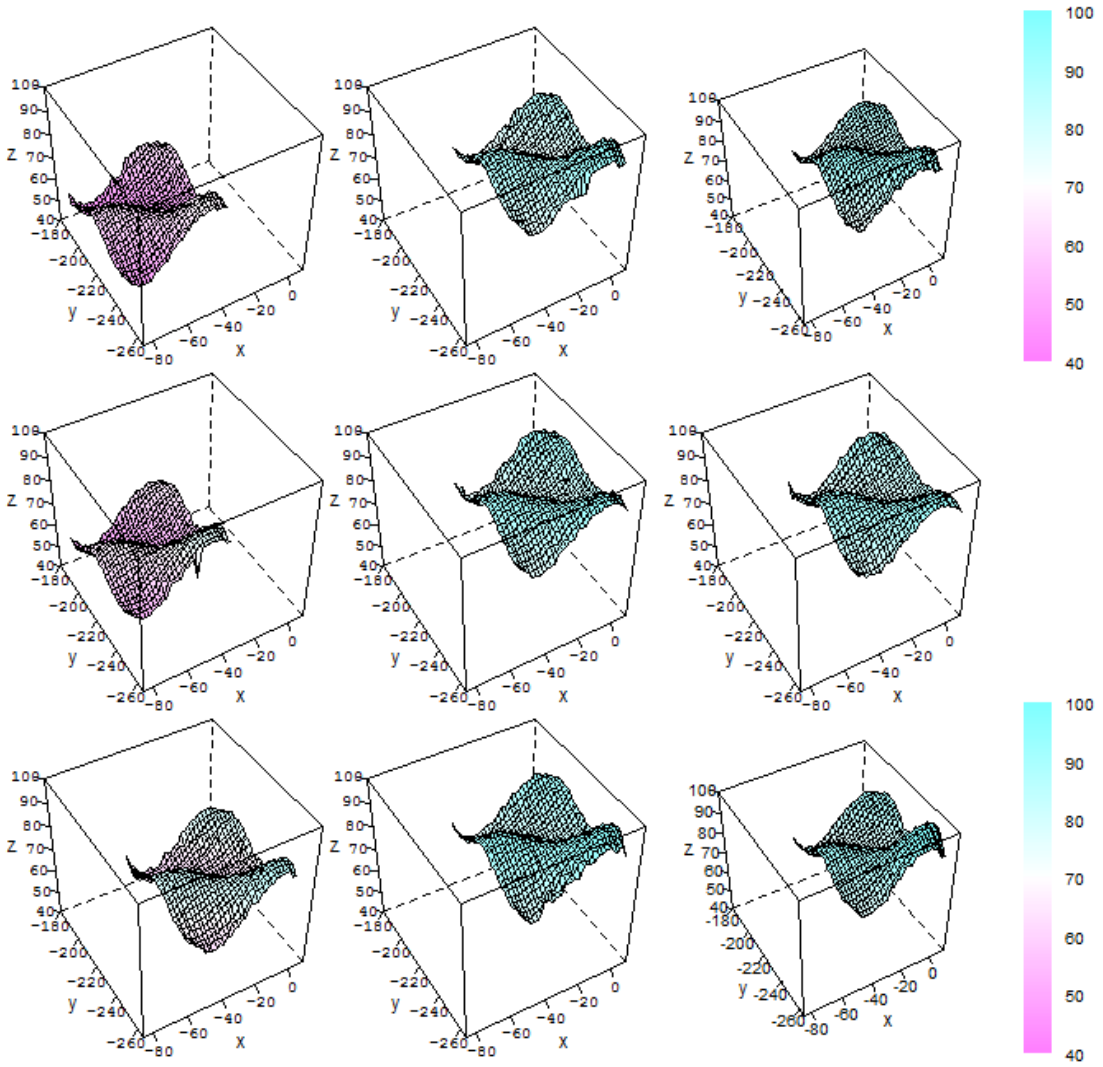

Figure 6: Surfaces in the first column are from three IC products and the ones in the second and third columns are from the OC products. The labels of the surfaces specify their positions in the sequence of all 24 products for Phase I monitoring.

described as follows. First, we randomly select 24 surfaces with replacement from the 18 IC surfaces. From each selected surface, we randomly select 15,000 observations without replacement as the resample surface. From the resulting 24 resample surfaces, we can calculate the related charting statistic value using a formula like (8). This process is then repeated for $B = 1,000$ times. The 0.95 quantile of the 1,000 values of the charting statistic is used as the control limit. The kernel function and the related bandwidth used in (4) are chosen to be the same as those in Section 3. For monitoring the total of 24 products, the positions of the 6 OC products in the sequence are randomly selected. They are placed at the 10th, 11th, 14th, 17th, 20th and 23rd places, respectively. The six control charts are shown in Figure 7. From the plots in that figure, it can be seen that (i) the first OC products can not be detected by all six methods, (ii) Q_{SRT} and Q_{ART} can detect 4 out of the remaining 5 OC products, with the 5th OC product undetected, (iii) Q_{ST} can detect 4 OC products while Q_{AT} can detect 3, and (iv) Q_S and Q_A cannot detect any OC products. Result (i) when the shift is a constant can be explained in the same way as that in Section 3 about cases with $g^{(4)}(x, y)$. Result (ii) shows that Q_{SRT} and Q_{ART} perform well in this example. The OC product that they cannot detect is the one with a relatively small shift size. Result (iii) shows that Q_{ST} and Q_{AT} also perform reasonably well. One explanation is that the orientation changes in the OC products might be quite small in this example. Result (iv) shows that surface monitoring without surface registration in advance is not effective at all, which is consistent with the results in Section 3.

