Vision, Modeling, and Visualization (2011)
Peter Eisert, Konrad Polthier, and Joachim Hornegger (Eds.)

# A Mathematical Model and Calibration Procedure for Galvanometric Laser Scanning Systems

Alkhazur Manakov[1] Hans-Peter Seidel[2] Ivo Ihrke[1,2]

[1]Universität des Saarlandes [2]MPI Informatik
amanakov@mmci.uni-saarland.de

**Abstract**

*Laser galvanometric scanning systems are commonly used in various fields such as three dimensional scanning, medical imaging, material processing, measurement devices and laser display systems. The systems of such kind suffer from distortions. On top of that they do not have a center of projection, which makes it impossible to use common projector calibration procedures. The paper presents a novel mathematical model to predict the image distortions caused by galvanometric mirror scanning systems. In addition, we describe a calibration procedure for recovering its intrinsic and extrinsic parameters.*

## 1. Introduction

Laser galvanometric scanning systems are used in many different applications. The most widely used application of such kind of system is the barcode reader [BKG*96]. Other applications are laser projection systems [CPP06, KKR03, Wa76] , imaging applications [PDJ98, YBP*04, Sta76, Gel03], including medical imaging, material processing [SRGS01, JLRI91, CLTL83] and measurement devices [Ger75, WW03]. A laser galvanometric scanning system consists of two mirrors driven by limited-rotation motors. Two mirrors are needed in a two-dimensional scanning system, as shown in Fig. 1. The laser beam enters the system and hits the mirror on the X scanner (first mirror). After reflection, the ray hits the second, Y mirror. By rotating the X mirror, one can control the reflection angle and thus the horizontal coordinate of the outcoming ray. The coordinate along the vertical axis is controlled by the Y scanner. Thus, coordinates of the outgoing ray beam are determined by the rotation angles of the mirrors. In some applications, e.g. in active light 3D-scanning, the positions of the illumination and the camera device are required for triangulation. The calibration of a camera is a well known procedure [Bou05]. If the illumination device is a standard projector, the same calibration procedure can be used as long as the projector has a center of projection. However, a laser galvanometric scanning system does not have a single center of projection and the common calibration procedure can not be used. Applications such as material processing require compensation for distortions in the whole field of view. Until now this problem was solved by avoiding galvanometric systems where possible e.g. by using two mechanical linear stages for laser ray positioning [Hak90] or by using narrow opening angles [HG07] for which the distortion is negligible.

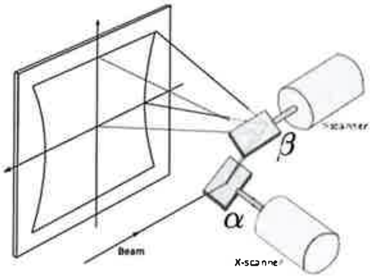


**Figure 1:** *A galvanometric scanning system consists of two mirrors driven by limited-rotation motors. Even in the ideal case, when the incoming ray hits the rotation axes of the mirrors, the system causes distortions.*

Unfortunately, the use of linear stages is often impossible and some applications require a large field of view. In other cases compensating for the distortions is performed using polynomial fitting techniques [HFI*08, CCH09]. The polynomial fitting techniques are applicable only if the distance from the system to the object is fixed and the object is planar. This makes polynomial fitting methods not applicable when the calibration needs to be valid for different depths.

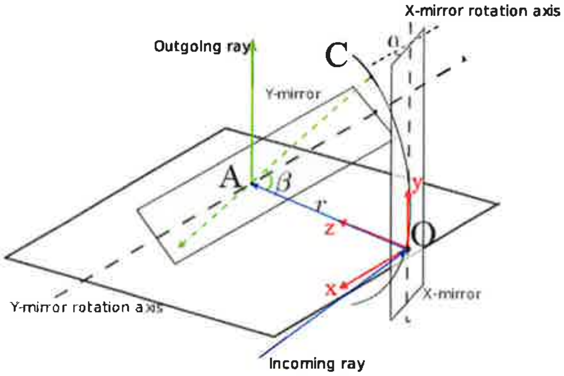


**Figure 2:** *Simple model*

In this paper we present a mathematical model to describe laser galvanometric scanning systems. The model describes the distortions introduced by properties of the system and predicts how the distortions will change depending on the parameters of the system. We only consider geometric distortions. Distortions caused by the nonlinear dependence of the mirror rotation angle from a voltage applied to a motor can be calibrated using known procedures [DPN09]. Based on our model, we describe a calibration procedure for estimating the position of the system (rotation and translation matrices) and its internal parameters determining its distortions.

## 2. Galvanometric scanning system theory

In optics, galvanometric laser control systems are well known and widely used. The distortions they introduce have been described, but mostly for simplified cases [CCH09]. In this paper we present a mathematical model to describe such systems. We start the derivation with a simplified model, exposing the main features of a double mirror scanning system. Though the model can not describe all possible distortions of a galvanometric system it still can be used for calibration, but the result is less accurate. We then extend the model to capture more system distortions and improve the calibration accuracy.

### 2.1. Simple model

We refer to the mirror that an incoming laser ray hits first as *first mirror* or *X-mirror*: This mirror is usually used for scanning along the horizontal axis. Accordingly, the *second mirror* or *Y-mirror* is the mirror in the system which is used to scan along the vertical axis. To derive a simplified model, we make the following assumptions:

1. The rotation axes of the mirrors are perpendicular,
2. The incoming ray hits the first mirror in the rotation axis,
3. The ray reflected from the first mirror hits the second mirror in the rotation axis, and,
4. The ray is thin in comparison to the mirror dimensions.

The assumptions are still reasonable for real systems because the position of nearly any system can be adjusted so that the incoming ray hits the first mirror in the rotation axis. If the system is designed and assembled well enough, then after reflection by the first mirror, the ray should hit the second mirror in the rotation axis. The assumption concerning the relative dimensions of the mirrors and the beam dimensions is reasonable because the mirrors can be produced large enough to fit the ray. Note, that larger mirror dimensions decrease the scanning speed of the system though. This issue is caused by the inertia of the mirrors: larger mirrors have a larger weight and consequently a larger inertia. A thorough discussion can be found in [TWBv95]. The position of the X-mirror is defined by an angle $\alpha$. Equally, the position of Y-mirror is described by an angle $\beta$. The origin of the laser coordinate system $O$ is located on the rotation axis in the middle of the X-mirror (see Fig. 2). The rotation axis of the Y-mirror is parallel to the X-axis of the coordinate system. The Y-axis coincides with the rotation axis of the X-mirror. The distance between the axes of the mirrors is denoted by $r$.

