Input and output manifold constrained Gaussian process regression for galvanometric setup calibration

Reference:
De Boi Ivan, Sels Seppe, De Moor Olivier, Vanlanduit Steve, Penne Rudi.- Input and output manifold constrained Gaussian process regression for galvanometric setup calibration
IEEE transactions on instrumentation and measurement / Institute of Electrical and Electronics Engineers [New York, N.Y.]- ISSN 1557-9662 - 71(2022), 2509408
Full text (Publisher's DOI): https://doi.org/10.1109/TIM.2022.3170968
To cite this reference: https://hdl.handle.net/10067/1883050151162165141

Input and Output Manifold Constrained Gaussian Process Regression for Galvanometric Setup Calibration

Ivan De Boi, Seppe Sels, Olivier De Moor, Steve Vanlanduit, and Rudi Penne

Abstract—**Data-driven techniques are finding their way into the calibration procedure of galvanometric setups. However, they bypass the underlying physical or mathematical model completely. Recent work has shown that a simple assumption about an underlying truth can improve the predictions: laser beams leaving the device follow a straight line. In this paper we take that approach a step further. Both the inputs (the pairs of rotations of the two mirrors) and outputs (the straight lines) lie on a manifold. We can incorporate this prior knowledge in the model via constraints built in the formulation of a covariance function. We propose two constrained models: one in which a linear constraint on the direction vector is written as a differential equation and one in which a quadratic constraint is imposed by a reparametrization of the line coordinates. We compare them to data-driven and unconstrained model-based approaches. We show that enforcing constraints improves the quality of the predictions significantly and thus the accuracy of the calibration. We validate our findings against real world data by predicting points on validation planes, calculating line segment distances, considering the training times for the models and assessing how much a predicted line resembles an actual straight line.**

Index Terms—**galvanometer calibration, Gaussian processes.**

I. Introduction

Many applications of galvanometric setups [1] demand accurate calibration. This means being able to control both the direction and the location of the laser beam guided by the rotating mirrors. This plays a vital role in Laser Doppler Vibrometry [2], 3D laser scanners [3], medical imaging [4], LIDAR systems [5] and in computer vision for self driving cars [6].

The common pinhole model, that is used in camera calibration, cannot be used for calibrating a galvanometric setup, because there is no single centre of projection. Several models have been proposed to correct the effects of the geometric distortion [7]. In [8], a transformation of a coordinate system is described. The main drawback of these approaches is their complexity, resulting in difficult non-convex optimization tasks. Furthermore, the models fail to account for all distortions such as the ones caused by non-planar mirrors.

Recently, more data-driven approaches have been proposed in various fields, in which the calibration challenge is transformed into a regression problem. This approach is also proved fruitful in spectroscopic calibration [9], [10], infrared spectroscopy [11], camera calibration [12] and LIDAR calibration [13].

Data-driven calibration for galvanometric laser scanners was proposed in [14], [15]. The authors implemented artificial neural nets (ANN), Support Vector Regression (SVR) and Gaussian processes (GP). The latter are a way to perform statistical (Bayesian) inference and learn a non-parametric regression model directly from the data. GPs are explained more in depth in section B. These methods proved to outperform the mathematical model of [7] and look-up table-based calibration procedures.

In [16], this idea was taken a step further by combining these two approaches: a semi-data-driven or hybrid approach. This model was formulated with only one major assumption about the underlying physical truth: laser beams guided by the two mirrors follow a straight line. This assumption is lacking in the purely data-driven approaches.

Our previous work [16] as well as [14], [15] naively assumed independent outputs for the GPs. Over the last years, multi-output GPs, in which the outputs are correlated, have been thoroughly investigated. An extensive review can be found in [17]. In our case, the outputs are not correlated. However, they are not fully independent either. Information about one of the outputs does not imply knowledge about any of the other outputs. But together, the outputs do follow a certain rule or constraint. Knowledge about all the outputs except one, does encode the underlying value for the missing one. For example, assume that a 3D direction vector is normalized to a length of one. Knowledge about one of the components does not reveal anything about the other two. However, knowing the values for the x- and y-component allows for the calculation of the z-component via $x^2 + y^2 + z^2 = 1$.

We can exploit these constraints via covariance functions or kernels of our GPs. Over the past years, several methods for doing so have been formulated in [18], [19]. An elaborate overview, including methods beyond those two, is given in [20].

In this work, we investigate both the input and output space of the dataset, which consists of the mapping between input angles of the two rotation mirrors and the laser beams. We represent the latter by six coordinates that together form the so called Plücker coordinates of a straight line. A short introduction on this is given in section C. First, we observe that the input space is not Euclidean, but topologically a manifold that is a torus. This leads us to propose a periodic kernel

Ivan De Boi, Seppe Sels, Olivier De Moor, Steve Vanlanduit and Rudi Penne are with the faculty of Applied Engineering, department Electromechanics, University of Antwerp, Groenenborgerlaan 171, B 2020 Antwerp, Belgium and InViLab research group (http://invilab.be). E-mail: ivan.deboi@uantwerpen.be

in which we separate the two input dimensions (i.e. the two mirror rotations). Second, we argue that straight lines can be described by points in a five-dimensional projective space $\mathbb{P}^5$ that live on a four-dimensional manifold M_2^4. This is a quadric, which means its points satisfy a quadratic equation. To enforce this quadratic constraint on the predictions of our model, we propose two different approaches:

1) Reformulate the constraint as a linear differential equation
2) Reparametrize the quadratic constraint and solve a matrix factorization

Both approaches are achieved by rewriting the covariance function of the underlying GPs.

We propose two models, in which we incorporate prior knowledge about the data in the kernels. By doing so, we enforce constraints on the predictions that improve their quality. We compare these models to the mathematical model of [7], the data-driven model proposed in [14], [15] and the semi data-driven approach by [16]. All models are validated on real world data gathered with a scanning laser Doppler vibrometer (type: Polytec PSV-QTec).

The rest of this paper is structured as follows. In the next section we explain how we gathered the data and give some theoretical background on Gaussian processes and Plücker coordinates used to describe straight lines. The third section describes the models we investigate. In a fourth section we present the results. In Section V we discuss these findings. Finally, conclusions are provided.