5 Conclusions and Future Research

We have discussed quality control for the products of 3D printing in this paper, which has become an important research problem as 3D printing gets more and more popular in recent years. It has been demonstrated that quality control of 3D printing products is challenging due mainly to the complexity of their shapes, irregularity of the locations of observations obtained by laser scanners, and geometric misalignment among different products. In the previous sections, we have discussed a new Phase I monitoring scheme that is effective for monitoring the top surfaces of 3D printing products. One important finding is that this method is reliable only when the top surfaces of different products are properly aligned beforehand.

There are still many issues that need to be addressed in our future research. For instance, the

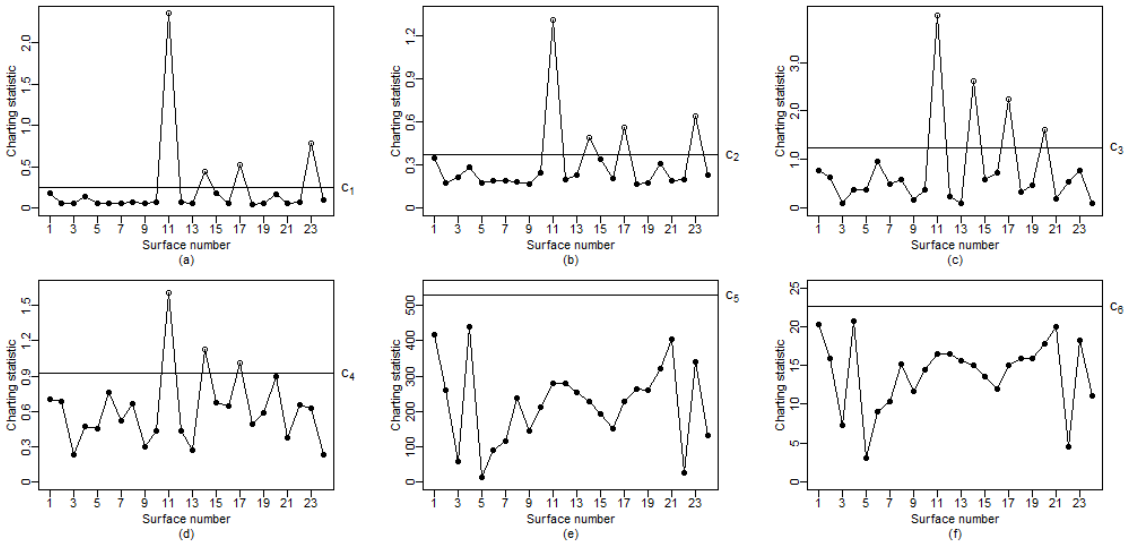

Figure 7: Phase I control charts for 3D printing surfaces. (a) Q_{SRT}, (b) Q_{ART}, (c) Q_{ST}, (d) Q_{AT}, (e) Q_S, and (f) Q_A.

current method is for Phase I process monitoring only. Control charts for Phase II online process monitoring need to be developed in the future. Also, in the current method, surface estimation, surface registration, and surface monitoring are finished in separate procedures. It might simplify its application if we can combine these procedures and finish all these tasks in a single procedure. Furthermore, in the current method, the three translation parameters are estimated using a simple center alignment procedure (7), before the three rotation parameters are estimated by a searching algorithm, for simplifying the overall computation. It may increase the accuracy of the parameter estimates if we estimate the six parameters simultaneously, which will be studied in our future research.

Acknowledgments: The authors thank the editor, the associate editor and two referees for many constructive comments and suggestions, which greatly improved the quality of the paper. This research is partially supported by an NSF grant.

References

Althof, R., Wind, M., and Dobbins, J. (1997). A rapid and automatic image registration algorithm with subpixel accuracy. *IEEE Transactions on Medical Imaging*, 19:308–316.

Audette, M. A., Ferrie, F. P., and Peters, T. M. (2000). An algorithmic overview of surface registration techniques for medical imaging. *Medical image analysis*, 4:201–217.

Beg, F., Miller, M., Trouv'e, A., and Younes, L. (2005). Computing deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, 6:139–157.

Besl, P. and McKay, N. (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:239–256.

Bhagalia, R., Fessler, J., and Kim, B. (2009). Accelerated nonrigid intensity-based image registration using importance sampling. *IEEE Transactions on Medical Imaging*, 28:1208–1216.

Brabanter, K., Brabanter, J., Suykens, J., and Moor, B. (2011). Kernel regression in the presence of correlated errors. *Journal of Machine Learning Research*, 12:1955–1976.