Suppose an incoming ray hits the X-mirror in $O$, intersecting the rotation axis according to the assumptions. Then the coordinates of this point are $O(0;0;0)$. If we rotate the X-mirror, the reflected ray scans a segment along the rotation axis of the Y-mirror. Additionally, $O$ is a projection center for this segment since it does not move while the X-mirror is rotating. Thus, every possible ray in the system passes through $O$ and a point $A$ on the segment parallel to the X-axis. Then the coordinates of $A$ depending on $\alpha$ are $A = (r{\cdot}\tan\alpha; 0; r)$.

Consider the ray coincident with the Z-axis of the coordinate system before intersection with the Y-mirror. A rotation of the Y-mirror changes the direction of the ray so, that if we imagine a moving laser source $O$ instead of the rotating mirror, it appears to move along an arc with radius $r$. Let us denote the point moving along the arc as $C$. The position of $C$ depends on the rotation angle $\beta$. The coordinates of $C$ are $(0; r{\cdot}\sin\beta; r - r{\cdot}\cos\beta;)$. Imagine an outgoing ray from the system. It reflects from the Y-mirror at the point $A$, but it is simpler to imagine the ray passing through the mirror. This assumption does not change the degrees of freedom of the outgoing ray, but simplifies the equations. Pay attention to the fact that in this case the range of change for $\beta$ is twice larger. The outgoing ray then passes through $A$ and $C$, and the ray equation is given by

$$\vec{CA} = \begin{pmatrix} 0 \\ r \cdot \sin\beta \\ r - r \cdot \cos\beta \end{pmatrix} + t \cdot \begin{pmatrix} r \cdot \tan\alpha \\ -r \cdot \sin\beta \\ r \cdot \cos\beta \end{pmatrix}. \quad (1)$$

Imagine we have a screen parallel to the $XOY$ plane at a distance $z_0$ from the origin. The equation of the plane is $z - z_0 = 0$. Then the intersection point $\vec{P}$ of the ray and the plane

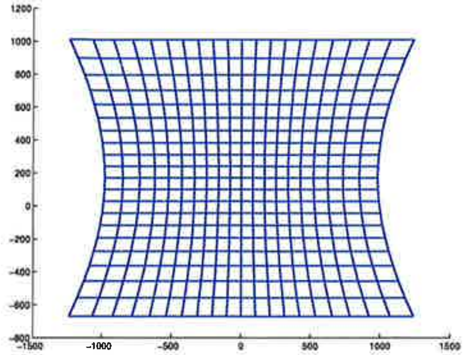


**Figure 3:** *The distortion described by the simple model. Horizontal lines are straight, the grid has vertical and horizontal symmetry.*

is

$$\vec{P}=\begin{pmatrix}0\\ r\cdot\sin\beta\\ r-r\cdot\cos\beta\end{pmatrix}+\left(\frac{z_0-r}{r\cdot\cos\beta}+1\right)\cdot\begin{pmatrix}r\cdot\tan\alpha\\ -r\cdot\sin\beta\\ r\cdot\cos\beta\end{pmatrix}. \tag{2}$$

Rewriting Eq.(2) explicitly we obtain:

$$\begin{aligned} x &= -r\cdot\tan\alpha\cdot\left(\frac{z_0-r}{r\cdot\cos\beta}+1\right),\\ y &= \frac{(z_0-r)}{\cos\beta}+r\cdot(1+\sin\beta),\\ z &= z_0, \end{aligned} \tag{3}$$

with *(x; y; z)* being the coordinates of $\vec{P}$ in the laser system. The model of Eq.(3) describes distortions as shown in Fig. 3. Note that these distortions appear even in the ideal case where the ray hits the mirrors in the rotation axes. This mathematical model does not describe the situation when an incoming ray hits one or both mirrors out of the rotation axis or when the rotation axes of the mirrors are not perpendicular. These situations are taken into account by the more profound model we propose next.

### 2.2. Complex model

In the complex model we remove most of the assumptions that we made for the simple model, except that the ray still has to be thin enough to fit onto the mirrors. For the complex model the X-axis is considered to be tilted by $\gamma$ (see Fig. 4), which is common for modern galvanometric systems. The coordinate system origin is located on the rotation axis of the X-mirror (see Fig. 4). The coordinate system is positioned so that the X-axis is parallel to the rotation axis of the Y-mirror and that the Y-axis is perpendicular to it. As in the simple model, the position of the Y-mirror is defined by the rotation angle $\beta$. The current position of the X-mirror is defined by two angles: $\gamma$, the incline of the rotation axis is fixed, and the rotation angle $\alpha$ which is describing the motion of the mirror. In addition, the incoming ray can have arbitrary incidence. It is defined by its origin $\vec{S}(s_x;s_y;s_z)$ and by a direction $\vec{l}=(l_x;l_y;l_z)$.

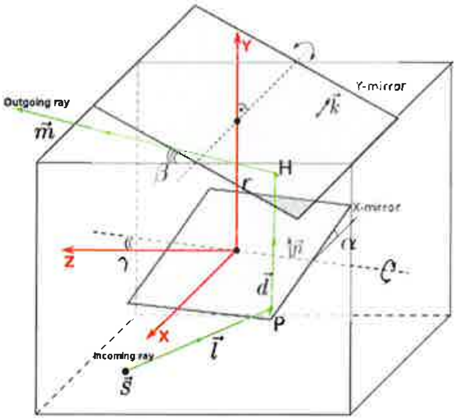


**Figure 4:** *Complex model*

To describe the position of the X-mirror, we need the normal and one point on the mirror. The origin of the coordinate system could be taken as a point on the mirror plane $O=(0;0;0)$. The normal depends on both $\gamma$ and $\alpha$ since the rotation is now performed around an inclined axis. Imagine a dashed coordinate system $X'Y'Z'$ which is derived from $XYZ$ by rotating the latter around the X-axis by $\gamma$. In the dashed coordinate system, vector $\vec{n}'$ = (0; 1; 0) is a normal for the plane when $\alpha$ = 0. In the $XYZ$ coordinate system the same vector is given by $\vec{n}$ = (0; cos$\gamma$; sin$\gamma$). When changing $\alpha$, the normal rotates around the $Z'$-axis. The rotation matrix for this case is

$$R_z=\begin{pmatrix}\cos\alpha & -\sin\alpha & 0\\ \sin\alpha & \cos\alpha & 0\\ 0 & 0 & 1\end{pmatrix}. \tag{4}$$

Thus, the normal in the $XYZ$ system is given by

$$\vec{n}=\begin{pmatrix}-\sin\alpha\\ \cos\alpha\cdot\cos\gamma\\ \cos\alpha\cdot\sin\gamma\end{pmatrix}. \tag{5}$$

