



Fast hierarchical fusion model based on least squares B-splines approximation

Luca Pagani^a, Jian Wang^b, Bianca M. Colosimo^c, Xiangqian Jiang^a, Paul J. Scott^{a,*}

^a EPSRC Future Metrology Hub, University of Huddersfield, Huddersfield, Queensgate, HD13DH, UK

^b State Key Lab of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, 430074, PR China

^c Department of Mechanical Engineering, Politecnico di Milano, via La Masa 1, 20156, Milan, Italy

ARTICLE INFO

Keywords:

Data fusion
Gaussian process
B-splines
Metrology

ABSTRACT

With manufacturing shifting from traditional products to high value products, the complexity and accuracy of the products are increasing in order to reduce energy costs, create friendly environment and better health care. Structured surfaces, freeform surfaces, and other functional engineering surfaces are becoming the core part of high value manufacturing products. However, measurement of these surfaces is becoming very difficult due to instrumental limitations including measurement range, speed, resolution and accuracy. Multi-instruments/sensors measurement are now being developed for freeform and structured surface assessment, which requires the fusion of the data into a unified system to achieve larger dynamic measurements with greater reliability. This paper discusses the process of combining data from several information sources (instruments/sensors) into a common representational format and the surface topography can be reconstructed using Gaussian processes and B-spline techniques. In this paper the Gaussian process model is extended in order to take into account the uncertainty propagation and a new data fusion model based on least squares B-splines that drastically reduce the computational time are presented. The results are validated by two for freeform surface measurements.

1. Introduction

In modern manufacturing technology, manufactured surfaces are characterized by complex features, designed to meet functional specifications. In this competitive environment, the ability to rapidly design, production and inspection of the specifications of these pieces is becoming essential. To this aim, measurement with more than one sensor can improve the accuracy and the precision of the measurement and decreases the acquisition time. The main idea of a multiple sensor measurement is to use different sensors to acquire the same object, the results are then combined to improve the results of each single measurement. A classification of multiple sensor systems according to the way the information contribute to the final objective can be found in Girao et al. [1]. A multiple sensor system can be classified into: (i) complementary: the sensors are not directly dependent but their output can be combined to yield better data, under a pre-defined criterion; (ii) competitive: the sensors provide independent measurements of the same quantity They can be identical or can use different measuring methods and they compete in the sense that when discrepancies occur, a decision on which to believe must be made, sensors are put in competition either to increase the reliability of a system or to provide it

with fault tolerance; (iii) cooperative: the sensors' outputs are combined to produce information that is unavailable from each sensor if used alone; (iv) independent: all the remaining cases, basically when the sensors' outputs are not combined. The process to combine multiple data from different sensors is defined in Weckenmann et al. [2] as *combining data from several sources (sensors) in order that the metrological evaluation can benefit from all the available sensor information data*.

The literature on data fusion often suggests sequential use of the sensors. In Carbone et al. [3] and Shen et al. [4] optical and contact sensors are sequentially used for reverse engineering applications. First, a point cloud is acquired with an optical system and there is a first digitization of the object. With this information, it is possible to write a CMM part program that acquires the desired point cloud. The final CAD model is thus built by refining the first raw model (based on non contact data) with the more precise CMM data points.

An example of competitive data fusion can be found in Ramasami et al. [5] where a multiscale approach based on wavelet decomposition is presented. The authors show how an appropriate multiscale model can increase the measurement result in micro-scale manufacturing.

Another example where the two sensors are used in a competitive way can be found in Xia et al. [6]. The authors used a data fusion

* Corresponding author.

E-mail address: p.j.scott@hud.ac.uk (P.J. Scott).

<https://doi.org/10.1016/j.precisioneng.2019.08.007>

Received 26 January 2019; Received in revised form 5 August 2019; Accepted 19 August 2019

Available online 22 August 2019

0141-6359/© 2019 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

technique in order to align point clouds coming from two different sensors. In this case, the main idea is to reconstruct the information provided by the Lo-Fi sensor with a Bayesian Gaussian Process (GP) model. Then, the alignment with the Hi-Fi data is performed thanks to a local kernel smoothing technique and with a rigid transformation matrix (translation and rotation).

Starting from the approach proposed by Xia et al. [6], Colosimo et al. [7] presented a data fusion model to merge data from non contact and contact devices with a GP model. The model was applied to reconstruct a surface with application to normal and large scale metrology. The fusion process was performed using contact data to locally correct the model of the non contact point cloud.

As an alternative approach, Wang et al. [8] summarized different weighted least squares fusion techniques for homogeneous multi-sensor data fusion, which show remarkable efficiency enhancement than GP-based methods.

Yin et al. [9] proposed a Bayesian multiscale data fusion model by designing composite covariance functions. The authors proposed to use different covariance functions to estimate different features composing the surface: smooth or rough.

In their following work Yin et al. [10] proposed an intelligent sampling plan for the data fusion model based on the uncertainty of the prediction.

Following the work of Colosimo et al. [7], the uncertainty from the Lo-Fi model is taken into account in the fusion model and an equation for the uncertainty propagation is provided. However, a fatal disadvantage of a data fusion model based on GP is the high estimation time, due to the time necessary to solve a large linear system multiple times. If the number of measured points is high, it may be not possible to estimate the model parameters, due to the limited CPU memory. To overcome these issues a data fusion model based on B-splines approximation is proposed. The proposed model allows to estimate the parameters quicker compared to the existing data fusion models based on Gaussian process, achieving similar reconstruction performances.

The paper is structured as follows: in Section 2 the data fusion models are described, in Section 3 the models are applied to real test cases and in Section 5 conclusions and future developments are given.

2. Fusion model

In this Section, the general scheme of an hierarchical fusion model is presented. The model described in Colosimo et al. [7] is briefly recalled and a formula to estimate the uncertainty of prediction of the fusion model is presented in Section 2.1. In Section 2.2 the proposed B-splines model and its properties are discussed.

Data coming from different measurement systems with different precision and/or accuracy, called Low-Fidelity (Lo-Fi) and High-Fidelity (Hi-Fi) data, are analyzed. Usually the number of Lo-Fi data is greater than the number of Hi-Fi data, but the Hi-Fi data are more accurate. The aim of a fusion model is to link the Lo-Fi to Hi-Fi data in order to achieve better prediction of the real, unknown, surface points. The data acquired using the Lo-Fi device can be expressed by

$$z_i(\mathbf{s}) = f_i(\mathbf{s}) + \varepsilon_i(\mathbf{s}) \quad (1)$$

where $\mathbf{s} = (u, v)^T$ is a vector describing the spatial location, $z_i(\mathbf{s})$ is the value of the measured Lo-Fi point at the spatial location \mathbf{s} , $f_i(\mathbf{s})$ is a function of the spatial location that describe the measured surface and $\varepsilon_i(\mathbf{s})$ represents the randomness of the measurement with the Lo-Fi device.

Once the Lo-Fi data are described, a model to merge the two sets of data can be defined as

$$z_h(\mathbf{s}) = \hat{z}_i(\mathbf{s}) + f_\Delta(\mathbf{s}) + \varepsilon_\Delta(\mathbf{s}) \quad (2)$$

where $z_h(\mathbf{s})$ is the value of the measured Hi-Fi point at the spatial location \mathbf{s} , $\hat{z}_i(\mathbf{s})$ is the prediction with the Lo-Fi model at location same location, $f_\Delta(\mathbf{s})$ is the function that describe the discrepancy between Lo-

Fi and the Hi-Fi data and $\varepsilon_\Delta(\mathbf{s})$ is the measurement error of the difference between the Hi-Fi device and the prediction of the Lo-Fi model.

Since the Lo-Fi and the Hi-Fi data can be acquired at different spatial locations, the prediction with the Lo-Fi model has to be used in the linkage model. A further advantage of using the prediction is that the random measurement error should be removed.

2.1. Gaussian Process model

Gaussian Process (GP) is a model commonly used in spatial statistics [11] and to describe the output of computer experiments [12]. Many authors, see for example Kennedy and O'Hagan [13], Qian et al. [14], Xia et al. [6], Colosimo et al. [7], have used this model to describe both Lo-Fi and Hi-Fi data in a data fusion framework. The observed Lo-Fi value, z_i , at a generic spatial location \mathbf{s} can be described, using a GP model, by the relation

$$z_i(\mathbf{s}) = \mathbf{f}_i^T(\mathbf{s})\boldsymbol{\beta}_i + \eta(\mathbf{s}) + \varepsilon_i(\mathbf{s}), \quad i = 1, \dots, n_i \quad (3)$$

where $\mathbf{f}_i^T(\mathbf{s})$ is the transpose of $\mathbf{f}_i(\mathbf{s}) \in \mathbb{R}^r$, which is a vector of known basis functions which act as regressors, $\boldsymbol{\beta}_i \in \mathbb{R}^r$ is a vector of unknown parameters, $\eta(\mathbf{s}) \sim \mathcal{GP}(0, \sigma_\eta^2, \vartheta_i)$ is a GP with zero mean and variance-covariance matrix defined through the parameter σ_η^2 and the vector of scale parameters $\vartheta_i \in \mathbb{R}^2$. The term $\varepsilon_i(\mathbf{s})$ is the random term of the Lo-Fi points, known also as nugget, which is assumed to be independent and normally distributed, i.e. $\varepsilon_i(\mathbf{s}) \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. The nugget effect is included to represent randomness characterizing the data.

The core of the GP is the variance-covariance matrix defined as

$$\text{Cov}[\eta(\mathbf{s}_i), \eta(\mathbf{s}_j)] = \sigma_\eta^2 r_\eta(\mathbf{s}_i, \mathbf{s}_j) \quad (4)$$

where $\mathbf{s}_i = (u_i, v_i)^T$ and $r_\eta(\mathbf{s}_i, \mathbf{s}_j)$ is the correlation function modeling the dependence between two distinct points \mathbf{s}_i and \mathbf{s}_j as a function of their distance. This type of correlation structure is useful if points closed in the space have similar values and hence a good prediction at a given location can be done by looking at its neighborhood. In this paper the power exponential distance is used

$$d(\mathbf{s}_i, \mathbf{s}_j) = \vartheta_1 |s_{i1} - s_{j1}|^{p_1} + \vartheta_2 |s_{i2} - s_{j2}|^{p_2} \quad 0 < p_k \leq 2 \quad \forall k = 1, 2. \quad (5)$$

When the scale parameters, ϑ_1 and ϑ_2 , have the same value the correlation is called isotropic, i.e. it does not depend on the specific spatial direction; otherwise the correlation is called anisotropic. The prediction at any new point \mathbf{s}_0 can be computed using the best linear unbiased predictor (BLUP) estimator (see for example Santner et al. [15] or Shabenberger and Gotway [16]), defined as

$$\hat{z}_i(\mathbf{s}_0) = \mathbf{f}_i^T(\mathbf{s}_0)\boldsymbol{\beta} + \sigma_\eta^2 \mathbf{r}_\eta^T \boldsymbol{\Sigma}_i^{-1} (z_i - \mathbf{F}_i \boldsymbol{\beta}) \quad (6)$$

where $\mathbf{r}_\eta = (r_\eta(\mathbf{s}_1, \mathbf{s}_0), r_\eta(\mathbf{s}_2, \mathbf{s}_0), \dots, r_\eta(\mathbf{s}_{n_i}, \mathbf{s}_0))$ is the correlation vector between the new point \mathbf{s}_0 and all the observed n_i points, $\boldsymbol{\Sigma}_i$ is the variance-covariance matrix of the Lo-Fi points and $\mathbf{F}_i = [\mathbf{f}_i(\mathbf{s}_1), \mathbf{f}_i(\mathbf{s}_2), \dots, \mathbf{f}_i(\mathbf{s}_{n_i})]^T \in \mathbb{R}^{n_i \times r}$ represent the matrix of regressors. The prediction variance can be computed as

$$\hat{\sigma}_i^2(\mathbf{s}_0) = \sigma_\eta^2 - \sigma_\eta^4 \mathbf{r}_\eta^T \boldsymbol{\Sigma}_i^{-1} \mathbf{r}_\eta + (\mathbf{f}_i(\mathbf{s}_0) - \mathbf{F}_i^T \boldsymbol{\Sigma}_i^{-1} \mathbf{r}_\eta)^T (\mathbf{F}_i^T \boldsymbol{\Sigma}_i^{-1} \mathbf{F}_i)^{-1} (\mathbf{f}_i(\mathbf{s}_0) - \mathbf{F}_i^T \boldsymbol{\Sigma}_i^{-1} \mathbf{r}_\eta). \quad (7)$$

The objective of a data fusion model is to combine the Lo-Fi and Hi-Fi data in order to improve predictions achievable by using the Lo-Fi or the Hi-Fi data sets alone. The core of the data fusion model is a linkage or second-stage model, which represents the connection between Lo-Fi and Hi-Fi data and can be expressed as [7,14]

$$z_h(\mathbf{s}_i) = \hat{z}_i(\mathbf{s}_i) + \delta_0 + \delta(\mathbf{s}_i) + \varepsilon_\Delta(\mathbf{s}_i) \quad i = 1, \dots, n_h. \quad (8)$$

The aim is to “correct” the Lo-Fi predictions $\hat{z}_i(\mathbf{s}_i)$ using a “shift” effects, represented by $\delta_0 + \delta(\mathbf{s}_i)$, with $\delta(\mathbf{s}_i) \sim \mathcal{GP}(0, \sigma_\delta^2, \vartheta_\Delta)$. Combining the linkage model in Equation (8) and first-stage model in Equation (3), the model obtained via data fusion allows one to predict a process realization at each new location \mathbf{s}_0 as:

$$\hat{z}_h(\mathbf{s}_0) = \hat{z}_l(\mathbf{s}_0) + \hat{\delta}_0 + (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} (\mathbf{z}_h - \hat{z}_l - \hat{\delta}_0 \mathbf{1}_{n_h}) \quad (9)$$

where

$$\hat{\delta}_0 = \frac{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} (\mathbf{z}_h - \hat{z}_l)}{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}}, \quad (10)$$

$\mathbf{r}_\delta = (r_\delta(\mathbf{s}_1, \mathbf{s}_0), r_\delta(\mathbf{s}_2, \mathbf{s}_0), \dots, r_\delta(\mathbf{s}_{n_h}, \mathbf{s}_0))$ is the correlation vector between the new point \mathbf{s}_0 and all the other n_h points, σ_0 is a vector with entries $\sigma_{0i} = \text{Cov}(\hat{z}_l(\mathbf{s}_i), \hat{z}_l(\mathbf{s}_0)) \forall i = 1, \dots, n_h$ and Σ_Δ is the variance-covariance matrix of the linkage model, computed as

$$\Sigma_\Delta = \Sigma_0 + \sigma_\delta^2 \mathbf{R}_\delta + \sigma_\Delta^2 \mathbf{I}_{n_h} \quad (11)$$

where Σ_0 is the kriging variance of the Lo-Fi model (given by the first stage equation), but computed at the Hi-Fi locations and \mathbf{R}_δ is the GP variance-covariance matrix.