Brown, L. (1992). A survey of image registration technique. *ACM Computing Surveys*, 24:326–376.

Chatterjee, S. and Qiu, P. (2009). Distribution-free cumulative sum control charts using bootstrap-based control limits. *Annals of Applied Statistics*, 3:349–369.

Chen, Y., Birch, J. B., and Woodall, W. H. (2015). Cluster-based profile analysis in phase 1. *Journal of Quality Technology*, 47:14–29.

Choi, I., Li, B., and Wang, X. (2013). Nonparametric estimation of spatial and space-time covariance function. *Journal of Agricultural, Biological, and Environmental Statistics*, 18:611–630.

Colosimo, B. M., Cicorella, P., Pacella, M., and Blaco, M. (2014). From profile to surface monitoring: Spc for cylindrical surfaces via gaussian processes. *Journal of Quality Technology*, 46:95–113.

Davis, M., Khotanzad, A., Flamig, D., and Harms, S. (1997). Physics-based coordinate transformation for 3-d image matching. *IEEE Transactions on Medical Imaging*, 19:317–328.

del Castillo, E., Colosimo, B. M., and Tajbakhsh, S. (2013). Geodesic gaussian processes for the reconstruction of a free-form surface. *submitted paper. Available as technical report at http://www2. ie. psu. edu/Castillo/researcli/EngineeringStatistics/publications. htm.*

Denton, E., Sonoda, L., Rueckert, D., Rankin, S., Hayes, C., Leach, M., Hill, D., and Hawkes, D. (1999). Comparison and evaluation of rigid, affine, and nonrigid registration of breast mr images. *Journal of Computer Assisted Tomography*, 23:800–805.

Hawkins, D., Qiu, P., and Kang, C. (2003). The changepoint model for statistical process control. *Journal of Quality Technology*, 35:355–366.

Jin, J. and Shi, J. (1999). Feature-preserving data compression of stamping tonnage information using wavelets. *Technometrics*, 41:327–339.

Kang, L. and Albin, S. (2000). On-line monitoring when the process yields a linear profile. *Journal of Quality Technology*, 32:418–426.

Kang, Y. and Qiu, P. (2014). Jump detection in blurred regression surfaces. *Technometrics*, 56:539–550.

Kim, K., Mahmoud, M., and Woodall, W. (2003). On the monitoring of linear profiles. *Journal of Quality Technology*, 35:317–328.

Mukherjee, P. and Qiu, P. (2011). 3-d image denoising by local smoothing and nonparametric regression. *Technometrics*, 53:196–208.

Page, E. (1954). Continuous inspection scheme. *Biometrika*, 41:100–115.

Pan, W., Qin, K., and Chen, Y. (2009). An adaptable-multilayer fractional fourier transform approach for image registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:400–412.

Qiu, P. (2005). *Image processing and jump regression analysis*. New York: John Wiley & Sons.

Qiu, P. (2014). *Introduction to Statistical Process Control*. Boca Raton, FL: Chapman Hall/CRC.

Qiu, P. and Nguyen, T. (2008). On image registration in magnetic resonance imaging. *IEEE Proceedings of the 2008 International Conference on BioMedical Engineering and Informatics*, pages 753–757.

Qiu, P. and Xing, C. (2013). On nonparametric image registration. *Technometrics*, 55:174–188.

Qiu, P., Zou, C., and Wang, Z. (2010). Nonparametric profile monitoring by mixed effects modeling (with discussions). *Technometrics*, 52:265–293.

Roberts, S. (1959). Control chart tests based on geometric moving averages. *Technometrics*, 1:239–250.

Shewhart, W. (1931). *Economic control of quality of manufactured product.* New York: D. Van Nostrand Company.

Shi, Z., Apley, D., and Runger, G. (2016). Discovering the nature of variation in nonlinear profile data. *Technometrics*, page in press.

Tustison, N., Avants, B., and Gee, J. (2009). Directly manipulated freeform deformation image registration. *IEEE Transactions on Image Processing*, 18:624–635.

Wang, A., Wang, K., and Tsung, F. (2014). Statistical surface monitoring by spatial-structure modeling. *Journal of Quality Technology*, 46:359–376.

Xia, H., Ding, Y., and Wang, J. (2008). Gaussian process method for form error assessment using coordinate measurements. *IIE Transactions*, 40:931–946.

Xing, C. and Qiu, P. (2011). Intensity based image registration by nonparametric local smoothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:2081–2092.

Yang, J., Li, H., and Jia, Y. (2013). Go-icp: solving 3d registration efficiently and globally optimally. *International Conference on Computer Vision (ICCV)*, pages 1457–1464.

Yeh, A. B., Huwang, L., and Li, Y.-M. (2009). Profile monitoring for a binary response. *IIE Transactions*, 41:931–941.

Zitova, B. and Flusser, J. (2003). Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000.

Zou, C., Tsung, F., and Wang, Z. (2008). Monitoring profiles based on nonparametric regression methods. *Technometrics*, 50:512–526.