The intersection point of the incoming ray with the X-mirror is

$$\vec{P}=\vec{S}+t_x\cdot\vec{l},\quad t_x=-\frac{(\vec{n},\vec{O}-\vec{S})}{(\vec{n},\vec{l})}, \tag{6}$$

where $(\cdot,\cdot)$ denotes the vector scalar product. The ray reflected from the X-mirror has $\vec{P}$ as an origin and the direction $\vec{d}$ of the reflected ray is

$$\vec{d}=2\cdot(\vec{n},-\vec{l})\cdot\vec{n}+\vec{l}. \tag{7}$$

The normal of the second mirror is

$$\vec{k}=\begin{pmatrix}0\\ \cos\beta\\ \sin\beta\end{pmatrix} \tag{8}$$

As point on the plane it is most convenient to use a point on the rotation axis $\vec{Q}=(0;r;0)$. Then the intersection of the

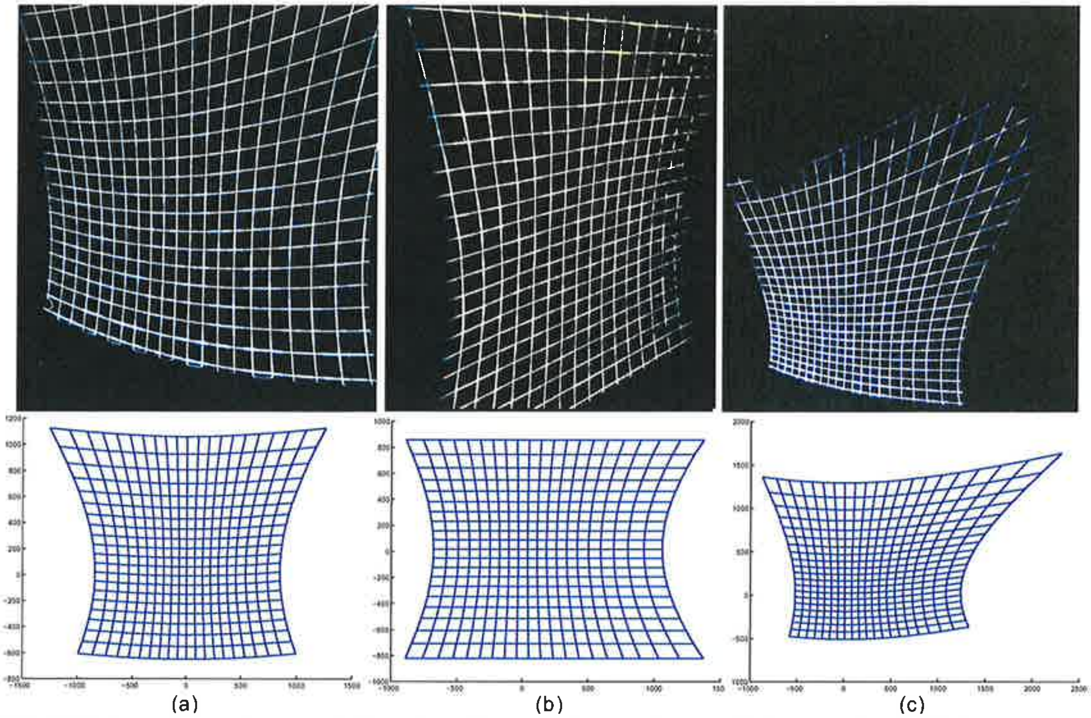


**Figure 5:** *Distortions: top-photographs,bottom-simulations with the complex model. From left to right:*(a) *Displacement, the ray hits the mirror out of the rotation axis.* (b) *The ray comes into the system under 15 degrees angle.* (c) *Combination of the two distortions. Note that the photographs show perspective distortion due to the projection plane being at an angle to the camera's optical axis.*

reflected ray with the Y-mirror is given by

$$\vec{H} = \vec{P} + t_y \cdot \vec{d}, \quad t_y = -\frac{(\vec{k}, \vec{Q} - \vec{P})}{(\vec{k}, \vec{d})} \tag{9}$$

The direction of the outgoing ray becomes

$$\vec{m} = 2 \cdot (\vec{k}, \vec{r}) \cdot \vec{k} - \vec{d}. \tag{10}$$

Finally, we obtain the equation of the outgoing ray

$$\vec{R}(t) = \vec{H} + t \cdot \vec{m}. \tag{11}$$

Intersecting the ray with a planar screen parallel to the $XOY$ plane at the distance $z_0$ from the origin and rewriting the result explicitly we get

$$\begin{aligned} x &= H_x + \frac{z_0 - H_z}{m_z} \cdot m_x, \\ y &= H_y + \frac{z_0 - H_z}{m_z} \cdot m_y, \\ z &= z_0. \end{aligned} \tag{12}$$

This model describes the behavior of a general two-mirror galvanometric system. Note that the complex model reduces to the simple one, Eq.(3), in the following case:

$$\vec{l} = (-1;0;0), \quad \vec{S} = (x;0;0), \quad \gamma = 0$$

where $x$ can be any positive number.

### 2.3. Expressiveness of the Complex Model

To show that the complex model is more expressive than the simple one we conduct a number of experiments. We change the parameters of the galvanometric system, rotating and shifting it with respect to the incoming ray. In this case the system causes distortions that the simple model can not express. We project a grid onto a plane and take photographs. Modeling the situation with the complex model, we compare the modeled images and the photographs.

As can be seen from Fig. 5 the complex model is more expressive then the simple one, which supports only the distortions shown in Fig 3. Qualitatively, the real distortions are well captured by our model. Before we show how our model fits real data, Sec. 4, we discuss how its parameters can be estimated from image measurements.

## 3. Calibration

The parameters describing the system can be divided into two sets: *internal parameters* for properties of the system and *external parameters* to define the position of the system with respect to the camera coordinate system. Consider a galvanometric system more closely for better understanding of the internal parameters. A typical galvanometric system consists of mirrors mounted on galvanometers, motor drivers, a controller and a power supply. The system works

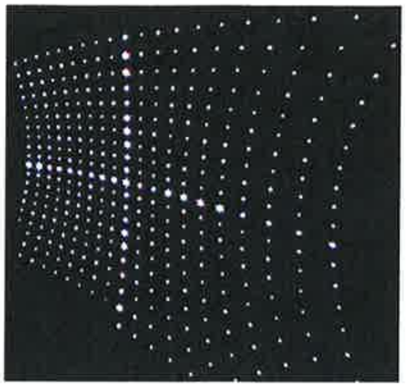


**Figure 6:** *A grid pattern projected onto a planar screen. We project points instead of lines because they are easier and more accurately detectable.*

as follows: the controller issues a command for the motor driver to rotate the mirror to a certain angle. The motor driver applies a voltage to rotate the mirror according to the command. Normally, motor drivers are adjusted for a particular motor so that the relation between applied voltage and rotation angle is linear

$$\alpha = s_x \cdot V_x + \alpha_0$$
$$\beta = s_y \cdot V_y + \beta_0 \qquad (13)$$

where $\alpha$ is the rotation angle of the X mirror, $V_x$ is the applied voltage, $\alpha_0$ is an offset of the rotation angle, and $s_x$ is the slope of the line describing the correspondence between $V_x$ and $\alpha$. The same holds for the other mirror.