II. Materials and Methods

A. Gathering experimental data

To compose a dataset consisting of straight lines, we used a Polytec PSV-QTec scanning laser vibrometer with built-in range detection. Its laser beam is controlled by two rotating mirrors. As described in the manufacturer's device manual [21], we first performed a 2D and 3D alignment to establish a reference coordinate system. We aimed the beam at a detection plane, on which we assigned a point that serves as the origin of the coordinate system. The Z-axis points from the origin towards the device horizontally. The Y-axis points upwards and the X-axis from left to right from the perspective of the device. Then we moved the plane in thirteen different positions and orientations. The 3D-coordinate of the intersection point of the laser with each detection plane was calculated by the built-in 3D scanning capabilities of the vibrometer. From the thirteen planes, we kept four aside as validation planes. Points on these planes were not used to train the models. A schematic overview of the setup is provided in Fig. 1.

The data includes 11 values for the first mirror rotation angles and 21 for the second. This results in 231 laser beams. Collinear points, generated by a particular pair of mirror rotations and thus a single laser beam, were filtered via RANSAC [22]. Straight lines are fitted on the inlier points using the least squares method described in [23]. Points on any of the validation planes were not included. A plot of the detected points can be seen in Fig. 2.

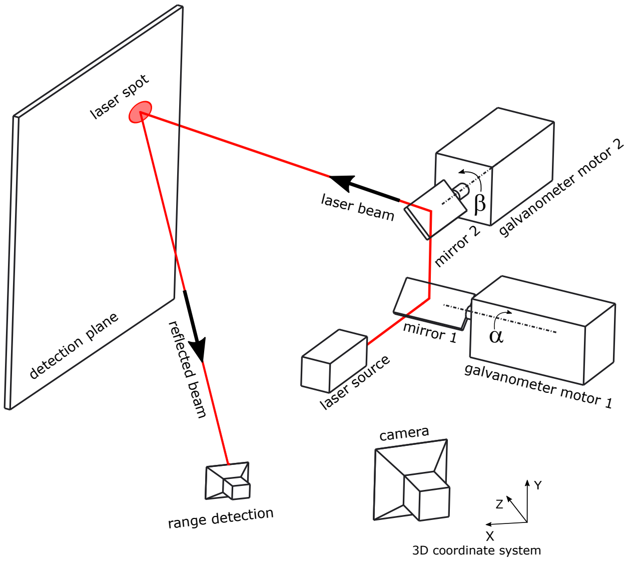

Fig. 1. Overview of the galvanometric setup. The laser beam is controlled by two rotating mirrors. The 3D-coordinate of the intersection point on the detection plane is measured by a range detector and camera.

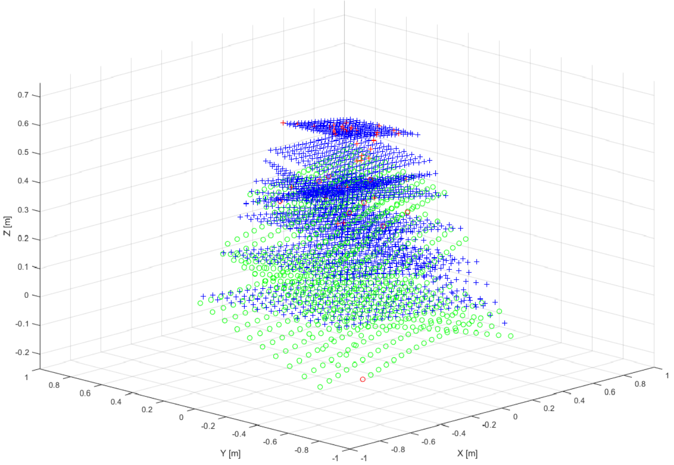

Fig. 2. Points where the laser beam hit the detection plane, which was moved to thirteen different positions. Blue crosses are points that are used in the best fit approach to construct a dataset of lines. Green circles are points on the four validation planes. Red circles are outlier points that are removed from the planes.

Furthermore, we took the same RANSAC and best fit approach to find the homogenous coordinates of the planes themselves. We calculate the intersection point of the found plane and every best fit line. These intersection points are considered the corrected versions of the measured points and are used in the validation of models that have the assumption of straight lines. For the data-driven approach, which does not make such an assumption, this correction was not used.

Since the main contribution of this paper is to improve the calibration procedure of a galvanometric setup, we only work with this real world generated data. No synthetically data was used throughout. In the following sections we explain how these measured intersection points of the laser with each detection plane can be exploited in a semi-data driven calibration method.

B. Gaussian processes

As thoroughly explained in [24], a Gaussian process (GP) is defined as a continuous collection of random variables, any finite subset of which is normally distributed as a multivariate distribution. We focus on Gaussian processes, as they are a flexible regression tool that provide a built-in mechanism against overfitting. Furthermore, they follow a non-parametric Bayesian paradigm, in which the data speaks for itself.

Formally, let $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ be a dataset of n observations, where $\mathbf{x}$ is an input vector of dimension d and y is a scalar-valued observation. We want to find a mapping $f : \mathbb{R}^d \to \mathbb{R}$,

$$y = f(\mathbf{x}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2), \tag{1}$$

with ϵ being identically distributed observation noise. This map can be found by sampling a Gaussian process, which is fully defined by its mean $m(\mathbf{x})$ and *covariance function*, or *kernel*, $k(\mathbf{x}, \mathbf{x}')$. It is generally denoted as

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \tag{2}$$

The de-facto default kernel for a GP is the squared exponential kernel (SE), also called the radial basis function kernel, the Gaussian kernel or exponentiated quadratic kernel. It has the form

$$k_{SE}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2l^2}\right), \tag{3}$$

in which σ_f^2 is a height-scale factor and l the length-scale that determines the radius of influence of the training points. In [15] the usage of different kernels in this context were investigated.