While the variance of prediction, or kriging variance, can be computed as:

$$\hat{\sigma}_\Delta^2(\mathbf{s}_0) = \sigma_\delta^2 + \hat{\sigma}_l^2(\mathbf{s}_0) - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0) + \frac{[1 - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}]^2}{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}}. \quad (12)$$

Further details about the parameters estimation and the computation of the prediction variance can be found in [Appendix A](#).

2.2. Least squares B-splines approximation fusion model

The least squares B-splines approximation (LSBA) is an algorithm that approximate the observed data using as basis functions using uniform cubic B-splines. The LSBA algorithm is first present, then the fusion model is analyzed.

The evaluation of a point of the surface, at a spacial location \mathbf{s} , can be computed as

$$z(\mathbf{s}) = z(u, v) = \sum_{k=0}^3 \sum_{l=0}^3 B_k(r) B_l(t) \phi_{(i+k)(j+l)} \quad (13)$$

where $\phi_{(i+k)(j+l)}$ are the coefficients (control points) to be estimated, $i = \lfloor u \rfloor - 1$, $j = \lfloor v \rfloor - 1$, $r = u - \lfloor u \rfloor$, $t = v - \lfloor v \rfloor$ and B are uniform cubic B-spline defined as

$$\begin{aligned} B_0(t) &= \frac{(1-t)^3}{6} \\ B_1(t) &= \frac{3t^3 - 6t^2 + 4}{6} \\ B_2(t) &= \frac{-3t^3 + 3t^2 + 3t + 1}{6} \\ B_3(t) &= \frac{t^3}{6} \end{aligned} \quad (14)$$

with $0 \leq t < 1$, where $\lfloor \cdot \rfloor$ represents the floor operator.

Since the basis B-splines surfaces are described through the cross product of two B-splines basis (see Equation (13)) the control points ϕ_{ij} are defined over a rectangular lattice domain. The control lattice of the model, Φ_k , is shown in [Fig. 1](#), it is defined on a rectangular domain and $\phi_{i,j}$ the value of the ij -th control point at (i, j) for $i = -1, 0, \dots, m + 1$ and $j = -1, 0, \dots, n + 1$.

In order to estimate the model parameters, the point evaluation must be rewritten in matrix notation as

$$\mathbf{z} = \mathbf{F} \boldsymbol{\beta} + \boldsymbol{\varepsilon} \quad (15)$$

where $\mathbf{z} \in \mathbb{R}^n$ is the vector of the observed data, $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of coefficients, $\mathbf{F} \in \mathbb{R}^{n \times p}$ is the sparse matrix of B-splines coefficients

$$\mathbf{F} = \begin{bmatrix} B_1(u_1, v_1) & \dots & B_p(u_1, v_1) \\ B_1(u_2, v_2) & \dots & B_p(u_2, v_2) \\ \vdots & & \vdots \\ B_1(u_n, v_n) & \dots & B_p(u_n, v_n) \end{bmatrix}$$

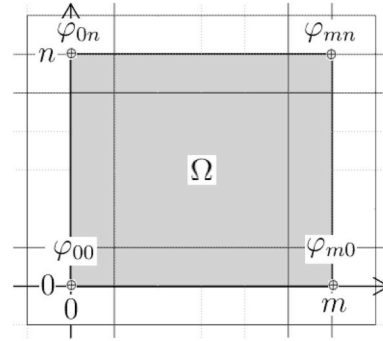


Fig. 1. Lattice control scheme on a rectangular domain.

where B correspond to the B-splines basis and $\boldsymbol{\varepsilon}$ is the vector the measurement errors, which are assumed to be independent and identically distributed normal random variables. As it is possible to observe from Equation (14) only 16 coefficients are not zero, so the matrix \mathbf{F} is extremely sparse. The unknown coefficients can be computed minimizing the function

$$\begin{aligned} \text{SSE} &= \sum_{i=1}^n (z_i - \hat{z}_i)^2 + \lambda \boldsymbol{\beta}^T \mathbf{E} \boldsymbol{\beta} \\ &= (\mathbf{z} - \mathbf{F} \boldsymbol{\beta})^T (\mathbf{z} - \mathbf{F} \boldsymbol{\beta}) + \lambda \boldsymbol{\beta}^T \mathbf{E} \boldsymbol{\beta} \end{aligned}$$

where z_i and \hat{z}_i are the value observed and predicted at location \mathbf{s}_i while the second term ($\lambda \geq 0$) is a smoothing penalty term, where $\mathbf{E} \in \mathbb{R}^{n \times n}$ is a symmetric sparse matrix whose entries are

$$e_{ij} = \iint \nabla B_i(u, v) \cdot \nabla B_j(u, v) \, dudv$$

where ∇ is the divergence operator and \cdot is the vector scalar product.

The minimum value of (16) respect to $\boldsymbol{\beta}$ can be found by solving the sparse linear system

$$(\mathbf{F}^T \mathbf{F} + \lambda \mathbf{E}) \boldsymbol{\beta} = \mathbf{F}^T \mathbf{z}. \quad (16)$$

To solve the sparse system in Equation (16) the sparse Cholesky decomposition implemented in the Eigen library [17] was used.

As suggested in Hjelle and Daehlen [18], λ is chosen between the multiples of

$$\lambda_d = \frac{\|\mathbf{F}^T \mathbf{F}\|_F}{\|\mathbf{E}\|_F}$$

where $\|\cdot\|_F$ is the Frobenius matrix norm.

As for the GP model the LSBA algorithm is firstly applied to the Lo-Fi data at the first stage. The observed Lo-Fi data can be described by

$$\mathbf{z}_l = \mathbf{F}_l \boldsymbol{\beta}_l + \boldsymbol{\varepsilon}_l \quad (17)$$

where \mathbf{z}_l are the observed Lo-Fi data, \mathbf{F}_l is the model matrix of the B-splines basis, $\boldsymbol{\beta}_l$ is the vector B-splines coefficients and $\boldsymbol{\varepsilon}_l$ is a vector of i.i.d. normally distributed random variables. The LS estimation of $\boldsymbol{\beta}_l$ is

$$\hat{\boldsymbol{\beta}}_l = (\mathbf{F}_l^T \mathbf{F}_l + \lambda_l \mathbf{E}_l)^{-1} \mathbf{F}_l^T \mathbf{z}_l. \quad (18)$$

The estimation of its variance-covariance matrix is

$$\hat{\Sigma}_{\boldsymbol{\beta}_l} = \hat{\sigma}_l^2 (\mathbf{F}_l^T \mathbf{F}_l + \lambda_l \mathbf{E}_l)^{-1} \mathbf{F}_l^T \mathbf{F}_l (\mathbf{F}_l^T \mathbf{F}_l + \lambda_l \mathbf{E}_l)^{-1} \quad (19)$$

Eventually, the estimate of σ_l^2 required in the last equation can be given by

$$\hat{\sigma}_l^2 = \frac{\|\mathbf{z}_l - \hat{\mathbf{z}}_l\|}{n_l - \text{tr}[\mathbf{F}_l (\mathbf{F}_l^T \mathbf{F}_l + \lambda_l \mathbf{E}_l)^{-1} \mathbf{F}_l^T]} \quad (20)$$

or (approximately) by

$$\hat{\sigma}_l^2 = \frac{\|\mathbf{z}_l - \hat{\mathbf{z}}_l\|}{n_l} \quad (21)$$

to avoid the computation of the solution of the linear system

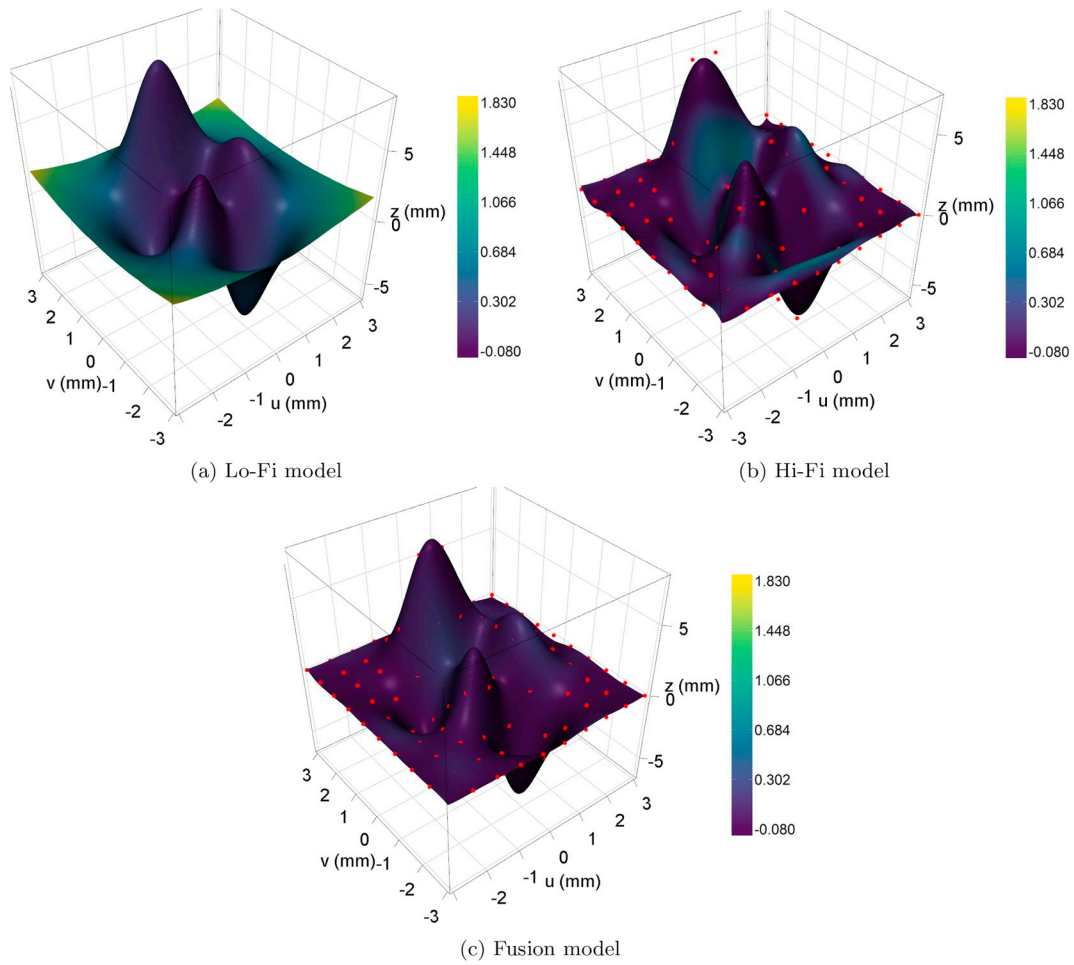


Fig. 2. Predicted surface with error map.

$(F_l^T F_l + \lambda_l E_l)^{-1} F_l^T$ if the number of measured points is high.

The variance of the prediction in a new location s_0 can be computed as

$$\mathbb{V}ar [\hat{z}_l(s_0)] = \mathbb{V}ar [f_l^T(s_0)\hat{\beta}_l] = f_l^T(s_0)\Sigma_{\hat{\beta}_l}f_l(s_0). \quad (22)$$

It should be noted that, since the inverse of $F_l^T F_l + \lambda_l E_l$ is not explicitly computed a linear system has to be solved for each new location.

The observed Hi-Fi points can be described as

$$\mathbf{z}_h = \hat{\mathbf{z}}_l + F_\Delta \hat{\beta}_\Delta + \varepsilon_\Delta. \quad (23)$$

Since the prediction with the Lo-Fi model has prediction variance that is a function of the spatial location (u, v) this information can be taken into account assigning a weight to each prediction. The weight correspond to inverse of the prediction variance, i.e. a smaller importance is given to predictions with higher variability. The variance of the residual process, $\mathbf{z}_\Delta = \mathbf{z}_h - \hat{\mathbf{z}}_l$, is a multiple of the variance of the predicted values with the Lo-Fi model

$$Cov(\mathbf{z}_\Delta) = Cov(\varepsilon_\Delta) = Cov(\mathbf{z}_h - \hat{\mathbf{z}}_l) = \sigma_\Delta^2 \mathbf{W}_\Delta^{-1}, \quad (24)$$

the variance of the observed Hi-Fi data become

$$Cov(\mathbf{z}_h) = Cov(\hat{\mathbf{z}}_l) + Cov(\mathbf{z}_\Delta) = \mathbf{W}_\Delta^{-1} + \sigma_\Delta^2 \mathbf{W}_\Delta^{-1}. \quad (25)$$