In other cases, when the correspondence is not linear it causes additional distortions. In this work we only consider the linear case. However, the linear model can easily be replaced by any other voltage-to-angle mapping model.

One internal parameter of the mirror system is the distance $r$ between the mirrors in the galvanometric system, Fig 4. Some modern galvanometric systems also have a certain angle $\gamma$ (different from zero degrees) between the rotation axes of the mirrors, Fig 4, which also has an influence on the distortions produced by the system. Usually the last two parameters, unlike the slopes and offsets, are given in the manual of the galvo-system. In case they are not mentioned in the manual or have been changed after production, our procedure allows to estimate them. The other two internal parameters which are valid only for the complex model are the origin of the incoming ray $\vec{S}$ and its direction $\vec{l}$. The external parameters are the rotation and translation matrices of the system with respect to the camera coordinate system.

As raw calibration data we use a point based grid pattern (Fig. 6) projected onto a planar screen, in the following simply called plane. We take $N$ photographs with different plane orientations for the calibration. The points of the pattern are equally spaced in terms of applied voltages. Note that the voltages applied to project the grid must also be recorded for voltage-to-angle conversion parameter estimation $(\alpha, s_x)$ and $(\beta, s_y)$. In practice we employ a linear DAC and directly use the DAC values in place of $V_x$ and $V_y$.

Before we start to estimate the parameters of the projection system we need to compute the three-dimensional coordinates of the projected grid points. We extract the image coordinates of the points and triangulate them. For the triangulation we need to know the position of the plane with respect to the camera. For that purpose it is convenient to have a camera calibration pattern printed on the plane. We denote coordinates of the triangulated points $(x_i^j, y_i^j, z_i^j)$, where $i$ is the number of the image and $j$ the number of the grid point in an image. $i = 1...N$, $j = 1...M$, where $N$ is the number of photographs and $M$ is the number of laser grid points on the photo. For a simplified exposition we assume that on every photo all the points of the grid are visible and the grid size is $LxL$ points. The case where not all points are visible is a straight-forward extension. Here and below the coordinates are expressed in camera coordinates if not mentioned otherwise.

### 3.1. Initial estimation of internal parameters

We start the estimation by computing the parameters of the voltage-to-angle conversion, Eq.(13). To establish this correspondence we need to know the rotation angles, which can be computed from the angles between the outgoing rays. The direction of the rays $f^j$, $j = 1...M$ can be estimated by fitting a ray to the three-dimensional points of a common index $j$, measured on different planes. The fitting is performed to the subset of points $W^j = \{w_i^j : i = 1...N\}$, minimizing the error measure

$$E(X^j) = \sum_{i=0}^{N} (d_j^i)^2, j = 1...M. \qquad (14)$$

where $d_j^i$ is the distance from the point $W_i^j$ to the estimated ray. We then measure the angles $\theta_l^k$ between the right-most ray $f_l^1$ and all other rays $f_l^k$ lying on the same horizontal line. Here, $l, k = 1...L$. Similarly, we measure the angles $\psi_k^l$ for the vertical axis the angles between the bottom ray $f_1^k$ and all other rays $f_l^k$, lying on the same vertical line, where $l = 2...L$ and $k = 1...L$ Having the angles we compute the slopes and the offsets Eq.(15).

$$s_x = \frac{\theta_l^1 - \theta_l^L}{V_x^{\theta_l^1} - V_x^{\theta_l^L}}, \quad s_y = \frac{\psi_k^1 - \psi_k^L}{V_y^{\psi_k^1} - V_y^{\psi_k^L}};$$
$$\alpha_0 = \theta_l^1 - s_x \cdot V_x^{\theta_l^1}, \ \ \beta_0 = \psi_k^1 - s_y \cdot V_y^{\psi_k^1}; \qquad (15)$$

$V_x^{\theta_l^k}$ and $V_y^{\psi_k^l}$ are the voltages corresponding to the angles $\theta_l^k$ and $\psi_k^l$.

Estimation of the parameters of the incoming ray is required only for the complex model calibration. The parameters of the incoming ray are estimated by analyzing the shape of the horizontal lines, namely their curvature and stretching since those strongly depend on the direction of the incoming ray. We generate (compute) a set of rays $h^j, j = 1...M$ using the voltage-to-angle mapping parameters estimated in the previous step using the complex model Eq.(12). Changing only the incoming ray parameters, we minimize the distance between corresponding fitted $f^j$ and generated $h^j$.

$$E(X^j) = \sum_{i=0}^{N} (\lambda \cdot (1 - (f^j, h^j)) + d_j)^2, j = 1...M \qquad (16)$$

The error to minimize is given by Eq.(16), where $d_j$ is the distance between corresponding rays, computed as a length of a segment perpendicular to both of them. $\lambda$ is a constant defined as a half of the checkerboard square size. We use the simple model assumptions (Sec. 2.1) as an initial guess for the optimization. To estimate the other internal parameters (for instance, the rotation axis inclination angle and the distance between the mirrors) and to make sure that the estimated internal parameters fit together, we perform a refinement.

We again generate a set of rays $g^j$, $j = 1...M$ using the internal parameters estimated in the previous step and minimize the error between them and the fitted rays $f^j$ by adjusting the internal parameters of the model. This optimization task can be solved by any suitable optimization method. We use the MATLAB optimization toolbox in our experiments.

In principle, it is possible to perform the internal parameter estimation using only two photographs with full field of view of the projected grid. However, we found that for a stable estimation which converges to reasonable parameters, at least 7 photos are required. It is necessary to include at least one photo with the full field of view for every axis (not necessarily both at once).

### 3.2. Initial estimation of external parameters

External parameter estimation can be performed with only two photos. Furthermore, it is not necessary to cover the full field of view. The only condition is that the points that lie on the vertical and horizontal central sheets, Fig. 7, must be visible. The *vertical central sheet* is a three-dimensional surface swept in space by the outgoing ray when the X-mirror is fixed in its median position and the Y-mirror scans the whole angular range. Similarly, the *horizontal central sheet* is a three-dimensional surface swept in space by the outgoing ray when the Y-mirror is fixed in its median position and the X-mirror scans the whole angular range. The $YOZ$ and $XOZ$ planes expressed in laser coordinates are tangent planes, Fig. 7, to the vertical and horizontal central sheets.

The ray which results if both mirrors are in their median positions is the intersection of the tangent planes to the central sheets and is called the *optical axis*. Note that in the simple model case the tangent planes are equal to the central sheets. Thus, the procedure is the same for both models. We consider the solution for the complex model and it can be reduced to the simple one.