In our previous work [16], we included a different length-scale l for both input arguments. In this case, these are the two mirror rotations. This technique is called automatic relevance determination (ARD) [25]. In general, for $\mathbf{x} \in \mathbb{R}^d$ the kernel has the form

$$k_{SEARD}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_{j=1}^{d}\left(\frac{\|\mathbf{x} - \mathbf{x}'\|}{l_j}\right)^2\right). \tag{4}$$

A covariance function is parametrized by a vector of hyperparameters $\boldsymbol{\theta}$. For the SE these are the length-scale and the height-scale. Their values are learned through a gradient based optimization algorithm that maximizes the log marginal likelihood. In our experiments we use the `quasi-Newton` method from the Matlab 2020 implementation of Gaussian processes [26]. In this process, a complexity term is involved that penalizes over-complex models, which reduces overfitting [24]. For a comprehensive discussion on various covariance functions, we refer to [25].

C. Plücker coordinates

In projective three-dimensional space $\mathbb{P}^3$, a straight line $\mathbf{L}$ can be represented by six homogenous coordinates $(l_1 : l_2 : l_3 : l_4 : l_5 : l_6)$. In the Euclidean setting, these can be interpreted as a normalized direction $\mathbf{l} = (l_1, l_2, l_3)$ and a moment $\bar{\mathbf{l}} = (l_4, l_5, l_6)$ with respect to the origin. This moment can be found by a point $\mathbf{p}$ the line passes through via

$$\bar{\mathbf{l}} = \mathbf{p} \times \mathbf{l}. \tag{5}$$

These six coordinates are called the *Plücker coordinates* or the *homogeneous line coordinates* of the line $\mathbf{L}$ [27]. They are independent of the choice of $\mathbf{p}$. Since we are not concerned about the orientation, $(\mathbf{l}, \bar{\mathbf{l}})$ and $(-\mathbf{l}, -\bar{\mathbf{l}})$ describe the same line.

Only six-tuples that obey the following two conditions are straight lines:

$$\|\mathbf{l}\| = 1, \tag{6}$$

$$\mathbf{l} \cdot \bar{\mathbf{l}} = 0. \tag{7}$$

The latter is called the *Grassmann-Plücker relation*. Plücker coordinates can also be interpreted as points in $\mathbb{P}^5$, which lie on the *Klein quadric* M_2^4 described by

$$l_1 l_4 + l_2 l_5 + l_3 l_6 = 0. \tag{8}$$

In general, a *screw centre* $C \in \mathbb{R}^6$ can be written as $(\mathbf{c}, \bar{\mathbf{c}})$. The pitch of C is defined as

$$\rho = \frac{\mathbf{c} \cdot \bar{\mathbf{c}}}{\|\mathbf{c}\|^2}. \tag{9}$$

This quantity serves as a measure of the deviation of C being a straight line. This only holds for $\mathbf{c}$ not being the zero vector, in which case C would be a line at infinity. A more in depth explanation of line geometry can be found in [28].

III. Models

The objective of the calibration of the galvanometric setup, is to find a mapping from the input parameters to the set of straight lines. In our case the input parameters are the rotation angles of the two mirrors. We will label them α for the first mirror and β for the second mirror. As stated in the previous section, straight lines can be described by a six-tuple of numbers. Thus the mapping has the form $f : (\alpha, \beta) \to (l_1, l_2, l_3, l_4, l_5, l_6)$. Since the aim is to find straight lines in Euclidean 3D-space, we want these six-tuples to satisfy Eq. 6 and Eq. 7. In other words, the output of the prediction should be a point on the manifold M_2^4. These are the constraints we impose on the outputs of the GPs. We propose two models that incorporate these constraints and compare them to a mathematical and data-driven models.

A. Mathematical model

In [7] a mathematical model was proposed to calibrate the galvanometric setup. This model consists of parameters that describe the distance between the mirrors, their position, the tilt angle between them, the position of the origin of the laser beam, the initial direction of the laser beam and the conversion parameters between the inputs and the actual angles of the mirrors. These values are found by minimizing the difference between a set of lines generated by these parameter values and the set of measured lines. We followed the procedure described in the paper and will refer to this model as Mathematical. The initial guess for the parameters were based on measurements on the device itself.

B. Data-driven

We implemented a purely data-driven approach as explained in [15], by defining three GPs for every detection plane. Each GP was trained on either the x-, y- or z-component of the 3D-coordinates of the intersection points of the laser beam and the plane. The covariance matrices were constructed with the SE ARD kernel described in Eq. 4.

C. Six Gaussian processes

To learn the Plücker coordinates directly, we implement 6 distinct Gaussian processes, i.e. one for each output component in $(l_1, l_2, l_3, l_4, l_5, l_6)$. The kernel used, is the SE with ARD as described in Eq. 4. To calculate the similarity between two pairs of input parameter values $\mathbf{x} = (\alpha, \beta)$ and $\mathbf{x}' = (\alpha', \beta')$, the Euclidean distance is used.

D. Six Gaussian processes with periodic kernel

The usage of Euclidean distances in the kernel is suboptimal, as $\mathbf{x}$ and $\mathbf{x}'$ are not points in $\mathbb{E}^2$. They are pairs of rotation angles. Both inputs components live in a space that is periodically curved on itself. From a topological point of view, the input space can be interpreted as a Cartesian product of two circles $\mathbb{S}^1 \times \mathbb{S}^1$, which is known as a manifold called a torus $\mathbb{T}^2$. As a first improvement on the kernel from the previous model, we implement a periodic kernel (PER), proposed by [29], which extended with ARD and a period of 2π for both α and β, has the from

$$k_{PER}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{2}{l_\alpha^2} \sin^2\left(\frac{|\alpha - \alpha'|}{2}\right)\right) \cdot \exp\left(-\frac{2}{l_\beta^2} \sin^2\left(\frac{|\beta - \beta'|}{2}\right)\right), \tag{10}$$

in which σ_f^2 is the height-scale and l_α^2 and l_β^2 are the length-scales for each input dimension. In this kernel, the inputs are periodic and the dimensions are separated from each other.