The variance of the fusion model is then bigger than the variance of $\hat{\mathbf{z}}_l$ because is the sum of two independent random vectors. The parameters can be estimated with the weighted least squares (WLS) procedure, the estimator of $\hat{\beta}_\Delta$ is

$$\begin{aligned} \hat{\beta}_\Delta &= (F_\Delta^T \mathbf{W}_\Delta F_\Delta + \lambda_\Delta E_\Delta)^{-1} F_\Delta^T \mathbf{W}_\Delta \mathbf{z}_\Delta \\ &= Q_\Delta \mathbf{z}_h - Q_\Delta F_l(S_h) Q_l \mathbf{y}_l \end{aligned} \quad (26)$$

where

$$Q_\Delta = (F_\Delta^T \mathbf{W}_\Delta F_\Delta + \lambda_\Delta E_\Delta)^{-1} F_\Delta^T \mathbf{W}_\Delta$$

$$Q_l = (F_l^T F_l + \lambda_l E_l)^{-1} F_l^T.$$

The prediction in a new location s_0 can be computed as

$$\begin{aligned} \hat{z}_h(s_0) &= \hat{z}_l(s_0) + \hat{z}_\Delta(s_0) = f_l^T(s_0)\hat{\beta}_l + f_\Delta^T(s_0)\hat{\beta}_\Delta \\ &= f_l^T(s_0)Q_l \mathbf{z}_l + f_\Delta^T(s_0)Q_\Delta(\mathbf{z}_h - \hat{\mathbf{z}}_l) \\ &= (f_l^T(s_0) - f_\Delta^T(s_0)Q_\Delta F_l(S_h))Q_l \mathbf{z}_l + f_\Delta^T(s_0)Q_\Delta \mathbf{z}_h \end{aligned}$$

and its variance is

$$\begin{aligned} \mathbb{V}ar [\hat{z}_h(s_0)] &= \mathbb{V}ar [(f_l^T(s_0) - f_\Delta^T(s_0)Q_\Delta F_l(S_h))Q_l \mathbf{z}_l + f_\Delta^T(s_0)Q_\Delta \mathbf{z}_h] \\ &= \mathbb{V}ar [(f_l^T(s_0) - f_\Delta^T(s_0)Q_\Delta F_l(S_h))Q_l \mathbf{z}_l] + \mathbb{V}ar [f_\Delta^T(s_0)Q_\Delta \mathbf{z}_h] \\ &\quad + 2 Cov [f_\Delta^T(s_0)Q_\Delta \mathbf{z}_h, (f_l^T(s_0) - f_\Delta^T(s_0)Q_\Delta F_l(S_h))Q_l \mathbf{z}_l] \\ &= \sigma_l^2 (f_l^T(s_0) - f_\Delta^T(s_0)Q_\Delta F_l(S_h))Q_l Q_l^T (f_l(s_0) - F_l^T(S_h)Q_\Delta^T f_\Delta(s_0)) \\ &\quad + f_\Delta^T(s_0)Q_\Delta (\mathbf{W}_\Delta^{-1} + \sigma_\Delta^2 \mathbf{W}_\Delta^{-1}) Q_\Delta^T f_\Delta(s_0) \\ &\quad + 2 \sigma_\Delta^2 f_\Delta^T(s_0)Q_\Delta F_l(S_h)Q_l Q_l^T (f_l(s_0) - F_l^T(S_h)Q_\Delta^T f_\Delta(s_0)) \end{aligned} \quad (27)$$

where

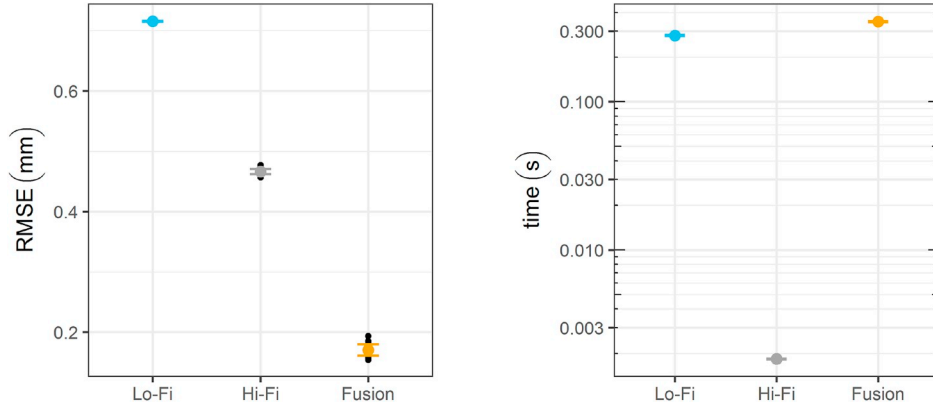


Fig. 3. Models' RMSE

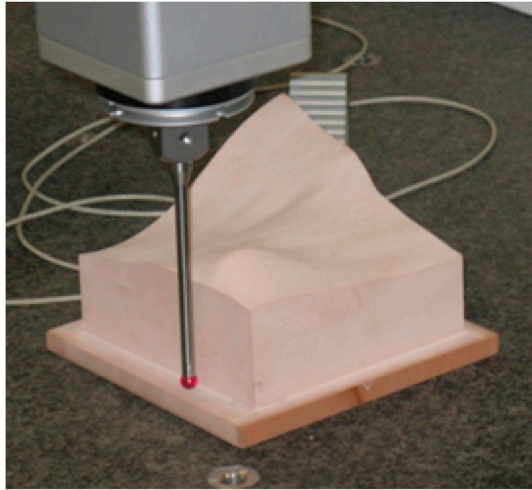


Fig. 4. Free form surface.

$$\begin{aligned}
 & \text{Cov} [\mathbf{f}_\Delta^T(s_0) \mathbf{Q}_\Delta \mathbf{z}_h, (\mathbf{f}_l^T(s_0) - \mathbf{f}_\Delta^T(s_0) \mathbf{Q}_\Delta \mathbf{F}_l(S_h)) \mathbf{Q}_l \mathbf{z}_l] \\
 &= \mathbf{f}_\Delta^T(s_0) \mathbf{Q}_\Delta \text{Cov} [\mathbf{F}_l(S_h) \mathbf{Q}_l \mathbf{z}_l + \mathbf{F}_\Delta \boldsymbol{\beta}_\Delta + \boldsymbol{\varepsilon}_\Delta, \mathbf{z}_l] \mathbf{Q}_l^T \\
 & \quad (\mathbf{f}_l^T(s_0) - \mathbf{F}_l^T(S_h) \mathbf{Q}_\Delta^T \mathbf{f}_\Delta(s_0)) \\
 &= \sigma_l^2 \mathbf{f}_\Delta^T(s_0) \mathbf{Q}_\Delta \mathbf{F}_l(S_h) \mathbf{Q}_l \mathbf{Q}_l^T (\mathbf{f}_l^T(s_0) - \mathbf{F}_l^T(S_h) \mathbf{Q}_\Delta^T \mathbf{f}_\Delta(s_0))
 \end{aligned}$$

because \mathbf{z}_l and $\boldsymbol{\varepsilon}_\Delta$ are independent. If it is possible to assume that the variance of the difference between the observed Hi-Fi data and the prediction of the Lo-Fi model is constant, i.e. $\text{Cov}(\mathbf{z}_h) = (\bar{\sigma}_l^2 + \sigma_\Delta^2) \mathbf{I}_{n_h}$ the parameters can be estimated using a LS algorithm. This approximation may be useful in a big data scenario because the computation of the values in Equation (22) can lead to high computational time. In order to

include an approximation of the variance of the predicted Lo-Fi points the prediction variance can be computed as

$$\begin{aligned}
 \text{Var} [\hat{\mathbf{z}}_h(s_0)] &= \sigma_l^2 (\mathbf{f}_l^T(s_0) - \mathbf{f}_\Delta^T(s_0) \mathbf{Q}_\Delta \mathbf{F}_l(S_h)) \mathbf{Q}_l \\
 & \quad \mathbf{Q}_l^T (\mathbf{f}_l^T(s_0) - \mathbf{F}_l^T(S_h) \mathbf{Q}_\Delta^T \mathbf{f}_\Delta(s_0)) \\
 & \quad + \mathbf{f}_\Delta^T(s_0) \mathbf{Q}_\Delta (\mathbf{V}_l + \sigma_\Delta^2 \mathbf{I}_{n_h}) \mathbf{Q}_\Delta^T \mathbf{f}_\Delta(s_0) \\
 & + 2 \sigma_l^2 \mathbf{f}_\Delta^T(s_0) \mathbf{Q}_\Delta \mathbf{F}_l(S_h) \mathbf{Q}_l \mathbf{Q}_l^T (\mathbf{f}_l^T(s_0) - \mathbf{F}_l^T(S_h) \mathbf{Q}_\Delta^T \mathbf{f}_\Delta(s_0))
 \end{aligned} \tag{28}$$

where

$$\mathbf{Q}_\Delta = (\mathbf{F}_\Delta^T \mathbf{F}_\Delta + \lambda_\Delta \mathbf{E}_\Delta)^{-1} \mathbf{F}_\Delta^T$$

and

$$\mathbf{V}_l = \text{diag}\{\hat{\sigma}_l^2\} \tag{29}$$

is a diagonal matrix with entry equal to the mean variance of the Lo-Fi model defined as

$$\hat{\sigma}_l^2 = \frac{1}{|S|} \int_S \hat{\sigma}_l^2(s) ds \tag{30}$$

where $\hat{\sigma}_l^2(s)$ is the predicted variance in the spatial location $\mathbf{s} = (u, v)$, S is the surface domain and $|S|$ is its area.

In order to speed up the computation it is assumed that the matrix \mathbf{W}_Δ is a diagonal matrix. The uncertainty of the Lo-Fi model is taken into account, but not the correlation between the predictions, the prediction variance the matrix \mathbf{W}_Δ become

$$\mathbf{W}_\Delta = \text{diag}\left\{ \frac{1}{\hat{\sigma}_l^2(s_i)} \right\}. \tag{31}$$

The data fusion model was coded in C++, the points evaluation and the variance computation was easily parallelized using OpenMP [19]. Although the computation of the prediction variance requires the solution of one (few) linear system for the Lo-Fi (fusion) model it is still

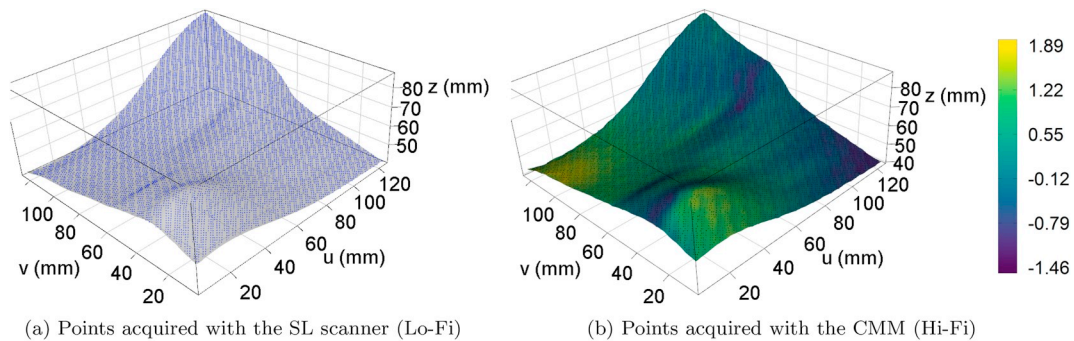


Fig. 5. Lo-Fi and Hi-Fi points analyzed.

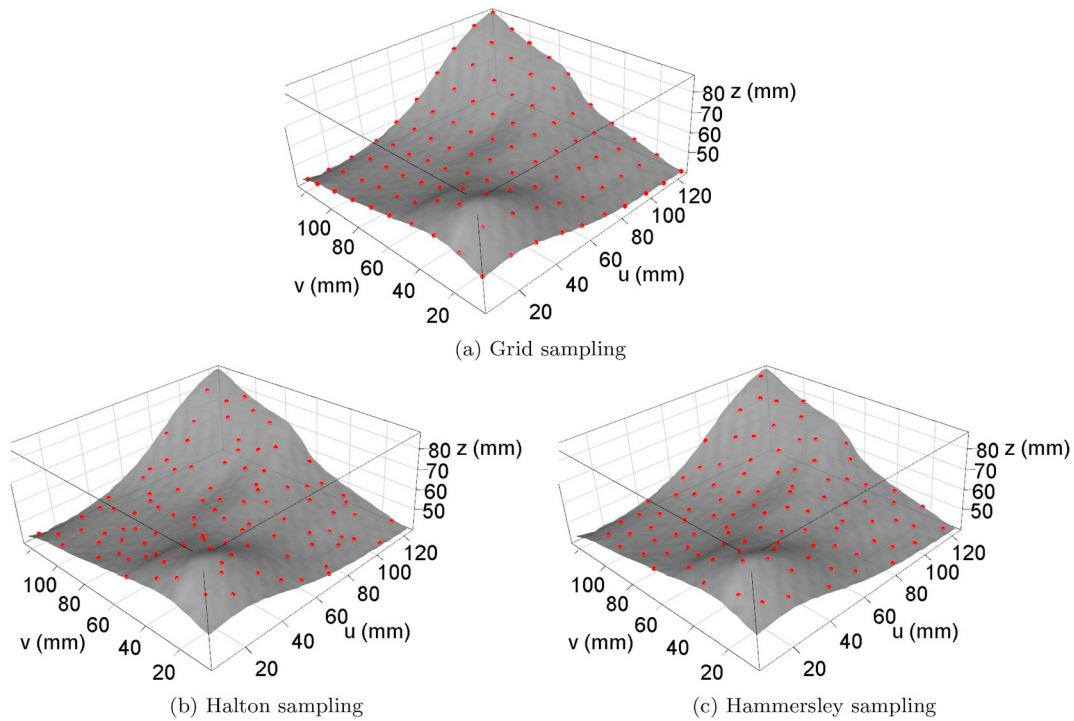


Fig. 6. Example of 100 Hi-Fi points sampled according to the analyzed sample strategies.

fast. For example the computation of the mean value and its variance of the Lo-Fi model takes about 1 s for 10000 points, for a model with a control grid of 30×30 control points, while it takes few seconds for the estimation of the prediction variance of a fusion model with a control grid of 10×10 control points. The computational times were recorder using a computer with an Intel(R) Core(TM) i7-5930K CPU @ 3.50 GHz processor.

2.2.1. Test case

In this Section the advantages of combining different data sets with the proposed hierarchical data fusion approach are investigated using some simulated data. To this aim, performance of three different models are compared:

- Lo-Fi model: the predictions are computed using a single model that considers the Lo-Fi data only;
- Hi-Fi model: the predictions are computed using a single model that considers the Hi-Fi data only;
- Fusion model: the predictions are based on the two-stage hierarchical model presented before.

The performances of the model estimated using both the Lo-Fi and Hi-Fi points as they both come from the same source, called addition, it is not investigated. As observed in Colosimo et al. [7], the addition model does not lead to smaller prediction if the Lo-Fi data have a systematic bias.

The performance indicator used throughout this paper is the root mean square (prediction) error (RMSE), defined as

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n e^2(s_i)} \tag{32}$$

where $e(s_i)$ is the prediction error at location s_i , given by

$$e(s_i) = z(s_i) - \hat{z}(s_i)$$

where $z(s_i)$ is the true point at location s_i and $\hat{z}(s_i)$ is the value predicted at the same location using one of the aforementioned approaches.