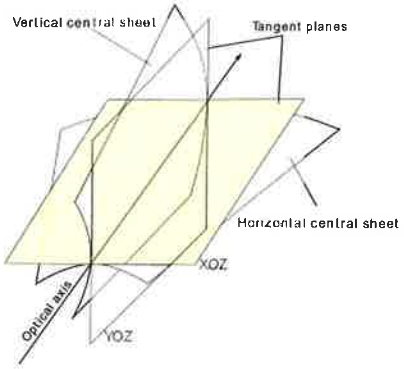


**Figure 7:** *Central sheets and planes*

The goal is to estimate the orientation of the laser coordinate system and the position of its origin. The main idea is to estimate the optical axis first and then fit the $YOZ$ and $XOZ$ tangent planes to the central sheets. The origin can be computed using the model and the 3D-points given in camera coordinates together with the fact that the projection of a point onto the tangent planes are the X- and Y-coordinates expressed in the laser coordinates.

To estimate the optical axis we select the points $D_i^{med}$, $i = 1...N$ corresponding to the intersection of the planar screen and the ray which results if both mirrors are in their median positions. The optical axis is fitted to the points minimizing the distance given by Eq.(14). For tangent plane estimation, we select the points on the median grid lines (lines corresponding to the median voltage values) $B_i^v$ and $B_j^h$, $i = 1...L \cdot N, j = 1...L \cdot N$. Then, we fit the tangent planes to the central sheets in such a way that they pass through the optical axis, remain perpendicular to each other and the squared distance between the plane and the corresponding points ($B_i^v$ and $B_j^h$) is minimal. The error is computed according to Eq.(17), where $d_i$ is the distance from the point to the corresponding tangent plane. As a result we obtain normals $\vec{n_v}$ and $\vec{n_h}$ and a point on every tangent plane.

$$E(X^j) = \sum_{i=0}^{L \cdot N} (d_i)^2. \qquad (17)$$

We define the Z-axis of the galvanometric system as the optical axis. Once we have computed the tangent planes, we know the three vectors (the normals to the planes $\vec{n_v}$ and $\vec{n_h}$ and the optical axis). Therefore, we can fix the rotational degrees of freedom of the system. Thus the rotation matrix $R$ is a matrix composed of those vectors in the following way:

$$R = (\vec{n_v} | \vec{n_h} | (\vec{n_v} \times \vec{n_h})), \qquad (18)$$

where $(\cdot \times \cdot)$ denotes the cross product. To estimate the position of the origin of the system, we need to know the coordinates of the points $(x_i^j; y_i^j; z_i^j)$ given in the laser coordinate system. Knowing the orientation of the galvanometric system $R$, we can compute the projections $(p_i^j; q_i^j)$ of the points $(x_i^j; y_i^j; z_i^j)$ onto both tangent planes. The projections are the coordinates of the points expressed in the laser coordinate system. The missing Z-coordinate can be computed using the model, the estimated internal parameters, and the measured voltages for every pair $(p_i^j;q_i^j)$. Note that the distance is from the $j^{th}$ point to the origin of the laser coordinate system. We use the median Z-value of the points lying on the optical axis as a guess for the origin of the laser coordinate system.

### 3.3. Parameter refinement

As mentioned above, the refinement is performed simultaneously for the external and internal parameters. The refinement is another minimization task. The goal is to know how the system converts applied voltages into an equation of the outgoing ray. We have a set of voltages $(V_x, V_y)$, which we used to project the pattern and the response of the system (photographs with the pattern projected on the plane). Thus we can adjust the parameters of the model of the system so, that the response of the model coincides with the response of the real system as well as possible. The sum of the distances between the points predicted by the model and the experimentally measured ones is used as the error measure. Note that the distance can be computed in camera image space or in three-dimensional space.

## 4. Experimental validation

To validate the developed model and calibration procedure we performed two experiments. The first is calibration, meaning estimation of all system parameters. In the second experiment we use a calibrated system to perform rectified projection onto a curved surface.

### 4.1. Calibration experiment

The validation is conducted in a leave-one-out fashion. We take 10 photographs, but use only 7 to perform the calibration. The accuracy is then measured for the remaining 3 photographs.

For the validation of the calibration procedure we chose a combined distortion case. The incoming laser ray is displaced from the rotation axes of the mirrors and enters the system under a certain angle with the X-axis, Fig. 4. The distortions for this case are shown in Sec. 2.3, see Fig. 5. In this case, the grid is asymmetric, it has neither vertical nor horizontal symmetry. Despite the fact that the applied voltages are equidistant, the distances between the corners are different. Furthermore, we can not see the whole grid because at a certain angle of rotation the ray leaves the mirror. Following the calibration procedure described in Sec. 3, we take photos of the projected grid, Fig. 6. The brighter points correspond to median values of the applied voltages. We also take photographs with the plane under good illumination to determine its position with respect to the camera. The detection of the calibration pattern is implemented in MATLAB. Assuming that the intensity inside of the beam is distributed as a Gaussian, we detect the center of the spot. After calibration of the system we predict the points on the planes that we did not use for calibration. The error measure is

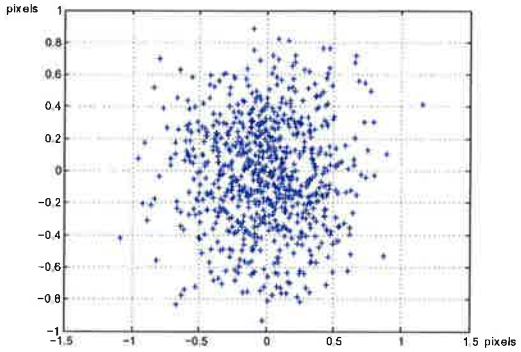


**Figure 8:** *Error distribution*

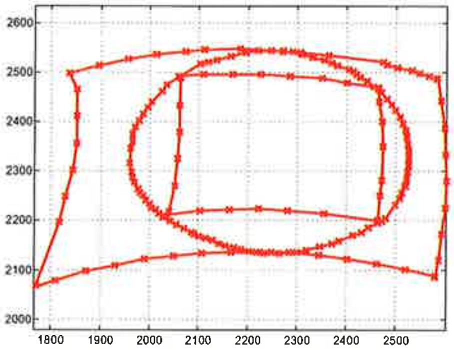


**Figure 9:** *Top: The rectified pattern in laser coordinates. Note: Marked points are provided the remained is interpolated by physical motion the galvo system.*

$$E = \frac{1}{M}\sum_{i=0}^{M}(x_i - p_i)^2, \tag{19}$$

where $M$ is the number of points for which the error is computed, $x_i$ are the measured points, and $p_i$ are the model predicted points. The error is computed in image coordinates and expressed in pixels. The number of points we used for validation is 277. The maximum error per point is 1.2 pixels or 30% of the laser beam width. The average error among all points on the plane is 0.96 pixel, or 27% of the laser beam width. A plot of the error distribution is shown in Fig. 8.