E. Direction and point on a plane

As described in [16], the Grassmann-Plücker relation can be enforced by construction. Instead of training a GP on each of the six components of the Plücker coordinates, we train on the directions of the straight lines and on the intersection points of those lines with a known plane parallel to the XY-plane. From these intersection points and the directions, the moments of the lines can be calculated via Eq. 5. This ensures zero-pitch lines according to Eq. 9, since the calculated moment is always perpendicular to the direction. For this approach three GPs are needed for the three components of the direction vectors and two GPs for the 3D-coordinates of the intersection with the plane (the z-component is fixed). We refer to this model as Dir IPP.

F. Linear constraint on the direction

The three GPs in the model proposed above, whose predictions together form the direction vectors for the straight lines, must follow Eq. 6, i.e. their norms must equal one. To put this another way, all these direction vectors can be seen as points on a unit sphere $\mathbb{S}^2$. This allows for the implementation of a constraint: the predicted output (a three-vector) should always lie on a unit sphere. To achieve this constraint, we perform two steps:

First, we incorporate the number of the three output dimensions as an extra input. This strategy is explained in [17]. Another elaborate explanation on multi-output Gaussian processes can be found in [30]. The input of the GP can be written as $\mathbf{x} = (\alpha, \beta, d)$, in which d is the number of the output dimension. In our case, this is either one, two or three. This results in three times as many data points. The output of the GP is still a scalar, but we can combine three GP-predictions, each for a different number of output dimension, in such a way that we get a three-vector. The kernel in this model is extended to take in an extra input. For independent outputs, the covariances for inputs that do not have the same number for their output dimension, are zero. When inputs are of the same output dimension, the kernel k_{PER} in Eq. 10 is used. This leads to a 3×3 block covariance matrix. The altered kernel now has the form

$$k_{PER3D}(\mathbf{x}, \mathbf{x}') = \begin{cases} k_{PER}([\alpha, \beta], [\alpha', \beta']), & \text{for } d = d' \\ 0, & \text{for } d \neq d'. \end{cases} \tag{11}$$

Second, we adjust the kernel following a method described in [18]. The authors describe a way to have linear constraints be built in the kernel function itself. An example of those would be a linear differential equation. We refer to their supplementary material for a full description. Here, we explain how this can be implemented in our context. Even though our constraint itself is quadratic in nature (outputs of the GPs describe points on the surface of a unit sphere $x^2 + y^2 + z^2 = 1$), we can rewrite this as a linear differential equation. After a reparametrization from two input angles to three Cartesian coordinates

$$f_1 : x = \cos(\alpha)\sin(\beta), \tag{12}$$

$$f_2 : y = \sin(\alpha)\sin(\beta), \tag{13}$$

$$f_3 : z = \cos(\beta), \tag{14}$$

we can formulate the following linear constraint

$$f_1 - \frac{\partial f_2}{\partial \alpha} + \frac{\partial f_3}{\partial \alpha} = 0. \tag{15}$$

This can be written more compactly as

$$\mathscr{F}_{\mathbf{x}}[\mathbf{f}(\mathbf{x})] = \mathbf{0}, \tag{16}$$

in which $\mathbf{x} = [x, y, z]^T$, $\mathbf{f} = [f_1, f_2, f_3]^T$ and $\mathscr{F}_{\mathbf{x}}$ is a linear operator mapping the function $\mathbf{f}(\mathbf{x})$ to another function $\mathbf{g}(\mathbf{x})$.

We can also relate $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ via another linear operator $\mathscr{G}_{\mathbf{x}}$ such that

$$\mathscr{G}_{\mathbf{x}}[\mathbf{g}(\mathbf{x})] = \mathbf{f}(\mathbf{x}). \tag{17}$$

The constraint in Eq. 15 can now be written as

$$\mathscr{F}_{\mathbf{x}}[\mathscr{G}_{\mathbf{x}}[\mathbf{g}(\mathbf{x})] = \mathbf{0}. \tag{18}$$

Reformulating the linear operators as matrix-vector multiplications yields

$$\mathscr{F}_{\mathbf{x}}\mathscr{G}_{\mathbf{x}} = \mathbf{0}. \tag{19}$$

This reformulation allows us to impose the constraint om the operator $\mathscr{G}_{\mathbf{x}}$ instead on the GP for $\mathbf{f}(\mathbf{x})$. With $\mathbf{g}(\mathbf{x})$ sampled from a GP with mean $\mu_g(\mathbf{x})$ and covariance $K_{\mathbf{g}}(\mathbf{x}, \mathbf{x}')$, this means that

$$\mathbf{f}(\mathbf{x}) = \mathscr{G}_{\mathbf{x}}\mathbf{g} \sim \mathcal{GP}(\mathscr{G}_{\mathbf{x}}\mu_{\mathbf{g}}, \mathscr{G}_{\mathbf{x}}K_{\mathbf{g}}\mathscr{G}_{\mathbf{x}'}^T). \tag{20}$$

A complete explanation on how to find $\mathscr{G}_{\mathbf{x}}$ for any given $\mathscr{F}_{\mathbf{x}}$ using the nullspace of the latter, can be found in [18]. In our case, we find

$$\mathscr{F}_{\mathbf{x}} = \left[1, -\frac{\partial}{\partial_\alpha}, \frac{\partial}{\partial_\alpha}\right], \tag{21}$$

$$\mathscr{G}_{\mathbf{x}} = \left[\frac{\partial}{\partial_\alpha}, 1, 0\right]^T, \tag{22}$$

which leads to

$$\mathscr{G}_{\mathbf{x}}\mathscr{G}_{\mathbf{x}'}^T = \begin{bmatrix} \frac{\partial^2}{\partial_\alpha \partial_{\alpha'}} & \frac{\partial}{\partial_\alpha} & 0 \\ \frac{\partial}{\partial_{\alpha'}} & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{23}$$