In order to show the prediction ability of the LSBA-based fusion model the Matlab peaks function was used. This function is a mixture of translated and scaled bivariate Gaussian probability density function. The peaks function is given by

$$f(s) = 3(1 - u)^2 e^{-u^2 - (v+1)^2} - 10 \left(\frac{u}{5} - u^3 - v^5 \right) e^{-u^2 - v^2} - \frac{1}{3} e^{-(u+1)^2 - v^2}. \tag{33}$$

The Hi-Fi data were simulated from the following model

$$z_h(s) = f(s) + \varepsilon_h, \quad \varepsilon_h = \mathcal{N}(0, 0.2^2). \tag{34}$$

while the Lo-Fi data model were randomly generated using the following equation:

$$z_l(s) = f(s) + \text{bias}(s) + \varepsilon_l, \quad \varepsilon_l = \mathcal{N}(0, 0.4^2) \tag{35}$$

where a bias term was added. This bias term was assumed to have the following model:

$$\text{bias}(s) = \frac{u^2}{10} + \frac{v^2}{10},$$

which describes a Lo-Fi measurement system where the bias increases as the u and v coordinates increase in module. This bias term gives evidence to the need of performing data fusion. As a matter of fact, the Hi-Fi data have to be used to correct the Lo-Fi bias where needed.

A total number of 250000 Lo-Fi points were generated from a regular grid of $[500 \times 500]$ points in u and v directions. Since the number of the available Hi-Fi points is usually lower than the Lo-Fi sample size, only 100 Hi-Fi points were generated on a regular grid of $[10 \times 10]$ points.

For the estimation of the Lo-Fi model a lattice of 30×30 control points and a smoothing parameter $\lambda = 1$ were used. A 10×10 control lattice and a smoothing parameters equal to 0.01 and 0.1 were used to estimate the Hi-Fi and fusion model respectively.

For each of the competitive models, an error map is computed as the difference between the predicted and the nominal value at each location ($e(s_i)$). The error maps are reported in Fig. 2. The shape of the error map of the Lo-Fi model shows that the simulated bias affected the accuracy of the predictions. The error map of the Hi-Fi model shows that

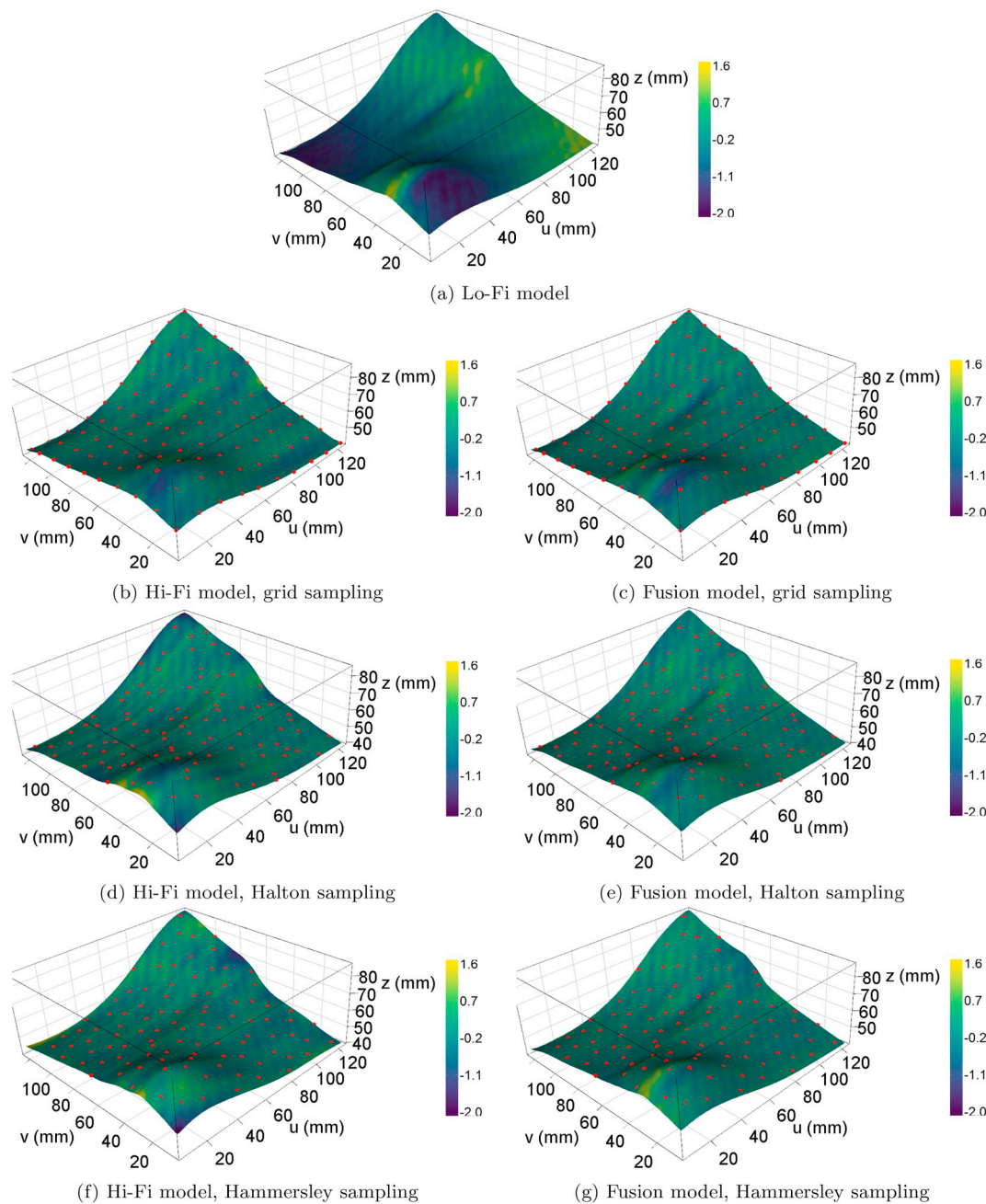


Fig. 7. Predicted surfaces and error map using the GP model.

the error is generally smaller (because those data are assumed to be characterized by zero bias). The error map of the fusion model shows that a hierarchical model can correct the error of the Lo-Fi model where needed, using only a small set of Hi-Fi data. It should be also noted that, since the fusion model has to estimate the difference between the prediction of the Lo-Fi model and the Hi-Fi data, a higher smoothing parameter, compared to the Hi-Fi model, can be set to avoid the ondulation in the flat portion of the surface.

Considering this peaks model as a toy example, 10 different realizations of the surface were run and the prediction ability of each model was tested using the RMSE.

The average time to estimate the parameters of the LSBA-based fusion model is 0.35s, so with this model is also possible to handle point clouds with a large amount of data.

The confidence interval on the mean of the RMSE computed on these 10 replications are drawn in Fig. 3. It is clear that the fusion

model outperforms all the other methods in this case study.

3. Case study: freeform surface reconstruction

In this Section the advantages of the data fusion model applied to the reconstruction of a real smooth freeform surface are investigated. In this case study, the performance of different models are computed while varying the number of Hi-Fi points. Furthermore, three different sampling plans to locate Hi-Fi points were analyzed: (i) a uniform sampling plan according to a classic regular grid; (ii) Halton [20] and (iii) Hammersley [21] sampling strategies which are widely used in metrology [22].

In their work, Petrò et al. [23] acquired the point clouds of the free form surface shown in Fig. 4 with both a Structured Light (SL) scanner (Lo-Fi) and a Coordinate Measuring Machine (CMM) Zeiss “Prismo 5 HTG VAST” equipped with a analog probe head with maximum probing

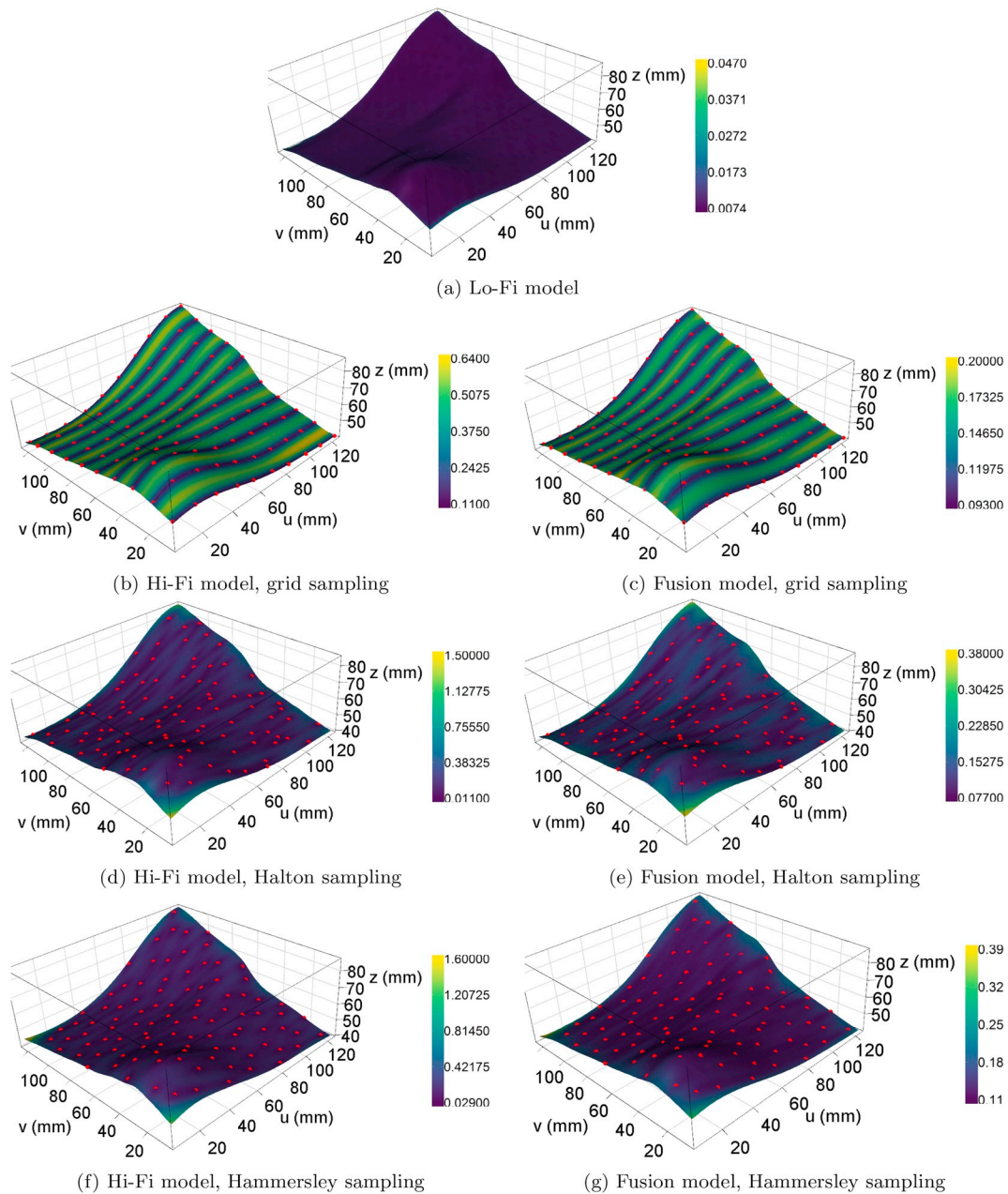


Fig. 8. Predicted surface and mean standard deviation using the GP model.

error MPEP = 2 μm (Hi-Fi).

This case study is used to compare performance of three competitive approaches:

- Lo-Fi: model based on the Lo-Fi (SL) points only;
- Hi-Fi: model based on the Hi-Fi (CMM) points only;
- Fusion: two-stage fusion model based on both the Lo-Fi and the Hi-Fi points.

The number of points acquired with each device was 9635. Clearly, acquiring this sample size via CMM required a long acquisition time, which is almost unfeasible in real practice. As a matter of fact, it is assumed that only a subset of these Hi-Fi (CMM) data were effectively available to perform data fusion, while the remaining set was used as test set to compute the prediction errors of all the competitive methods. In other words, $n_h = h\% \cdot 9635$ data were used to reconstruct the surface with all the proposed methods, while the remaining set of

$n_{test} = 9635 - n_h$ points was used to represent the “real” surface. Predictions at all the locations of n_{test} data were computed with all the competitive methods and the corresponding prediction errors ($e(s_i)$) will be computed.

The whole set of 9635 Lo-Fi and Hi-Fi points representing a smooth freeform surface are shown in Fig. 5. The color map in Fig. 5b represents the difference error of the Lo-Fi points, computed as the difference between the measured Hi-Fi and Lo-Fi points. The negative errors of the SL scanner are located near the peak, and on the right corner, while the positive errors are located on the left corner.

In order to compute the performance of each model as a function of the number of Hi-Fi points, the percentage h of the Hi-Fi data used to fit the surface models was changed (from $n_h = 25$ to a maximum value of 4% of the total amount, i.e., $n_h = 400$ Hi-Fi data points). The sampling of 100 Hi-Fi points, for each of the analyzed sampling strategy, are shown in Fig. 6. Since the “real” points were available only on a finite set of locations (the ones measured using the CMM) after computing the

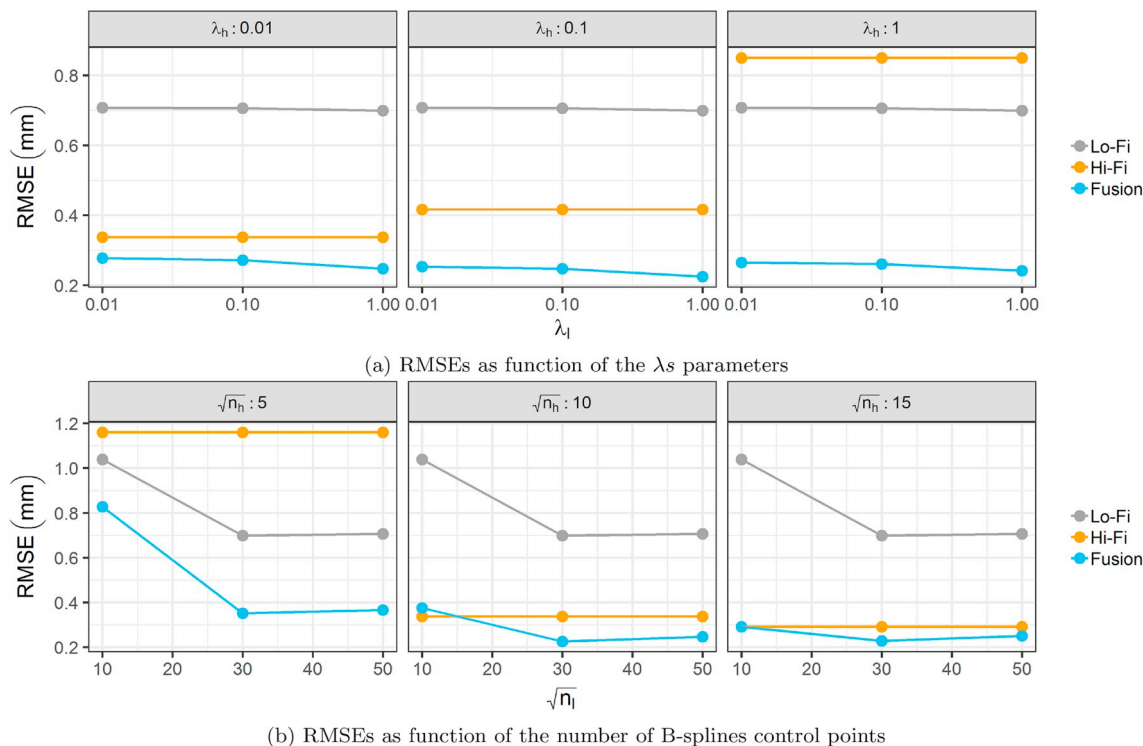


Fig. 9. RMSEs as function of the control lattices and the smoothing parameters.

theoretical locations the closest point, between the points not already included in the set, were selected as Hi-Fi point.