### 4.2. Projection experiment

Using our model and the complete calibration information, including internal parameters, we can rectify any pattern in such a way, that, while projecting onto a curved surface, the pattern appears straight from one particular view-point, (see Fig. 10). For our experiment we used a sphere with a diameter of 23.4 cm. The setup consists of a calibrated camera, a laser-galvanometric system and the sphere (see Fig. 10). The rectified pattern we project, represented in laser coordinates is shown in the Fig. 9.

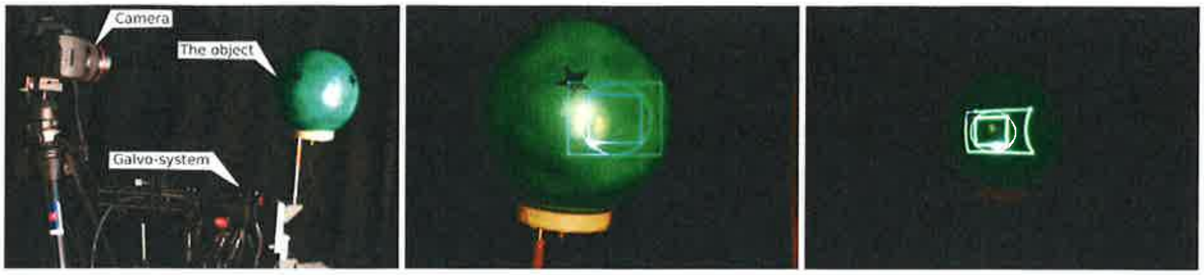


**Figure 10:** *Projection experiment. From left to right: The setup: the camera, the laser and the sphere; view from the camera, the pattern is straight; view from a different camera perspective, the pattern is distorted.*

The rectification is done as follows: Suppose we have a pattern given by a set of points $U = \{u_i : i = 1...N\}$ in the camera image coordinate system and a description of the object geometry. We virtually project the desired pattern from the camera view onto the object geometry. The intersection is given by a set of three dimensional points $\{x'_i : i = 1...N\}$. These are then converted to the laser coordinate system $x_i = R^{-1}(x'_i - T)$, where $R$ and $T$ are rotation and translation matrices to convert from the laser coordinate system to the camera coordinate system. By inverting Eq.(12) and computing the required voltages $V = V^i_x, V^i_y, i = 1...N$, we can project a rectified pattern onto the surface. This experiment shows one possible application of our laser model. Note that for this experiment we also used the combined distortion system. The results are shown in Fig. 10.

## 5. Conclusion

In this work, we present two mathematical models to describe the behavior of galvanometric laser scanners and an associated calibration technique for computing their parameters in real setups. The distinctive feature of the proposed calibration method is that it uses a mathematical model of the galvanometric system. The use of a mathematical model allows the computation of a calibration which is valid for a whole range of depths. Although the simple model is not expressive enough since it is able to describe only one type of distortions that can occur, it can be used for calibration of systems for which the assumptions are valid. The complex model is more expressive and can handle all the distortions introduced by a galvanometric scanning system.

## References

[BKG*96] BRIDGELALL R., KATZ J., GOREN D., DVORKIS P., LI Y.: Laser scanning system and scanning method for reading 1-d and 2-d barcode symbols.

[Bou05] BOUGUET J.-Y.: Camera Calibration Toolbox for Matlab, 2005.

[CCH09] CHEN M.-F., CHEN Y.-P., HSIAO W.-T.: Correction of field distortion of laser marking systems using surface compensation function. *Optics and Lasers in Engineering 47*, 1 (2009), 84 – 89.

[CLTL83] CELLER G. K., LEAMY H. J., TRIMBLE, LEE E.: Method of increasing the grain size of polycrystalline materials by directed energy-beams, September 1983.

[CPP06] CALLISON J. P., PEASE J. S., PEASE R. W.: Laser projection system, November 2006.

[DPN09] DUMA V. F., PODOLEANU A. G., NICOLOV M.: Modeling a galvoscanner with an optimized scanning function. 539–548.

[Gel03] GELBART D.: Method for imaging a continuously moving object, December 2003.

[Ger75] GERHARD W.: Method and apparatus for measuring the distance and/or relative elevation between two points in an opto-electronic manner, August 1975.

[Hak90] HAKKY S. I.: Laser resectoscope with mechanical and laser cutting means, September 1990.

[HFI*08] HULLIN M. B., FUCHS M., IHRKE I., SEIDEL H.-P., LENSCH H. P. A.: Fluorescent immersion range scanning. *ACM Trans. Graph. (Proc. SIGGRAPH 2008) 27*, 3 (2008), 87:1–87:10.

[HG07] HELMCHEN F., GO W.: Imaging cellular network dynamics in three dimensions using fast 3d laser scanning. *Nature Methods 4*, 1 (2007), 73–79.

[JLRI91] JOHNSON J. C., LO H., ROWLEY D., IRLAND T.: Method and apparatus for laser processing a target material to provide a uniformly smooth, continuous trim profile, October 1991.

[KKR03] KURTZ A. F., , KRUSCHWITZ B. E., RAMANUJAN S.: Laser projection display system, June 2003.

[PDJ98] PODOLEANU A. G., DOBRE G. M., JACKSON D. A.: En-face coherence imaging using galvanometer scanner modulation. *Optics Letters 23* (Feb. 1998), 147–149.

[SRGS01] SUKMAN Y. P., RISSER C. J., GORHAM E. W., SCHULTZ D. W.: Laser material processing system with multiple laser sources apparatus and method, November 2001.

[Sta76] STARKWEATHER G. K.: Scanning system for imaging characters combined with graphic curves, August 1976.

[TWBv95] TUMER I., WOOD K. L., BUSCH-VISHNIAC I. J.: Modeling of dynamic effects caused by the beam delivery system in selective laser sintering. In *Proceedings of the 1995 Solid Freeform Fabrication Symposium* (Aug. 1995), pp. 353–361.

[Wa76] WHITBY C. M., AND D. L. W.: Combined laser recording and color projection system, June 1976.

[WW03] WULF O., WAGNER B.: Fast 3d scanning methods for laser measurement systems. 3–8.

[YBP*04] YUN S., BOUDOUX C., PIERCE M., DE BOER J., TEARNEY G., BOUMA B.: Extended-cavity semiconductor wavelength-swept laser for biomedical imaging. *Photonics Technology Letters, IEEE 16*, 1 (jan. 2004), 293 –295.