Operator $\mathscr{G}_{\mathbf{x}}\mathscr{G}_{\mathbf{x}'}^T$ can only be applied on functions that are at least twice differentiable. To that end, we need to combine the two cases in the covariance function in Eq. 11. Furthermore, the absolute value signs can be dropped, since the sine function is an odd function which is then squared: $\sin^2(-\theta) = \sin^2(\theta)$. We propose the formulation

$$\begin{aligned} k_{COM}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp&\left(-\frac{2}{l_\alpha^2}\sin^2\left(\frac{\alpha - \alpha'}{2}\right)\right) \\ \cdot \exp&\left(-\frac{2}{l_\beta^2}\sin^2\left(\frac{\beta - \beta'}{2}\right)\right) \\ \cdot \exp&\left(-\frac{(d - d')^2}{\epsilon^2}\right), \end{aligned} \tag{24}$$

in which ϵ is an arbitrary small value. This results in the last factor to be either one for equal dimension number d or (very close to) zero for different dimensions. Applying Eq. 23 on Eq. 24 yields a covariance function

$$k_{DIF} = \mathscr{G}_{\mathbf{x}}k_{COM}\mathscr{G}_{\mathbf{x}'}^T = \begin{bmatrix} A & B & 0 \\ C & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} k_{COM}, \tag{25}$$

in which, with $\gamma = \frac{\alpha - \alpha'}{2}$ (for brevity in notation),

$$A = -\frac{\sin^2(\gamma)\cos^2(\gamma)}{l_\alpha^4} + \frac{\cos^2(\gamma)}{l_\alpha^2} - \frac{\sin^2(\gamma)}{l_\alpha^2}, \tag{26}$$

$$B = -\tfrac{2}{l_\alpha^2}\sin(\gamma)\cos(\gamma), \tag{27}$$

$$C = \tfrac{2}{l_\alpha^2}\sin(\gamma)\cos(\gamma). \tag{28}$$

This describes a covariance function that imposes the quadratic constraint, written as a linear differential equation, which states that the norm of the 3D vector predicted by the GP should always be equal to one. We apply this for the prediction of the direction of the line. We refer to this model as DirCon IPP.

G. One Gaussian proces with quadratic constraints

In this section we derive an alternative kernel to the one proposed above. The aim is to impose the constraints in Eq. 6 and Eq. 7 simultaneously. We follow the method described in [19] by applying the following steps.

First, let $\mathbf{z} = (l_1, l_2, l_3, l_4, l_5, l_6)$ and $\mathbf{Q} = \mathbf{z}^T\mathbf{z}$, which encodes all the quadratic terms. This is a 6×6 symmetric matrix and thus it is fully determined by its upper triangular part. We now formulate a training point $\mathbf{y} \in \mathbb{R}^{21}$ as the concatenation of those upper triangular elements

$$\mathbf{y} = [\mathbf{Q}_{11}, ..., \mathbf{Q}_{ij}, ..., \mathbf{Q}_{66}]^T, \text{with } i \leq j. \tag{29}$$

Second, by incorporating the number of the output dimension as an input, we apply the same method as explained above to construct a multi-output GP. Again, the input of the GP can be written as $\mathbf{x} = (\alpha, \beta, d)$, in which d is the number of the output dimension. Here d ranges from 1 to 21 and thus the dataset is twenty-one times larger. The covariance function for this model is given by Eq. 11.

As stated and proved in [19], a set of training examples $\{\mathbf{y}_1, ..., \mathbf{y}_n\}$ that satisfy the linear constraints $\mathbf{A}\mathbf{y}_i = \mathbf{b}$ will result in a GP for which the mean prediction $\mu(\mathbf{x}_*)$ also satisfies $\mathbf{A}\mu(\mathbf{x}_*) = \mathbf{b}$. The authors demonstrate their findings on a similar challenge, namely a rotation (quaternion) of unit norm. In our case, $\mathbf{A}$ is a 2×21 matrix and $\mathbf{b} = [1, 0]^T$.

However, we are interested in predictions in the form of the six dimensional $\mathbf{z}_*$ and not the twenty-one dimensional $\mathbf{y}_*$. Generally, this is achieved by casting this problem to a matrix factorization problem and minimizing the Frobenius norm between the factorization and the output of the GP. The solution can be obtained in closed form as follows. For our case, we first compose the matrix

$$\mathbf{Q}_* = \begin{bmatrix} y_{*1} & y_{*2} & \cdots & y_{*6} \\ y_{*2} & y_{*7} & \cdots & y_{*11} \\ \vdots & \vdots & \vdots & \vdots \\ y_{*6} & y_{*11} & \cdots & y_{*21} \end{bmatrix} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T, \tag{30}$$

in which $\mathbf{V}$ is a one column matrix and $\mathbf{\Sigma}$ is just a scalar. We then calculate

$$\mathbf{z}_* = \sqrt{\mathbf{\Sigma}}\mathbf{V}^T. \tag{31}$$

The solution to the factorization is not unique. There is a sign ambiguity that arises when taking the square root. As proposed in [19], this can be solved by taking into account information from the context. In our, case we know that the z-component of the direction vector of the predicted lines should be negative. If this is not the case, the line points in the opposite direction and the Plücker coordinate can be multiplied by -1 to reverse the signs of its six components.

IV. Results

We compare all models via 5-fold cross-validation (CV). The data described above is split into five blocks. The models are trained on four of them. Predictions are made on the inputs from the remaining block and validated against the blocks used for training. This process is repeated five times, at every iteration with a different block for validation. This ensures that every block is included in the validation set exactly once. We trained all models on four different sizes for the dataset: 50, 100, 150 and 200 lines. We reran the training ten times for each size. To assess the quality of the models, four evaluation methods are considered.

First, for every input pair, we predict points on the validation planes. The data-driven model predicts those via three GPs (one for each 3D-coordinate component), while the other models calculate an intersection point between a predicted line and each validation plane. To assess the models, the root mean squared error (RMSE) of the Euclidean distance between the predicted points and the measured points (after correction as described in the Gathering data section) is calculated. An overview can be found in Table I. This validation method assesses the practical validity of our models. In a real world scenario, one would be interested in where the laser beam exactly hits an object under inspection. Working with a correct set of predefined inputs (the mirror rotations) is of utmost importance when accuracy is being pursued.