3.1. GP-based fusion model

The surfaces reconstructed with the GP model, using 100 H-Fi points, are shown in Fig. 7. The color represents the difference between the test point and the predicted one. The errors of the Lo-Fi model has an higher magnitude on the left of the surface. On the peak the error is positive, while on the bottom left it is negative. These discrepancies are systematic due to the SL measurement system. On the contrary the systematic error of both the Hi-Fi and the fusion models appear smaller, so the reconstruction result is better. The error map of the Hi-Fi model shows that, due to the small number of Hi-Fi points available, the model has high errors in some locations of the reconstructed surface. On the contrary, the fusion model uses the information of both the Lo-Fi model and the Hi-Fi points to better reconstruct the surface reducing the error magnitude everywhere. The major differences between the Hi-Fi and the fusion models can be observed along the border of the surface, where there aren't Hi-Fi points available, or in the portion of the surface where the distance between the Hi-Fi points is higher. The GP fusion model can therefore “correct” the prediction of the Lo-Fi model using the Hi-Fi points as “attractors”.

Fig. 8 shows the standard deviation of the predicted mean value. The uncertainty of the points predicted using the Lo-Fi model is lower compared to the other models because the parameters are estimated using a higher number of points. Since the number of Hi-Fi points is low, the uncertainty of the predictions of the Hi-Fi model is the highest. The standard deviation using a grid sampling shows a sinusoidal pattern. This is due to the effect of the anisotropic correlation function, since the estimated correlation along the u direction is higher, the prediction uncertainty is lower. The uncertainty of the fusion model has the same behaviour of the Hi-Fi model one, but with a lower magnitude.

3.2. LSBA-based fusion model

In order to use the LSBA algorithm, the first step is to choose the three parameters that characterize the model: the number of control points in u and v direction and the smoothing factor λ . For the fusion model the number of parameters to set is double because there are two different models involved. The RMSE values were computed using the whole set of Lo-Fi data and 100 Hi-Fi points sampled according to a uniform grid. The values of λ explored during the experiments were 0.01, 0.1 and 1, while the control lattice were 10×10 , 30×30 and 50×50 for the Lo-Fi model, and 5×5 , 10×10 and 15×15 for both the Hi-Fi and the fusion models. The experiment was run for all the 81 possible combinations of the parameters, in Fig. 9 are reported the RMSEs as a function of the smoothing parameters (with a control lattice of 30×30 for the Lo-Fi model and 10×10 control points for both the Hi-Fi and fusion models) and the control lattices (with $\lambda_l = 1$, $\lambda_h = 0.01$ and $\lambda_\Delta = 0.1$) for the three models. Both the results of the Lo-Fi and fusion models are robust respect to the parameter λ , while a small parameter, 0.01, is preferable for the Hi-Fi model. The behaviour of the RMSE is similar for the three analyzed model: if few points are used, the accuracy of the model is not good.

The surface reconstructions using the LSBA algorithm and the reconstruction errors are shown in Fig. 10. The behavior of the error is comparable with the one of the GP models, but the error of the Hi-Fi models is higher if there are not points located on the boundary of the surface. The standard deviation of the mean prediction, show in Fig. 11, is lower compared to the GP models for all the model analyzed. Since the spatial correlation of the points is not taken into account the standard deviation of the Hi-Fi model has a check board effect, it is low where there are the measured points and higher in the other locations. Differently from the GP model, the behaviour of the standard deviation of the fusion model is similar to the one of the Lo-Fi model, i.e. the uncertainty due to the second stage model is low.

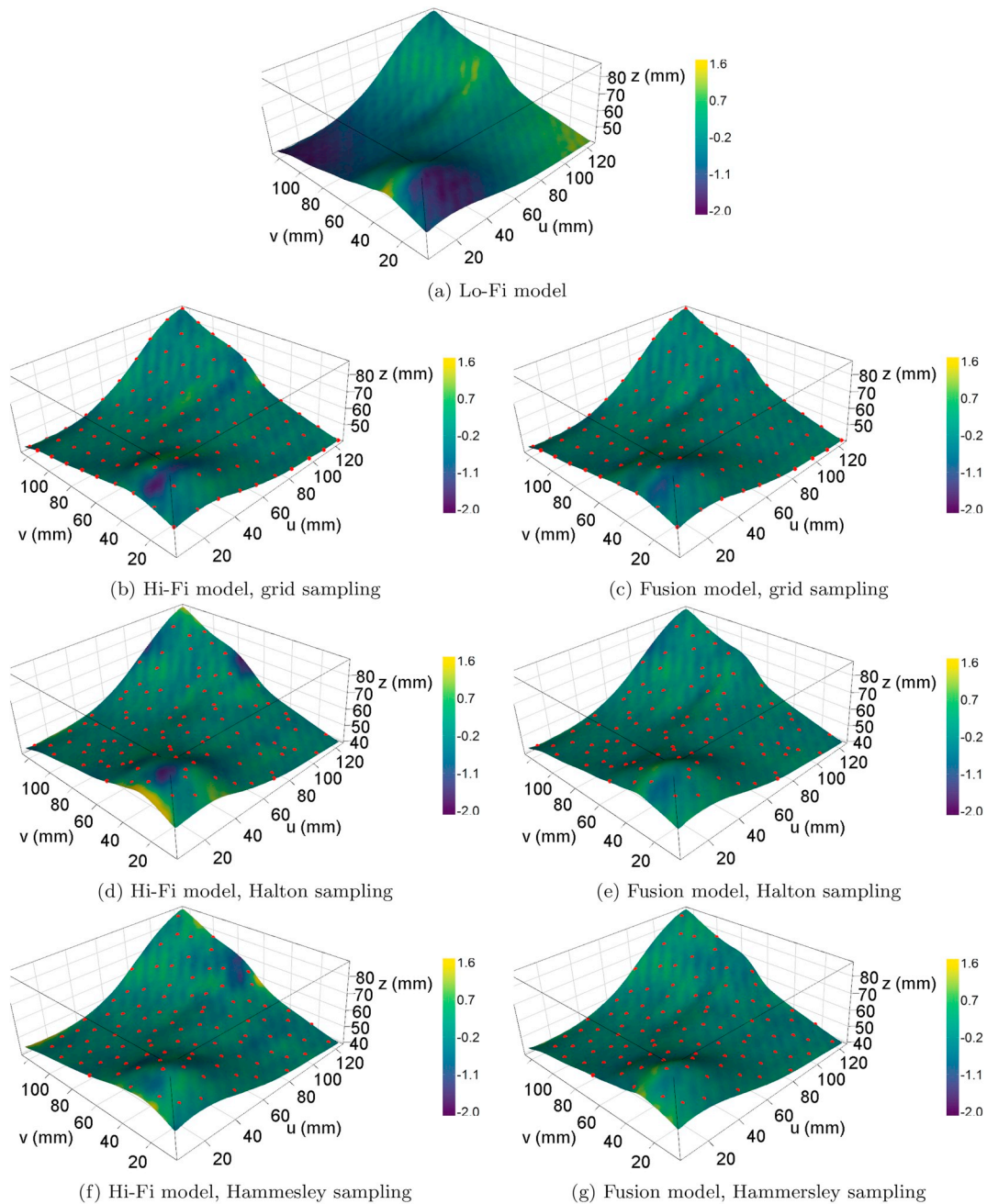


Fig. 10. Predicted surface and error map using the LSBA model.

3.3. Performance comparison

In this Section the reconstruction performances of the analyzed model are compared according to the RMSE metric. The smoothing parameters of the LSBA algorithm were $\lambda_l = 1$, $\lambda_h = 0.01$ and λ_Δ . A lattice grid of 30×30 control points for the Lo-Fi points, while a lattice equal to $\min\{\lfloor \sqrt{n_h} \rfloor, 30\} \times \{\lfloor \sqrt{n_h} \rfloor, 30\}$ for both the Hi-Fi and the fusion models. Fig. 12 shows the values of the computed RMSEs. The fusion models can improve the performances of the Lo-Fi data, but if the number of Hi-Fi points is high enough the RMSEs of the fusion and the Hi-Fi model are comparable. In Fig. 13 are reported the RMSEs of the LSBA and GP model for the three analyzed sampling strategies. Both the LSBA and GP models seems not be affected by the analyzed sampling

strategies, except the Hi-Fi model with 100 uniform sampled points. The performance of the LSBA algorithm is better if there are few Hi-Fi points, while if their number increase the RMSE values are similar.

Fig. 14 shows the computational times, in seconds, needed by the GP-based and the LSBA-based approaches as a function of the number of Hi-Fi points used. Since in the fusion model there are two stages, it takes longer compared to the Lo-Fi and Hi-Fi models. The slowest fitting for the LSBA model takes less than 1s, while the GP estimation is slower. The time to estimate the parameters of the Lo-Fi model takes approximately 1 h because a full matrix of approximately 9635×9635 has to be solved many times in order to estimate the parameters of the correlation function.

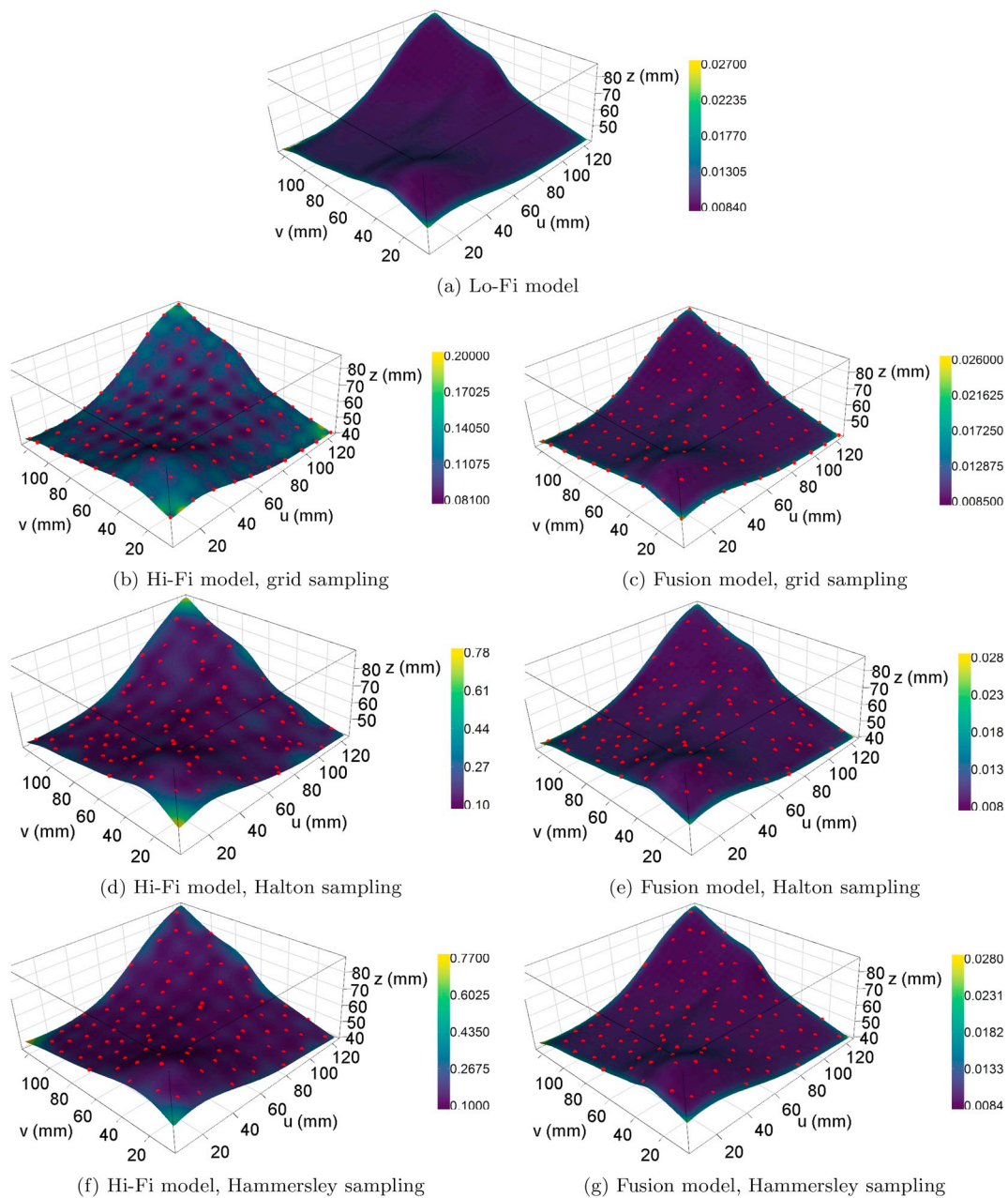


Fig. 11. Predicted surface and mean standard deviation using the LSBA model.

4. Case study: portion of impeller blade

A portion of an impeller blade was measured using both a structured light scanner and a CMM. The SL measurement was carried out using Creafom HandyScan 7000, with a $20 + 60 L/1000 \mu\text{m}$ uncertainty, L expressed in mm. Before the measurement, the blade was painted to matt and a set of anchor points was arranged in the exterior region of the measuring blade as the coordinate reference as shown in Fig. 15a. The painting was performed according to the ISO 3452 [24], the sensitivity of detection was equal or less than $0.5 \mu\text{m}$. The red-bounded area was then densely scanned with the sampling resolution of 0.2 mm (0.05 mm with the refinement algorithm). The employed CMM was a Hexagon Global Classis SR 05.07.05 with the composite uncertainty in $2.3 + 3 L/1000 \mu\text{m}$, L expressed in mm. The measuring probe was a

Renishaw SP25 with a 1.5 mm diameter ruby ball. Without using a CAD model, the blade was automatically measured by manually setting the initial scanning points, directions and scanning bounds as shown in Fig. 15b. Once the center coordinates of the ruby ball were obtained after scanning, the surface points were calculated by compensating a ball radius along the surface normal, which were fitted using a set of neighboring points.