# Software calibration of scan system distortions

John R. Weisz

TLSI
Somerville, Massachusetts 02143

## ABSTRACT

The need for extremely precise positioning of pre-objective galvonometer based laser pointing systems necessitates calibration to correct for various system distortions. Such distortion errors fall into two categories: those inherent to the geometry of the scanning system layout, including galvonometer mounting configuration and lens specification; and those related to the behavior of the galvonometer as a precise positioning mechanism.

This paper examines system error sources in pre-objective scanning systems and proposes several techniques to correct them.

## 2. INTRODUCTION

A typical use of a galvonometer based laser pointing system is to precisely position a laser pulse on a material to alter its electrical characteristics. Given the rigorous accuracy demands (often better than one part per 10000) of such systems , a calibration is required to correct system positioning distortions.

To provide a basis for understanding scanning system beam positioning, this paper is divided into four sections. The first offers a brief overview of scanning system architecture; the second discusses several system *edge detection* techniques. which are the means by which a scanning system determines its position within the beam field; the third discusses the causes of beam positioning error; and the fourth discusses means for correcting these errors.

## 3. SCANNING SYSTEM ARCHITECTURE

Pre-objective scanning systems comprise two orthogonal mirrors mounted on separate galvonometer motors such that the first galvo mirror deflects a laser beam about one axis and onto the second galvo mirror. which then deflects the laser beam about the other axis, through a focussing lens, and onto the material. The system can therefore position a laser spot to any $(x, y)$ position on the device.

The system computer effects galvo deflection by issuing a digital position command (often 16 to 18 bits in length; this paper will use 18 bits in all equations) to a servomotor. which then generates a voltage in proportion to the commanded position. The galvonometer motor then rotates the mirror in proportion to the command voltage. The complexity of the error correction calculations and of the calibration procedure required to achieve extremely precise position accuracy necessitates that such corrections be implemented in software. Correction calculations execute in real-time during galvo motions to correct the galvo position.

## 4. EDGE DETECTION

A calibration characterizes a system's behavior relative to a reference. In this case, a reference grid composed of accurately spaced lines etched on a stable substrate is used to measure and correct the scan system. The grid is placed in the field of view so that the system can detect the locations of these lines. The efficacy of the detection mechanism directly impacts ultimate system accuracy; obviously. a system's positioning accuracy can be no more precise than the resolution and repeatability by which it detects edges. Two popular methods are described: the *laser eye scan* and the *video eye scan*.

### 5.2 Errors caused by galvonometers

We now isolate the galvonometer to examine how well it positions. A galvonometer rotates a mirror in response to an electrical command, and the relationship between these two variables is linear for most galvonometers. The equations for angular displacement for the $x$ axis galvonometer are

$$\theta_y = \theta_{y0} \tag{7}$$

$$\theta_x = \theta_{x0} + e \cdot G \tag{8}$$

where $\theta_{y0}$ is angle between the reflected beam and the $x$ axis, which is usually 90°, $e$ is the input command in volts, $G$ is the input gain constant, in units of degrees of rotation per volt, and $\theta_{x0}$ is the rest position about the $x$ axis. Note that equation (8) is identical to equation (1). Equations for the y galvonometer are complementary.

Errors due to variations of $\theta_{y0}$ from 90° are known as *wobble*. Errors caused by variations of $G$ are known as *gain-drift* when they occur slowly over many minutes or hours, and as *linearity* when they are a function of $\theta_x$. Likewise, errors due to variations of $\theta_{x0}$ are known as *zero-drift* when they occur slowly and *repeatability* when they occur between successive positionings. The causes of these errors are electromechanical in nature and are discussed extensively elsewhere.[2]

Using galvonometers to perform laser eye scans adds inaccuracies in two ways. First, since the galvo is moving during a scan, a *lag* exists between the commanded position and the actual position. This lag is constant over the field. Second, the laser eye scan edge finding algorithm has inaccuracies, which are unfortunately difficult to distinguish from positioning repeatability.

## 6. CORRECTION OF POSITIONING ERROR SOURCES

How well a scanning system calibration performs is measured by how closely beam positioning accuracy approaches beam positioning repeatability. This is a natural consequence of the observation that any error that is *repeatable* is *correctable*. Certain practical aspects restrict the design and implementation of a correction scheme: a calibration should not take more than about 30 minutes to run; the scanning system should only need recalibration on the order of days or weeks; the calibration should execute with minimal operator intervention; calculation of corrected positions using calibration information must execute quickly enough to not slow down the system.

The calibration method discussed comprises a two-dimensional third-order least-squares estimation of system error and a bilinear interpolation of residual errors remaining after the regression *(gridmap)*. A method for aligning a device to be blasted to a calibrated field is also discussed.

### 6.1 Least-squares model of system error

Clearly, derivation of an exact closed-form solution of system error is extremely tedious. By using a Taylor series expansion to convert all trigonometric functions to polynomial form, total system error is instead approximated as a polynomial function of $(x, y)$. We choose a third order polynomial in both axes since contributions to trigonometric expressions from terms greater than third order are insignificant at the 18 bit accuracy level. We perform a least-squares approximation to determine coefficients of this polynomial.

The least-squares approximation provides the best solution to an overdetermined system of equations.[3] In matrix form, a system is given by

$$A \cdot cx = e \tag{9}$$

where the vector $cx$ must be solved to find the least-squares solution. Using this, we compute calibration coefficients for an axis by scanning $n$ $(x, y)$ locations around the beam field, filling the vector $e$ with the scan error in lsbs at each location, and solving the system

$$\begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & y_1 & x_1 y_1 & x_1^2 y_1 & \cdots & x_1^3 y_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 & y_2 & x_2 y_2 & x_2^2 y_2 & \cdots & x_2^3 y_2^3 \\ & & & & \vdots & & & & \\ 1 & x_n & x_n^2 & x_n^3 & y_n & x_n y_n & x_n y_n^2 & \cdots & x_n^3 y_n^3 \end{bmatrix} \cdot cx = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix} \tag{10}$$

The dimensions are $A^{N\times 16} \cdot cx^{16\times 1} = e^{N\times 1}$. This coefficient derivation must be computed for both axes, so a total of 32 calibration coefficients characterize system beam positioning error.

The evaluation of the coefficient polynomial at $(x, y)$ for an axis is the product of the row vector in the format of a row in $A$ above and the coefficient vector $cx$ (or $cy$). In summation notation,

$$\hat{C}_x(x,y) = \sum_{j=1}^{4} \sum_{i=1}^{4} cx_{i\cdot j} \cdot x^{i-1} \cdot y^{j-1} \tag{11}$$

Evaluation of this summation is not computationally burdensome; Horner's rule may be used to reduce the number of multiplies. DSP chips can evaluate this polynomial in under 35 microseconds.