Second, for the line-based models, the distance between the predicted line and the measured line is calculated as described in [28]. We do not consider the line itself, but only a finite oriented line segment with boundaries in a region of interest, i.e. values that are appropriate to the measured points of the dataset. Let $\mathbf{p_1}$ and $\mathbf{p_2}$ be the intersection points of lines $\mathbf{L_1}$ and $\mathbf{L_2}$ with the xy-plane. These lines also intersect another plane, parallel to the xy-plane, at points $\mathbf{q_1}$ and $\mathbf{q_2}$. The distance d between the line segments is defined as

$$d^2 = (\mathbf{p_1} - \mathbf{p_2})^2 + (\mathbf{q_1} - \mathbf{q_2})^2 + (\mathbf{p_1} - \mathbf{p_2}) \cdot (\mathbf{q_1} - \mathbf{q_2}). \tag{32}$$

This distance serves as the error in the RMSE of the line segment distances in Table II.

Third, we record the total time it takes to train a model for all cross-validation iterations and all reruns. The results can be found in Table III. Finally, in a fourth evaluation method, we consider the pitch of the predicted line for the line-based models. These findings are summarized in Table IV.

TABLE I
RMSE PREDICTED POINTS ON VALIDATION PLANES

Methods	n = 50	n = 100	n = 150	n = 200
Mathematical	21221.57	19410.13	19037.90	17599.49
Data-driven	12538.62	12585.36	10733.94	6078.32
6 GP	35.09	22.38	117.95	27.25
6 GP PER	8.79	1.22	0.46	0.26
Dir IPP	6.71	1.68	1.47	0.61
DirCon IPP	5.36	1.68	1.37	0.62
QCon	15.84	1.17	0.42	0.27

The RMSE in [µm] of the distances calculated between points on the validation planes and the predictions made by the models.

TABLE II
RMSE LINE SEGMENT DISTANCES

Methods	n = 50	n = 100	n = 150	n = 200
Mathematical	35207.23	32456.77	31800.85	29527.41
Data-driven	NA	NA	NA	NA
6 GP	51.46	86.04	284.63	51.00
6 GP PER	13.00	2.21	0.82	0.44
Dir IPP	11.33	2.85	0.87	1.01
DirCon IPP	8.93	2.82	0.83	1.00
QCon	22.34	1.96	0.70	0.43

The RMSE in [µm] of the line segment distances between the measured lines and the lines predicted by the models. The data-driven model does not produce lines.

TABLE III
TRAINING TIMES

Methods	n = 50	n = 100	n = 150	n = 200
Mathematical	566.52	1193.43	1713.65	2351.08
Data-driven	66.50	91.43	174.89	265.00
6 GP	108.59	191.22	303.26	529.85
6 GP PER	14.35	41.73	93.16	160.16
Dir IPP	10.89	29.86	68.59	122.64
DirCon IPP	29.69	104.68	214.90	370.89
QCon	642.56	2652.05	3612.35	3744.28

The time in [s] it took to train each model. These are the cumulations of the times for each cross validation iteration and each rerun.

V. Discussion

As thoroughly discussed in [15], the mathematical model is based on a hard to solve non-convex high-dimensional optimization problem. The procedure requires a qualitative initial guess. Moreover, several time-consuming reruns are needed to find an optimal solution. The amount of training time highly depends on the stopping criteria for the optimization process. For every laser beam, this model calculates its intersection point with the second mirror and a direction. As such, the pitch of the generated line is always zero, even though this does not result in more accurate predictions.

The data-driven approach is free from the optimization problem of the mathematical model, as the model is bypassed completely. However, removing all assumptions about the underlying truth was shown to be too restrictive in [16]. The semi data-driven 6 GP model, in which six fully independent GPs are implemented, performs better than the mathematical or data-driven approach. However, training six GPs instead of three takes twice as long.

The most notable improvement is the introduction of a kernel in the 6 GP PER model that takes into account that the inputs are not points in Euclidean space, but live on a torus. Even though the kernel is slightly more complex, the actual training time for this model is lower. The training of a GP implies finding a set of hyperparameters that maximizes the log marginal likelihood [24]. This process converges sooner for a kernel that describes the underlying reality or dataset better. Furthermore, we exploit the fact that the period for both dimensions is exactly 2π. These hyperparameters do not have to be learned during the training of the GPs.

TABLE IV
RMSE PITCHES

Methods	n = 50	n = 100	n = 150	n = 200
Mathematical	0	0	0	0
Data-driven	NA	NA	NA	NA
6 GP	5.23	34.49	58.02	5.17
6 GP PER	0.89	0.26	0.13	0.05
Dir IPP	0	0	0	0
DirCon IPP	0	0	0	0
QCon	0.53	0.05	0.02	0.01

The RMSE in [µm] of the pitches of the lines predicted by the models. The data-driven model does not produce lines. The mathematical model, the Dir IPP and DirCon IPP model produce lines with a pitch that is exactly zero, apart from machine accuracy.

In contrast to the data-driven, the 6 GP and the 6 GP PER model, the Dir IPP model produces lines that do fulfil the Grassmann-Plücker. This is achieved by construction. For the smallest dataset, this approach resulted in better predictions. For larger datasets, the gain in accuracy in comparison to the other models vanishes. Furthermore, this model requires only five GPs, which take less time to train than the six GPs of the 6 GP PER model.

The model with the linear differential equation constraint, also always produces lines with a pitch of zero. Moreover, the constraint on the norm of the direction of the line, as described in Eq. 6, yields predictions that are slightly better than those of the models without the constraint. This effect diminishes as the datasets grow. Since these datasets have lines with a unit norm for their direction vectors, the models all produce lines with a direction vector with a norm close to one. This constraint only has an effect on a minor defect in the prediction. Thus making the overall gain from this constraint minimal. Only when the dataset is relatively small, and the predicted directions tend to deviate from points on the unit sphere does this model outperform the other ones.

The model with the quadratic reparametrization is trained on both constraints simultaneously. Therefore, it finds an optimum that satisfies both. This results in sacrificing one constraint in favour of the other one. When trained on the smallest dataset, it is outperformed by the other models. When trained on bigger datasets, it outperforms the other models significantly.