A total number of 291 293 Lo-Fi (with the SL scanner) and 1616 Hi-Fi points (with the touching probe) were measured. The two point clouds were first aligned using the rigid iterative point closest algorithm implemented in CloudCompare [25]. The measured points are shown in Fig. 16 along with the estimated systematic error of the Lo-Fi data. Since, due the sampling size, is not possible to see the measured points, a decimated version is shown in Fig. 16b.

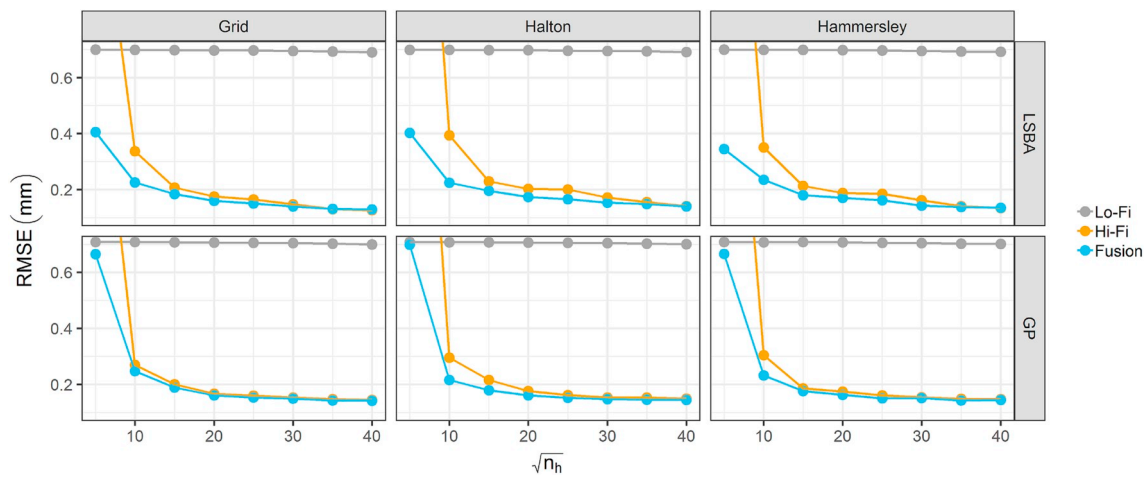


Fig. 12. Models' RMSEs.

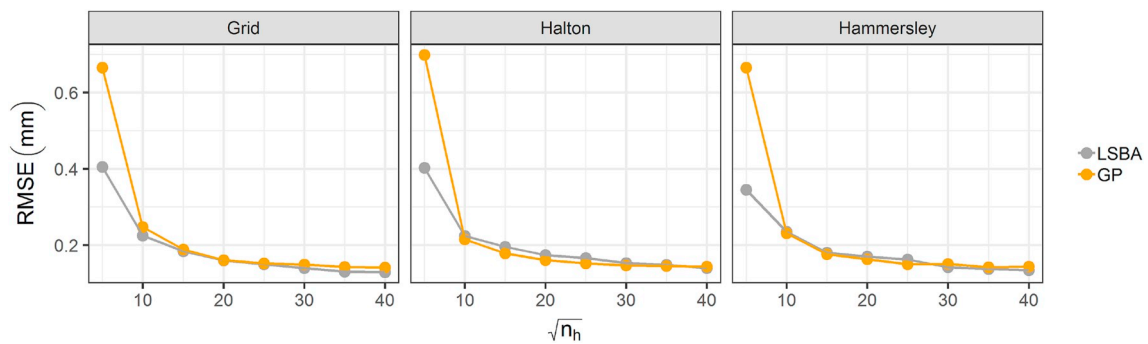


Fig. 13. Fusion models' RMSEs.

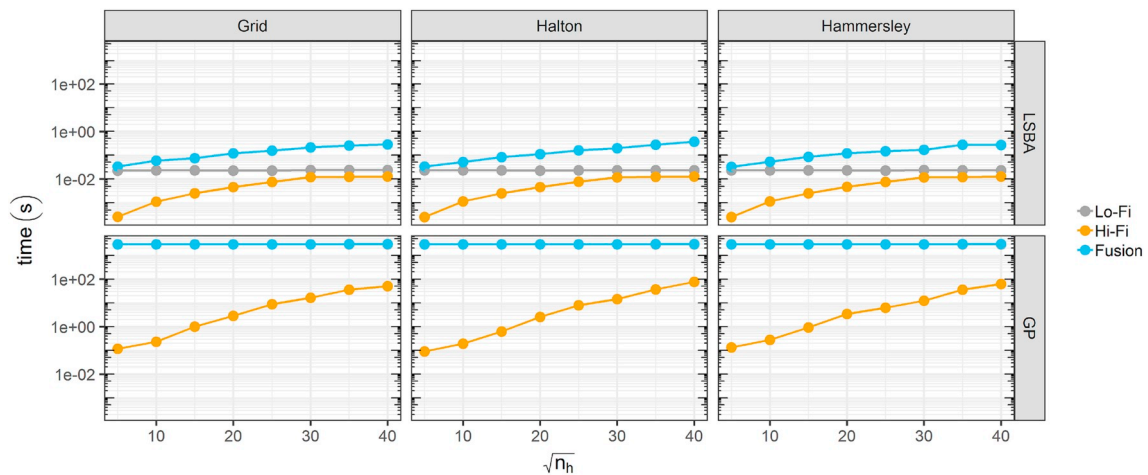


Fig. 14. Computation times of the GP and LSBA algorithm (axes in log scale).

In this second test case the systematic error appears to have non linear shape, it is positive in the top and on the bottom left portion of the surface and negative elsewhere. It is not possible to estimate the parameters using the GP because the number of the Lo-Fi points is too high, with a 64 GB RAM is not possible to compute the correlation matrix. To check the prediction performance of the GP model, the SL point cloud was decimated using a uniform sampling in the space with a minimum sampling distance distance of 0.5 mm. The decimated data, shown in Fig. 16b, has a total number of 3828 points.

In order to check the goodness of the analyzed models (Lo-Fi, Hi-Fi and fusion) a subset of Hi-Fi points was used to estimated the models' parameters, while the remaining points were used to compute the prediction error. A total number of 25, 100, 225 and 400 points were sub-sampled from the CMM data, using the grid, Halton and Hammersley samples strategies. For each sampling strategy the nominal points were first generated, the closest available Hi-Fi points were then selected to perform the analysis. Since the duplicate points were deleted, the total number of Hi-Fi points is smaller than the designed one.

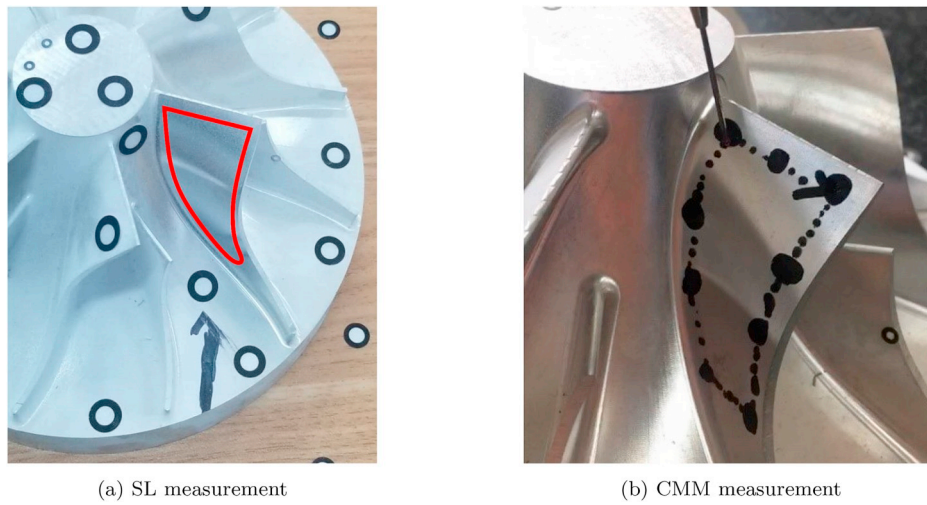


Fig. 15. Measured configurations.

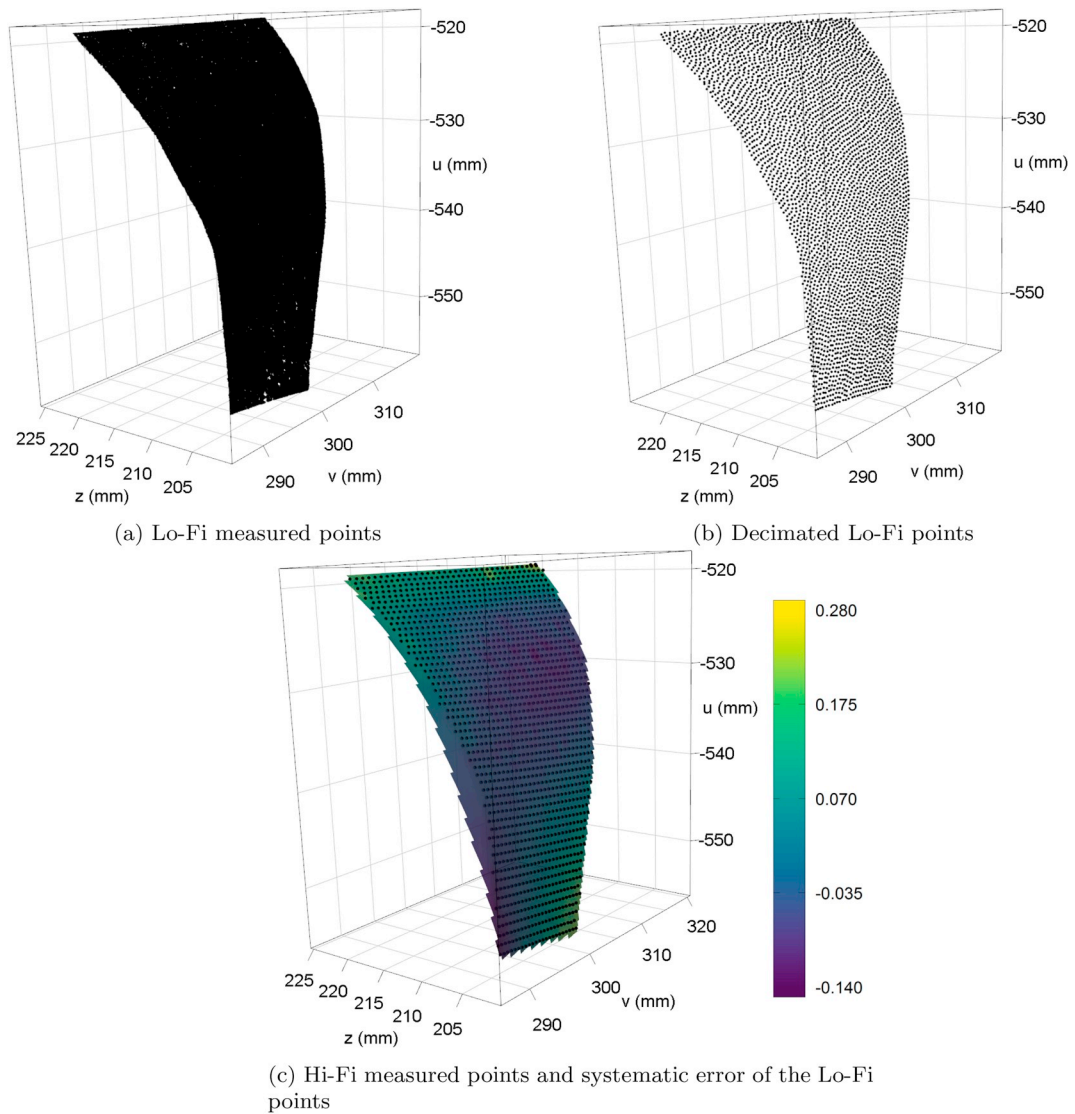


Fig. 16. Measured points and systematic error of the portion of impeller blade.

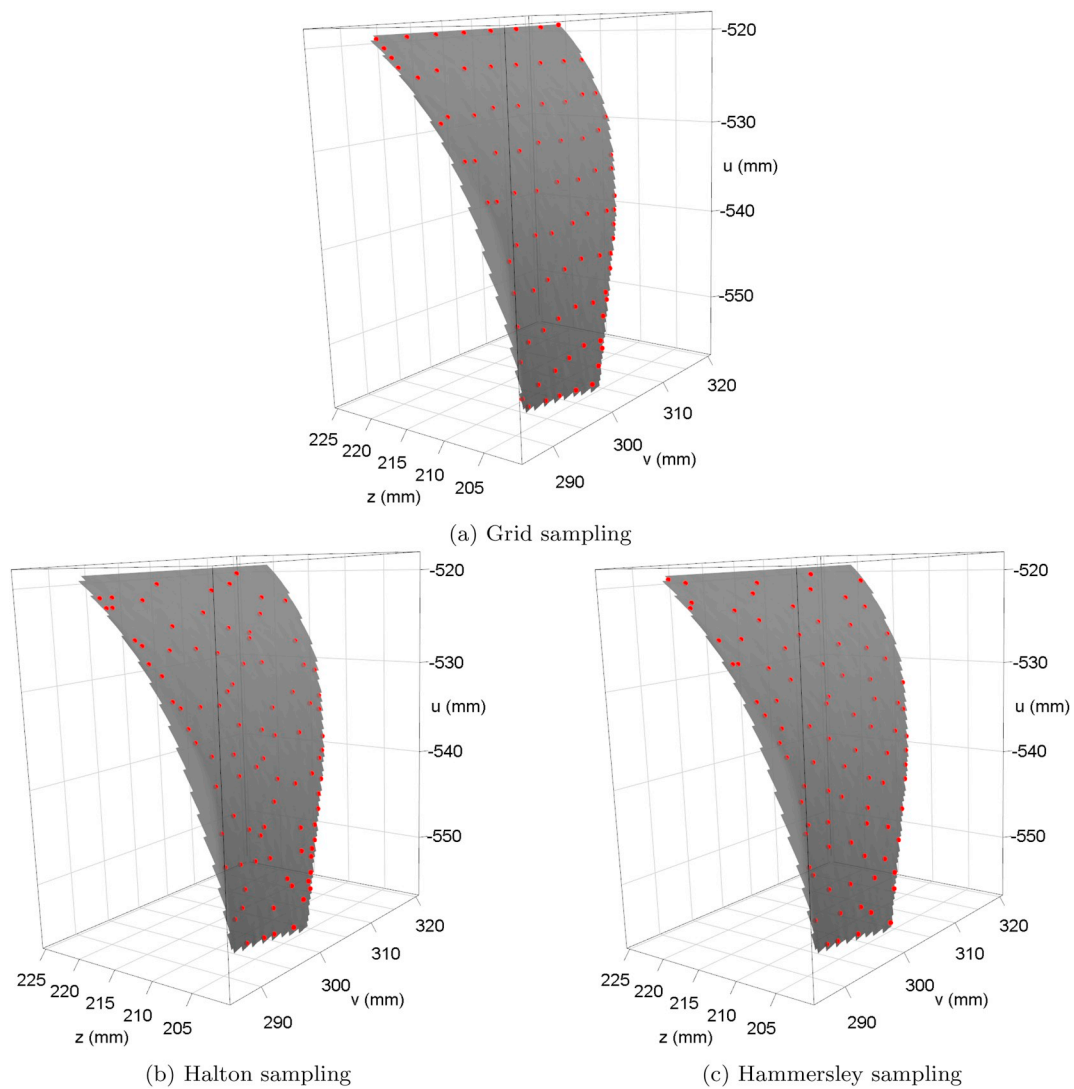


Fig. 17. Example of 100 Hi-Fi points sampled according to the analyzed sample strategies.

The analyzed Hi-Fi points (nominal size of 100 samples) are shown in Fig. 17. The RMSEs were then computed using the remaining Hi-Fi points inside the domain of the training points. Fig. 18 shows the results of the test case. A lattice grid of 30×30 control points and a λ parameter of 0.1 were used for the Lo-Fi model, while a grid of 10×10 control points and λ equal to 0.01 were set for the Hi-Fi and fusion models. The execution time of the Hi-Fi model is the lowest due to the limited number of measured points, but the best RMSE can be obtained using the fusion model. It should be noted that the estimation times of both the Lo-Fi and fusion model are fast, both the models can estimate the parameters in less than 1 s.

The results of using the decimated version of the Lo-Fi data are shown in Fig. 19. Due to the limited complexity of the analyzed mesh, it is possible to achieve the same prediction performances. As in the previous test case, the results of the GP and LSBA methods are comparable, but the time required to estimate the model parameters of the LSBA method is one order of magnitude faster.

5. Conclusions

In this paper a LSBA-based fusion model and a formula for the

propagation of the uncertainty of prediction from the Lo-Fi to the linkage model were presented. A formula to compute the prediction variance for the GP-based fusion model proposed in Ref. [7] was also proposed.

The advantages of the proposed LSBA-based fusion model were firstly explored with artificially generated data, then the prediction performances of the proposed model were then compared to the GP-based fusion model with a real test case. It was shown the RMSE of the two models are comparable, but the time required for the parameters' estimation of LSBA-based fusion model is orders of magnitude lower compared to the GP-based one. It was further observed that the uncertainty of the prediction of the LSBA model is lower compared to the GP one.

In the test case three sampling strategies were analyzed, the results shown that the sampling had no effect on the RMSE values. Other sampling plans based both on the curvature [26,27] of the Lo-Fi model or on the uncertainty prediction uncertainty [28] has to be investigated.

A second test case on real data showed that the proposed LSBA model can deal with a larger amount of points in a lower execution time compared to the GP model.

The proposed LSBA model uses a B-splines basis function that

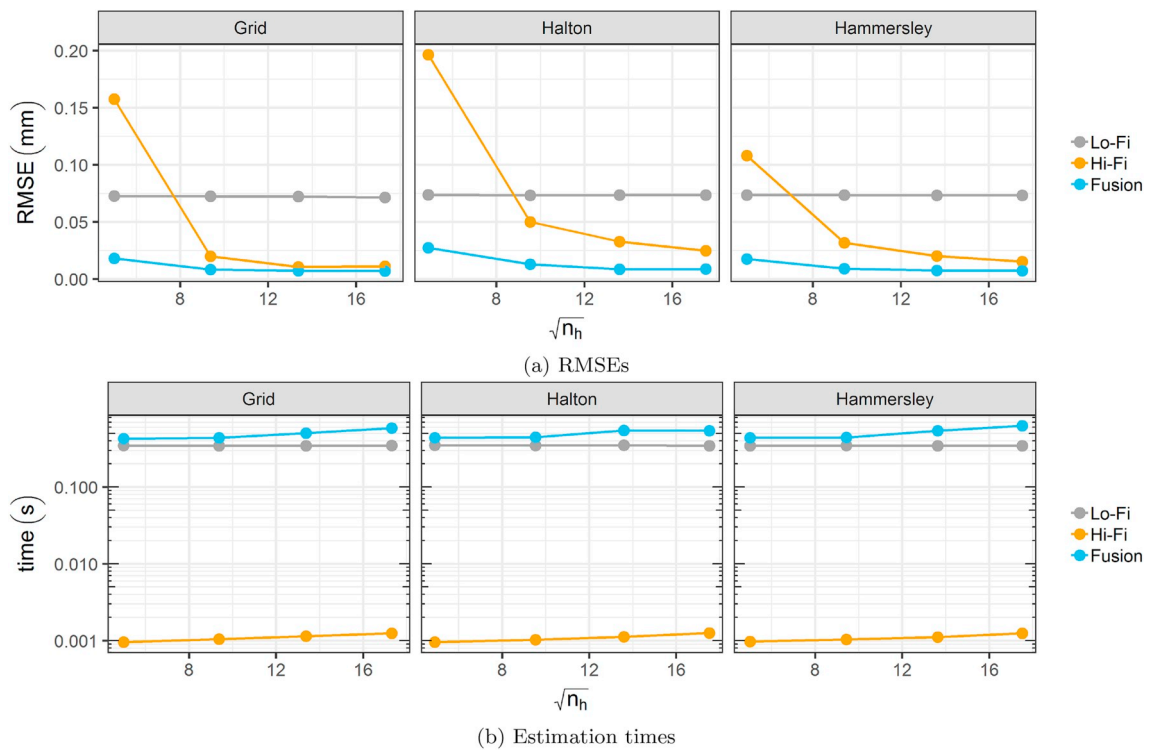


Fig. 18. RMSEs and estimation times of the portion of impeller blade.

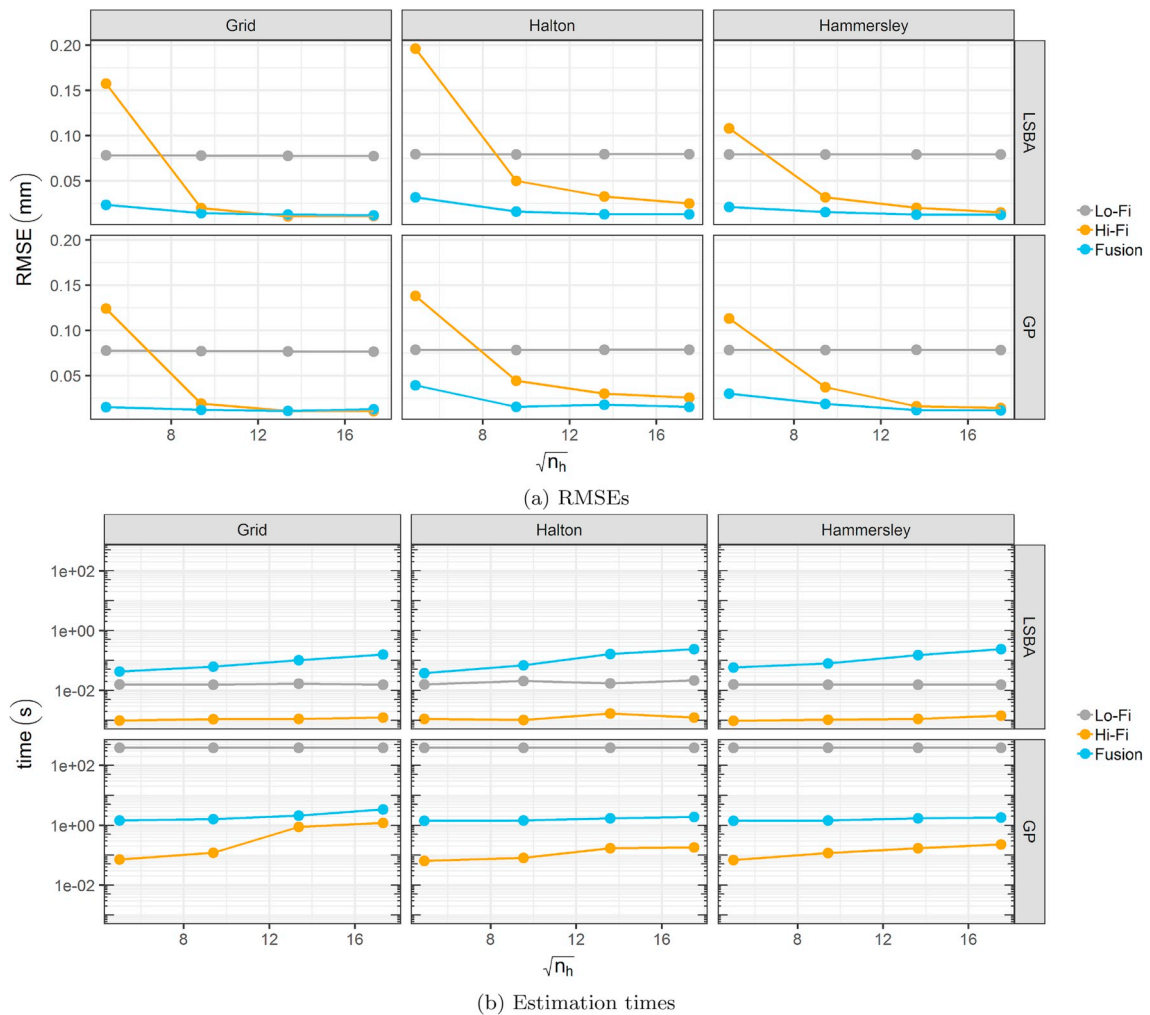


Fig. 19. RMSEs and estimation times of the portion of impeller blade (after the decimation of the Lo-Fi points).

correspond to the cross product of the two B-splines basis function in u and v directions, a data fusion model based on local B-splines function, such as the truncated hierarchical B-splines [29] or the locally refined B-splines [30] has to be developed.

The reconstruction of shapes from large data set can be performed by combining segmentation (to identify surface patches) and surface fitting via data fusion, new approaches for data fusion of multi-patches surfaces have to be developed.

Appendix A. Prediction of new Point and its Propriety

The prediction value and the uncertainty interval of a new observations at location \mathbf{s}_0 is now computed. As for the classic GP model, the starting point is the joint distribution between \mathbf{z}_h and $z_h(\mathbf{s}_0) = z_0$

$$\begin{pmatrix} \mathbf{z}_h \\ Z_0 \end{pmatrix} | \mathbf{z}_h, \mathbf{z}_h \sim \mathcal{N}_{n_h+1} \left(\begin{pmatrix} \hat{\mathbf{z}}_l + \delta_0 \mathbf{1}_{n_h} \\ \hat{z}_l(\mathbf{s}_0) + \delta_0 \end{pmatrix}, \begin{bmatrix} \Sigma_\Delta \sigma_\delta^2 \mathbf{r}_\delta + \sigma_0 & \\ \sigma_\delta^2 \mathbf{r}_\delta^T + \sigma_0^T \sigma_\delta^2 + \hat{\sigma}_l(\mathbf{s}_0) + \sigma_\Delta^2 \end{bmatrix} \right)$$

The conditional distribution of the new prediction is

$$Z_h(\mathbf{s}_0) | \mathbf{z}_h, \mathbf{z}_l \sim \mathcal{N}(\hat{z}_l(\mathbf{s}_0) + \delta_0 + (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} (\mathbf{z}_h - \hat{\mathbf{z}}_l - \delta_0 \mathbf{1}_{n_h}), \sigma_\delta^2 + \hat{\sigma}_l(\mathbf{s}_0) + \sigma_\Delta^2 - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)).$$

Since the real parameter values are unknown, we can substitute them with their estimates (plug-in procedure).

The parameters can be estimated with the maximum likelihood (ML) procedure, i.e. by minimizing the function

$$-2l = n_h \log 2\pi + \log |\Sigma_\Delta| + (\mathbf{z}_h - \hat{\mathbf{z}}_l - \delta_0 \mathbf{1}_{n_h})^T \Sigma_\Delta^{-1} (\mathbf{z}_h - \hat{\mathbf{z}}_l - \delta_0 \mathbf{1}_{n_h}) \tag{A.1}$$

where Σ_Δ is known as the mean squares prediction error (MSPE) or kriging variance.

$\mathbf{R}_\delta = \{r_\delta(\mathbf{s}_i, \mathbf{s}_j)\}$ is the correlation matrix of the GP for the shift effect $\delta(\mathbf{s}_i)$, $\mathbf{I}_{n_h} \in \mathbb{R}^{n_h \times n_h}$ is an identity matrix and $\mathbf{1}_{n_h} \in \mathbb{R}^{n_h}$ is a vector of ones.