In order to construct multi-output GPs, the number of the output dimension is transformed into an input variable. The new set of outputs are the components of the original dataset stacked in a single column of scalers. For the model DirCon IPP, which is based on the differential equation, this yields a dataset that has three times as many entries. The dataset for the model QCon, with the quadratic reparametrization, is composed of twenty-one times more entries. This is an important point of concern. With n the number of data points in a dataset, the standard complexity of the Gaussian process is $\mathcal{O}(n^3)$ for computation and $\mathcal{O}(n^2)$ for storage [31]. Herein lies the biggest trade-off of the models proposed. There is a gain in quality of the predictions, but at a significant time penalty.

In the last two decades, several approaches have been proposed to overcome the hurdle of large datasets for Gaussian processes. An overview can be found in [31]. This would be a fruitful area for further work, especially for the QCon model. Furthermore, in this work, we made no assumption about the relationship between the straight lines. The implemented covariance functions assume smoothness in the Plücker coordinates when varying the rotation angles of the mirrors. However, this is still a naive approach that can be taken further by incorporating the relationship between the straight lines into the model. Although, when doing so, the model becomes more complex and less data-driven, which was one of the strong shoots of the approach taken. More research is needed to assess where the optimum lies of the hybrid mix between fully data-driven and purely mathematical models.

VI. Conclusion

We presented two constrained models that, given a pair of rotation angles for the mirrors of a galvanometric setup, predict a straight line. Improving the kernel to take into account the manifolds on which the inputs and outputs live, leads to better predictions. This is of great importance for everyone who works with setups that require an accurate calibration to guide a laser beam.

However, this comes with a considerable extra computational time for the training of the GP(s), as the kernels become more complex and the datasets are re-parametrized into larger volumes. Even though this increase in training time is only an issue during the calibration process itself. It is not an issue at inference time when predictions are being made.

By means of cross-validation, we investigated our models on real world data in order to improve the calibration of a Polytec PSV-QTec scanning laser Doppler vibrometer. After calibration, we investigated the accuracy of the predicted laser beams and points those beams hit in a real world scenario.

Acknowledgment

This research has been funded by the University of Antwerp (BOF FFB200259, Antigoon ID 42339).

References

[1] X. Sun, B. Zhou, W. Xie, and Y. Zhang, “The design of laser scanning galvanometer system,” in *The International Conference on Photonics and Optical Engineering (icPOE 2014)*, A. Tian, A. Asundi, W. Liu, and C. Zhang, Eds., vol. 9449, International Society for Optics and Photonics. SPIE, 2015, pp. 676 – 684. [Online]. Available: https://doi.org/10.1117/12.2082774

[2] P. Castellini, G. M. Revel, and E. P. Tomasini, “Laser doppler vibrometry,” in *An Introduction to Optoelectronic Sensors*. World Scientific, 2009, pp. 216–229.

[3] H. El-Din Fawzy, “3d laser scanning and close-range photogrammetry for buildings documentation: A hybrid technique towards a better accuracy,” *Alexandria Engineering Journal*, vol. 58, no. 4, pp. 1191–1204, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1110016819301061

[4] H.-K. Park, J.-W. Chung, and H.-S. Kho, “Use of hand-held laser scanning in the assessment of craniometry,” *Forensic Science International*, vol. 160, no. 2, pp. 200–206, 2006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0379073805005773

[5] T. Fersch, R. Weigel, and A. Koelpin, “Challenges in miniaturized automotive long-range lidar system design,” in *Three-Dimensional Imaging, Visualization, and Display 2017*, B. Javidi, J.-Y. Son, and O. Matoba, Eds., vol. 10219, International Society for Optics and Photonics. SPIE, 2017, pp. 160 – 171. [Online]. Available: https://doi.org/10.1117/12.2260894

[6] Y. Zhang, L. Wang, X. Jiang, Y. Zeng, and Y. Dai, "An efficient lidar-based localization method for self-driving cars in dynamic environments," *Robotica*, p. 1–18, 2021.

[7] A. Manakov, H.-P. Seidel, and I. Ihrke, "A Mathematical Model and Calibration Procedure for Galvanometric Laser Scanning Systems," in *Vision, Modeling, and Visualization (2011)*, P. Eisert, J. Hornegger, and K. Polthier, Eds. The Eurographics Association, 2011.

[8] X. Meng, G. Cao, X. Li, and Q. Lv, "2-d scanning galvanometer error analysis and its correction," *Journal of Physics: Conference Series*, vol. 1345, p. 022068, nov 2019. [Online]. Available: https://doi.org/10.1088/1742-6596/1345/2/022068

[9] J. R. Long, V. G. Gregoriou, and P. J. Gemperline, "Spectroscopic calibration and quantitation using artificial neural networks," *Analytical Chemistry*, vol. 62, no. 17, pp. 1791–1797, 1990. [Online]. Available: https://doi.org/10.1021/ac00216a013

[10] P. J. Gemperline, J. R. Long, and V. G. Gregoriou, "Nonlinear multivariate calibration using principal components regression and artificial neural networks," *Analytical Chemistry*, vol. 63, no. 20, pp. 2313–2323, 1991. [Online]. Available: https://doi.org/10.1021/ac00020a022

[11] T. Næs, K. Kvaal, T. Isaksson, and C. Miller, "Artificial neural networks in multivariate calibration," *Journal of Near Infrared Spectroscopy*, vol. 1, no. 1, pp. 1–11, 1993. [Online]. Available: https://doi.org/10.1255/jnirs.1

[12] D.-M. Woo and D.-C. Park, "Implicit camera calibration using an artificial neural network," in *Neural Information Processing*, I. King, J. Wang, L.-W. Chan, and D. Wang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 641–650.

[13] J. Shi, Z. Zhu, J. Zhang, R. Liu, Z. Wang, S. Chen, and H. Liu, "Calibrcnn: Calibrating camera and lidar by recurrent convolutional neural network and geometric constraints," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 10 197–10 202.

[14] S. Lüdtke, B. Wagner, R. Bruder, P. Stüber, F. Ernst, A. Schweikard, and T. Wissel, "Calibration of galvanometric laser scanners using statistical learning methods," in *Bildverarbeitung für die Medizin 2015*, H. Handels, T. M. Deserno, H.-P. Meinzer, and T. Tolxdorff, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 467–472.