If the partial derivative with respect to δ_0 are computed and set to 0, the ML estimator of δ_0 is given by:

$$\hat{\delta}_0 = \frac{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} (\mathbf{z}_h - \hat{\mathbf{z}}_l)}{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}}. \tag{A.2}$$

The other unknown parameters can be found by minimizing twice the logarithm of the restricted likelihood [31] given by:

$$-2l_{\hat{\delta}_0} = n_h \log 2\pi - \log(\mathbf{F}_\Delta^T \mathbf{F}_\Delta) + \log |\Sigma_\Delta| + \log |\mathbf{F}_\Delta^T \Sigma_\Delta^{-1} \mathbf{F}_\Delta| + (\mathbf{z}_h - \hat{\mathbf{z}}_l - \hat{\delta}_0 \mathbf{1}_{n_h})^T \Sigma_\Delta^{-1} (\mathbf{z}_h - \hat{\mathbf{z}}_l - \hat{\delta}_0 \mathbf{1}_{n_h}). \tag{A.3}$$

Substituting $\hat{\delta}_0$ from Equation (A.2) the prediction in a new location \mathbf{s}_0 can be rewritten as

$$\begin{aligned} \hat{z}_h(\mathbf{s}_0) &= \mathbb{E}(z_h(\mathbf{s}_0) | \mathbf{z}_h, \mathbf{z}_l) = \hat{z}_l(\mathbf{s}_0) + \hat{\delta}_0 + (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} (\mathbf{z}_l - \hat{\mathbf{z}}_l - \hat{\delta}_0 \mathbf{1}_{n_h}) \\ &= \hat{z}_l(\mathbf{s}_0) + [1 - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}] \hat{\delta}_0 - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \hat{\mathbf{z}}_l + \\ &\quad + (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \mathbf{z}_l \\ &= \hat{z}_l(\mathbf{s}_0) - \left\{ [1 - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}] \frac{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1}}{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}} - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \right\} \\ &\quad \cdot \Sigma_\Delta^{-1} \hat{\mathbf{z}}_l + \left\{ [1 - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}] \frac{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1}}{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}} + \right. \\ &\quad \left. + (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \right\} \mathbf{z}_l \\ &= \hat{z}_l(\mathbf{s}_0) - \mathbf{a}^T \hat{\mathbf{z}}_l + \mathbf{a}^T \mathbf{z}_l \end{aligned}$$

with

$$\mathbf{a}^T = [1 - (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}] \frac{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1}}{\mathbf{1}_{n_h}^T \Sigma_\Delta^{-1} \mathbf{1}_{n_h}} + (\sigma_\delta^2 \mathbf{r}_\delta + \sigma_0)^T \Sigma_\Delta^{-1}. \tag{A.4}$$

\mathbf{a}^* is the vector which minimizes the Mean squared prediction error (MSPE) of $\hat{Z}_h(\mathbf{s}_0)$.

First of all, the prediction is unbiased

$$\begin{aligned} \mathbb{E}[\hat{z}_h(\mathbf{s}_0)] &= \mathbb{E}[\hat{z}_l(\mathbf{s}_0) - \mathbf{a}^T \hat{\mathbf{z}}_l + \mathbf{a}^T \mathbf{z}_l] = \\ \mathbb{E}[\hat{z}_l(\mathbf{s}_0) - \mathbf{a}^T \hat{\mathbf{z}}_l + \mathbf{a}^T \hat{\mathbf{z}}_l + \mathbf{a}^T \mathbf{1}_{n_h} \hat{\delta}_0 + \delta + \varepsilon_h] &= \\ = \mathbf{f}_\Delta^T(\mathbf{s}_0) \boldsymbol{\rho} + \mathbf{a}^T \mathbf{1}_{n_h} \delta_0 = \mathbb{E}[Z_h(\mathbf{s}_0)] = \mathbf{f}_\Delta^T(\mathbf{s}_0) \boldsymbol{\rho} + \delta_0 \\ \Rightarrow \mathbf{a}^T \mathbf{1}_{n_h} \delta_0 = \delta_0 \Rightarrow \mathbf{a}^T \mathbf{1}_{n_h} &= 1. \end{aligned} \tag{A.5}$$

Starting from the definition it is possible to compute the MSPE, or the so called kriging variance

Acknowledgement

The authors gratefully acknowledge the UK's Engineering and Physical Sciences Research Council (EPSRC) funding of the EPSRC Fellowship in Manufacturing: Controlling geometrical variability of products in the digital and smart manufacturing era (Ref: EP/R024162/1). JW acknowledges the National Science Foundation of China (NSFC) funding (No. 51705178).

$$\begin{aligned}
& \mathbb{E} \{ \hat{z}_h(\mathbf{s}_0) - z_h(\mathbf{s}_0) \}^2 \\
&= \mathbb{E} \left\{ \left[\hat{z}_l(\mathbf{s}_0) + \mathbf{a}_*^T \boldsymbol{\delta} + \underbrace{\mathbf{a}_*^T \mathbf{1}_{n_h} \delta_0}_{\substack{(\mathbf{A}, \mathbf{s}) \\ =0}} - \delta_0 + \right. \right. \\
&\quad \left. \left. - \mathbf{a}_*^T \hat{z}_l + \mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta - \mathbf{f}_\Delta^T(\mathbf{s}_0) \rho - \delta(\mathbf{s}_0) - \varepsilon_\Delta \right]^2 \right\} \\
&= \mathbb{E} \{ [\hat{z}_l(\mathbf{s}_0)(\hat{z}_h(\mathbf{s}_0) - z_l(\mathbf{s}_0)) - \mathbf{a}_*^T (\hat{z}_h - z_h) + \mathbf{a}_*^T \boldsymbol{\delta} + \mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta + \\
&\quad - \delta(\mathbf{s}_0) - \varepsilon_\Delta]^2 \} \\
&= \mathbb{E} \{ (\hat{z}_h(\mathbf{s}_0) - z_l(\mathbf{s}_0))^2 + \mathbf{a}_*^T (\hat{z}_l - z_h) (\hat{z}_l - z_h)^T \mathbf{a}_* + \\
&\quad + \mathbf{a}_*^T \boldsymbol{\delta} \boldsymbol{\delta}^T \mathbf{a}_* + \mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta \boldsymbol{\varepsilon}_\Delta^T \mathbf{a}_* + \delta^2(\mathbf{s}_0) + \varepsilon_\Delta^2 + \\
&\quad - 2\hat{z}_l(\mathbf{s}_0)(\hat{z}_h(\mathbf{s}_0) - z_l(\mathbf{s}_0))(\hat{z}_l - z_h)^T \mathbf{a}_* + 2\hat{z}_l(\mathbf{s}_0)(\hat{z}_h(\mathbf{s}_0) - z_l(\mathbf{s}_0)) \mathbf{a}_* \boldsymbol{\delta} + \\
&\quad + 2\hat{z}_l(\mathbf{s}_0)(\hat{z}_h(\mathbf{s}_0) - z_l(\mathbf{s}_0)) \mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta - 2\hat{z}_l(\mathbf{s}_0)(\hat{z}_h(\mathbf{s}_0) - z_l(\mathbf{s}_0)) \delta(\mathbf{s}_0) + \\
&\quad - 2\hat{z}_l(\mathbf{s}_0)(\hat{z}_h(\mathbf{s}_0) - z_l(\mathbf{s}_0)) \boldsymbol{\varepsilon}_\Delta - 2\mathbf{a}_*^T (\hat{z}_l - z_h) \mathbf{a}_*^T \boldsymbol{\delta} + \\
&\quad + 2\mathbf{a}_*^T (\hat{z}_l - z_h) \delta(\mathbf{s}_0) - 2\mathbf{a}_*^T (\hat{z}_l - z_h) \mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta + 2\mathbf{a}_*^T (\hat{z}_l - z_h) \boldsymbol{\varepsilon}_\Delta + \\
&\quad + 2\mathbf{a}_*^T \boldsymbol{\delta} \mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta - 2\mathbf{a}_*^T \boldsymbol{\delta} \delta(\mathbf{s}_0) - 2\mathbf{a}_*^T \boldsymbol{\delta} \boldsymbol{\varepsilon}_\Delta - 2\mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta \delta(\mathbf{s}_0) - 2\mathbf{a}_*^T \boldsymbol{\varepsilon}_\Delta \boldsymbol{\varepsilon}_\Delta - 2\delta(\mathbf{s}_0) \boldsymbol{\varepsilon}_\Delta \} \\
&= \hat{\sigma}_l(\mathbf{s}_0) + \mathbf{a}_*^T \boldsymbol{\Sigma}_0 \mathbf{a}_* + \sigma_\delta^2 \mathbf{a}_*^T \mathbf{r}_\delta \mathbf{a}_* + \sigma_\Delta^2 \mathbf{a}_*^T \mathbf{a}_* + \sigma_\delta^2 + \sigma_\Delta^2 - 2\mathbf{a}_*^T \boldsymbol{\sigma}_0 \\
&\quad - 2\sigma_\delta^2 \mathbf{a}_*^T \mathbf{r}_\delta \\
&= \hat{\sigma}_l(\mathbf{s}_0) + \sigma_\delta^2 + \sigma_\Delta^2 + \mathbf{a}_*^T (\boldsymbol{\Sigma}_0 + \sigma_\delta^2 \mathbf{r}_\delta + \sigma_\Delta^2 \mathbf{I}_{n_h}) \mathbf{a}_* - 2\mathbf{a}_*^T (\sigma_\delta^2 \mathbf{r}_\delta + \boldsymbol{\sigma}_0) \\
&= \hat{\sigma}_l(\mathbf{s}_0) + \sigma_\delta^2 + \sigma_\Delta^2 + \mathbf{a}_*^T \boldsymbol{\Sigma}_\Delta \mathbf{a}_* - 2\mathbf{a}_*^T (\sigma_\delta^2 \mathbf{r}_\delta + \boldsymbol{\sigma}_0) \\
&= \hat{\sigma}_l(\mathbf{s}_0) + \sigma_\delta^2 + \sigma_\Delta^2 - (\sigma_\delta^2 \mathbf{r}_\delta + \boldsymbol{\sigma}_0)^T \boldsymbol{\Sigma}_\Delta^{-1} (\sigma_\delta^2 \mathbf{r}_\delta + \boldsymbol{\sigma}_0) + \frac{[\mathbf{1} - (\sigma_\delta^2 \mathbf{r}_\delta + \boldsymbol{\sigma}_0)^T \boldsymbol{\Sigma}_\Delta^{-1} \mathbf{1}_{n_h}]^2}{\mathbf{1}_{n_h}^T \boldsymbol{\Sigma}_\Delta^{-1} \mathbf{1}_{n_h}}.
\end{aligned}$$

References

- [1] Girao P, Pereira J, Postolache O. Multisensor data fusion and its application to decision making. 2006. p. 47–59.
- [2] Weckenmann A, Jiang X, Sommer K, Neuschaefer-Rube U, Seewig J, Shaw L, Ester T. Multisensor data fusion in dimensional metrology. *Manuf Technol* 2009;58:701–21.
- [3] Carbone V, Carocci M, Savio E, Sansoni G, De Chiffre L. Combination of a vision system and a coordinate measuring machine for the reverse engineering of freeform surfaces. *Int J Adv Manuf Technol* 2001;17:263–71.
- [4] Shen T, Huang J, C M. Multiple-sensor integration for rapid and high-precision coordinate metrology. In: Conference II, editor. *Advanced intelligent mechatronics*. 1999. p. 908–15.
- [5] Ramasami S, Raja J, Boudreau B. Data fusion strategy for multiscale surface measurement. *J Micro Nano-Manufacturing* 2013;1:1–6.
- [6] Xia H, Ding Y, Mallick B. Bayesian hierarchical model for combining misaligned two-resolution metrology data. *IIE Trans* 2011;43:242–58.
- [7] Colosimo BM, Pacella M, Senin N. Multisensor data fusion via Gaussian process models for dimensional and geometric verification. *Precis Eng* 2015;40:199–213.
- [8] Wang J, Pagani L, Leach R, Zeng W, Colosimo B, Zhou L. Study of weighted fusion methods for the measurement of surface geometry. *Precis Eng* 2017;47:111–21.
- [9] Yin Y, Ren MJ, Sun L, Kong L. Gaussian process based multi-scale modelling for precision measurement of complex surfaces. *CIRP Annals* 2016;65:487–90.
- [10] Yin Y, Ren MJ, Sun L. Dependant Gaussian processes regression for intelligent sampling of freeform and structured surfaces. *CIRP Annals* 2017;66:511–4.
- [11] Cressie N. *Statistics for spatial data*. New York: John Wiley & Sons; 1993.
- [12] Santner T, Williams B, Notz W. *The design and analysis of computer experiments*. New York: Springer; 2003.
- [13] Kennedy M, O'Hagan A. Predicting the output from a complex computer code when fast approximations are available. *Biometrika* 2000;87:1–13.
- [14] Qian Z, Seepersad C, Joseph V, Allen J, Wu C. Building surrogate models based on detailed and approximate simulations. *ASME J Mech Des* 2006;128:668–77.
- [15] Santner T, Williams B, Notz W. *The design and analysis of computer experiments*. Springer Verlag; 2003.
- [16] Shabenberger O, Gotway C. *Statistical methods for spatial data analysis*. Chapman & Hall/CRC; 2005.
- [17] Guennebaud G, Jacob B, et al. *Eigen v3*. <http://eigen.tuxfamily.org>; 2010.
- [18] Hjelle O, Daehlen M. Multilevel least squares approximation of scattered data over binary triangulations. *Comput Visual Sci* 2005;8:83–91.
- [19] OpenMP Architecture Review Board. *OpenMP application program interface URL*: <http://www.openmp.org>; 2018.
- [20] Halton JH. *Algorithm 247: radical-inverse quasi-random point sequence*. *Commun ACM* 1964;7:701–2.
- [21] Hammersley JM, Handscomb DC. *Monte Carlo methods*. New York: John Wiley & Sons; 1964. Methuen & Co., London.
- [22] Wang J, Jiang X, Blunt L, Leach R, Scott P. Intelligent sampling for the measurement of structured surfaces. *Meas Sci Technol* 2012;23.
- [23] Petró S, Cavallaro M, Moroni G. Performance evaluation of non contact measuring systems considering bias. *Proceedings of the 4th international conference on advanced Research on rapid prototyping*. 2009. p. 167–73.
- [24] ISO 3452. *Non-destructive testing - penetrant testing*. ISO; 2013.
- [25] CloudCompare. *Cloudcompare (version 2.9) [gpl software] URL*: <http://www.cloudcompare.org/>; 2018.
- [26] Elkott D, Elmaraghy H, Elmaraghy W. Automatic sampling for CMM inspection planning of free-form surfaces. *Int J Prod Res* 2002;40:2653–76.
- [27] Pagani L, Scott PJ. Curvature based sampling of curves and surfaces. *Comput Aided Geom Des* 2018;59:32–48.
- [28] Wang J, Pagani L, Zhou L, Liu X, Lu W, Leach R, et al. Uncertainty-guided intelligent sampling strategy for high-efficiency surface measurement via free-knot b-spline regression modelling. *Precis Eng* 2019;56:38–52.
- [29] Giannelli C, Jttler B, Speleers H. Thb-splines: the truncated basis for hierarchical splines. *Comput Aided Geom Des* 2012;29:485–98. *Geometric Modeling and Processing* 2012.
- [30] Dokken T, Lyche T, Pettersen KF. Polynomial splines over locally refined box-partitions. *Comput Aided Geom Des* 2013;30:331–56.
- [31] Harville D. Maximum likelihood approaches to variance component estimation and to related problems. *J Am Stat Assoc* 1977;72:320–38.