[15] T. Wissel, B. Wagner, P. Stüber, A. Schweikard, and F. Ernst, "Data-driven learning for calibrating galvanometric laser scanners," *IEEE Sensors Journal*, vol. 15, no. 10, pp. 5709–5717, 2015.

[16] I. De Boi, S. Sels, and R. Penne, "Semi data-driven calibration of galvanometric setups using Gaussian processes," *IEEE Transactions on Instrumentation and Measurement*, 2021. [Online]. Available: https://doi.org/10.1109/TIM.2021.3128956

[17] M. A. Álvarez, L. Rosasco, and N. D. Lawrence, "Kernels for vector-valued functions: A review," *Foundations and Trends® in Machine Learning*, vol. 4, no. 3, pp. 195–266, 2012. [Online]. Available: http://dx.doi.org/10.1561/2200000036

[18] C. Jidling, N. Wahlström, A. Wills, and T. B. Schön, "Linearly constrained Gaussian processes," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/71ad16ad2c4d81f348082ff6c4b20768-Paper.pdf

[19] M. Salzmann and R. Urtasun, "Implicitly constrained Gaussian process regression for monocular non-rigid pose estimation," *Advances in Neural Information Processing Systems*, vol. 23, pp. 2065–2073, 2010.

[20] L. P. Swiler, M. Gulian, A. L. Frankel, C. Safta, and J. D. Jakeman, "A survey of constrained Gaussian process regression: Approaches and implementation challenges," *Journal of Machine Learning for Modeling and Computing*, vol. 1, no. 2, pp. 119–156, 2020.

[21] Polytec, "Polytec scanning vibrometer theory manual," 2018. [Online]. Available: https://www.pearl-hifi.com/06_Lit_Archive/05_LDV/Polytec_Software_Manuals/PSV_400_Theory.pdf

[22] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, p. 381–395, Jun. 1981. [Online]. Available: https://doi.org/10.1145/358669.358692

[23] V. Lesueur and V. Nozick, "Least square for grassmann-cayley agelbra in homogeneous coordinates," in *Image and Video Technology – PSIVT 2013 Workshops*, F. Huang and A. Sugimoto, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 133–144.

[24] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[25] D. Duvenaud, "Automatic model construction with Gaussian processes," Ph.D. dissertation, Computational and Biological Learning Laboratory, University of Cambridge, 2014.

[26] *MATLAB version 9.9.0.1570001 (R2020b)*, The Mathworks, Inc., Natick, Massachusetts, 2021.

[27] H. Pottmann, M. Peternell, and B. Ravani, "An introduction to line geometry with applications," *Comput. Aided Des.*, vol. 31, pp. 3–16, 1999.

[28] H. Pottmann and J. Wallner, *Computational line geometry*. Springer Science & Business Media, 2009.

[29] D. J. MacKay *et al.*, "Introduction to Gaussian processes," *NATO ASI series F computer and systems sciences*, vol. 168, pp. 133–166, 1998.

[30] H. Liu, J. Cai, and Y.-S. Ong, "Remarks on multi-output Gaussian process regression," *Knowledge-Based Systems*, vol. 144, pp. 102–121, 2018.

[31] A. G. Wilson, C. Dann, and H. Nickisch, "Thoughts on massively scalable Gaussian processes," *arXiv preprint arXiv:1511.01870*, 2015.

Ivan De Boi obtained his Master's degree in Electronics and ICT Engineering Technology from the University of Antwerp in 2001. After working as a project engineer for VRT and Atlas Copco for 7 years as, he decided to throw his career around and became a lecturer at the Bachelor in Multimedia Technology program at the Karel de Grote University College. Apart from his teaching assignment he also worked on various research projects in the field of AI and AR/VR, mainly in healthcare and Industry 4.0 context. In 2020, he joined the research group InViLab at the University of Antwerp (faculty of Applied Engineering, department Electromechanics) to work on his PhD regarding the calibration of 3D-sensors using line variety models.

Seppe Sels holds a Master's degree in industrial sciences (2012, KU Leuven) and obtained his PhD at the University of Antwerp (Op3Mech, 2019). During his PhD thesis, entitled "Linking real-world measurements and sensor data with 3D CAD models" he developed expertise in object detection and pose estimation techniques using computer vision algorithms. In his work, a software framework is designed to quickly calibrate different type of camera systems. With this framework, systems where developed that can map sensor data and camera images to digital twins of measured objects. Currently, he is working at the University of Antwerp (InViLab) improving pose estimation and mapping techniques to to monitor the heat management of a human body.

Olivier De Moor obtained his Master's degree in Electromechanical Engineering from the University of Antwerp in 2020. During his studies he acquired expertise in the field of laser Doppler vibrometry. After his graduation he started as a researcher in the InViLab research group at the University of Antwerp. His current research is focused on robotics, vision techniques, machine learning and optical measurement techniques. Currently he is working on the improvement of vision-based quality control for the textiles sector.

Steve Vanlanduit obtained his PhD entitled “High spatial resolution modal analysis” at the Department of Electrical Engineering of the Vrije Universiteit Brussel (VUB) in 2001. In 2003 he was appointed as a professor in the Department of Mechanical Engineering of VUB, and in 2014 he became head of the Department of Electromechanical Engineering of the University of Antwerp. The research of prof. Vanlanduit is focused on laser and camera-based optical measurement techniques. He has authored over 200 journal publications on the use of optical measurement techniques in different domains (flow measurement, vibration engineering, robotics, quality control, materials inspection, etc.). He is editor of the scientific journals Measurement (Elsevier), Measurement: Sensors (Elsevier), Sensors (MDPI), and Metrology (MDPI). Steve is actively involved in the organization of several international conferences on optical measurement techniques.

Rudi Penne received his D.Sc. degree (1992) in mathematics from the University of Antwerp (Belgium). There he was appointed a professor (2013) in the Faculty of Applied Engineering. His research interests cover a broad area, including topology, combinatorial and computational geometry, structural topology and rigidity, and projective geometry with applications in robot kinematics and computer vision. Besides fundamental research, he promotes and disseminates science by means of popularizing articles, books, blogs and lectures.