Systems using the video eye scan may need an extra calibration step if the television illumination wavelength differs significantly from the laser wavelength. In this case, the reference to be scanned is not a grid but a series of lines actually cut by the laser. The same two-dimensional, third-order least-squares fit can be performed on the error data, which are the differences between the nominal commanded position and the position of the laser blast within the field of view. After the system is fully calibrated, the calibration correction only is applied when using the frame grabber to align the device to be blasted. When the laser is fired, both the calibration correction and the visual-laser correction must be applied to correctly position the laser blast. If the visual correction coefficients are in the vector $vx$, the expression for $x$ axis visual correction is

$$\hat{V}_x(x,y) = \sum_{j=1}^{4} \sum_{i=1}^{4} vx_{i\cdot j} \cdot x^{i-1} \cdot y^{j-1} \tag{12}$$

### 6.2 Gridmap correction

The least squares approximation cannot perfectly model the beam positioning system; in practice, the technique has modelled systems to a precision of 80 parts per million. The remaining error can be calculated by using a table known as a *gridmap*. A gridmap is simply a table (one for each axis) of the residual error remaining after the least-squares calculation. Its format is the sequence of tuples $(x, y, e)$ where the $(x, y)$ coordinates are those used in the least-squares calibration and $e$ is the residual error.

Using this table, a bilinear approximation is used to determine the residual error for any point in the field. A bilinear approximation uses the error values from the four table entries corresponding to $(x, y)$ positions surrounding the point to be evaluated. As an example, let $(x_{11}, y_{11})$ be the $(x, y)$ coordinates of the gridmap entry to the lower left of the point to be evaluated, and let $e_{11}$ be the corresponding gridmap error at that point. Likewise, let $(x_{12}, y_{12})$ be the lower right point, $(x_{21}, y_{21})$ be the upper left point, and $(x_{22}, y_{22})$ be the upper right point. The gridmap error function $\hat{G}(x, y)$ is given by the equation

$$\hat{G}(x,y) = e_{11} + x' \cdot \frac{\partial E}{\partial x} + y' \cdot \frac{\partial E}{\partial y} + x' \cdot y' \cdot \frac{\partial^2 E}{\partial x \partial y} \tag{13}$$

where $x' = \frac{x - x_{11}}{x_{12} - x_{11}}$, $y' = \frac{y - y_{11}}{y_{21} - y_{11}}$, $\frac{\partial E}{\partial x} = e_{12} - e_{11}$, $\frac{\partial E}{\partial y} = e_{21} - e_{11}$, and $\frac{\partial^2 E}{\partial x \partial y} = (e_{11} + e_{22}) - (e_{12} - e_{21})$.

As an example, let the lower left coordinate be $(10, 10)$ with table entry 5. Similarly, let the lower right coordinate be $(20, 10)$ with table entry 15, the upper left coordinate be $(10, 20)$ with table entry 10, and the upper right coordinate be $(20, 20)$ with table entry 25. Evaluating the function at $(12, 15)$ yields

$$\hat{G}(12,15) = 5 + \frac{12-10}{20-10}\cdot(15-5) + \frac{15-10}{20-10}\cdot(10-5) + \frac{(12-10)(15-10)}{(20-10)(20-10)}\cdot[(25+5)-(10+15)] = 10.$$

### 6.3 Field of view alignment correction

All mathematical procedures to calibrate a beam positioning system discussed thus far may be characterized as a transformation from the *nominal* coordinate system of the beam positioner to the *corrected* coordinate system defined by the reference grid. After calibration, all is well if every device to be blasted is placed in the field of view in exactly the orientation as the reference grid; obviously, this never happens, so an alignment transformation to map the *corrected* coordinate system to the device's *actual* coordinate system is required. This alignment adjusts any theta rotation and axis offset introduced by the placement of the device in the field of view, and additionally provides a fast first-order correction of any gain-drift present since calibration. Calibration interval is on the order of days, and alignment interval is on the order of seconds.

Five parameters describe the calibrated beam positioner to device alignment. Those parameters are $x$ *offset*, the difference between an $x$ position on the device and the beam positioner; $y$ *offset*, the corresponding difference for the $y$ axis; *theta*, the rotation of the device under the field of view; $x$ *scale*, the expansion or contraction of one coordinate system relative to the other, and $y$ *scale*, the corresponding effect for the $y$ axis. The notation $x_0, y_0, \theta, x_s, y_s$ will denote these parameters. Note that the $x$ and $y$ offsets not only correct device placement error, but also zero-drift galvonometer error.

These five parameters can be determined with five scans. The first two determine $x$ and $y$ offset. The third scan, measuring theta, is some distance away from either the first or the second to provide a reasonable baseline. Likewise, the two scans for $x$ scale and $y$ scale are also a reasonable distance away. The equations expressing alignment correction for $(x, y)$ may be simplified to

$$A_x(x,y) = (x \cdot x_s)cos\theta - (y \cdot y_s)sin\theta + x_0 \tag{14}$$

$$A_y(x,y) = (y \cdot y_s)cos\theta + (x \cdot x_s)sin\theta + y_0 \tag{15}$$

A better alignment parameter determination technique uses a least-squares fit of data taken from more than five scans. Solution of this overdetermined system yields a more precise approximation of the needed coordinate transformation.

### 6.4 Total system error correction

We now derive an expression to correct total system error. The input to the equation is an $(x, y)$ location, in system distance units; the output is an $(\tilde{x}, \hat{y})$ lsb value pair to send to the galvonometers.

The first step is alignment correction, a function whose output is distance units. (All successive equations have distance units for input values and lsb units for output values.) These aligned coordinates then undergo the conversion to a nominal lsb value, calibration correction, gridmap correction, and possibly visual correction. All outputs are summed and the result sent to the galvonometer. The equations for a position $(x, y)$ are therefore

$$\hat{X}_{galvo} = \hat{N}(x') + \hat{C}_x(x',y') + \hat{G}_x(x',y') + \hat{V}_x(x',y') \tag{16}$$

$$\hat{Y}_{galvo} = \hat{N}(y') + \hat{C}_y(x',y') + \hat{G}_y(x',y') + \hat{V}_y(x',y') \tag{17}$$

where $x' = A_x(x, y)$ and $y' = A_y(x, y)$.

## 7. CONCLUSION

Sources of beam positioning error within a pre-objective galvonometer based laser pointing system have been examined. Errors can be attributed to both system kinematics and to the behavior of the galvonometer; a generalized two-dimensional third-order least-squares approximation of all system error provides an extremely accurate and computationally efficient model for a calibration. Residual errors remaining after regression polynomial evaluation may be corrected with a gridmap, a table used for bilinear interpolation of residual error. In a calibrated system, an effective alignment method ensures that the coordinate systems of the system and the target device are equivalent. The execution speed of a software implementation of the entire beam positioning calibration and correction procedure is within the throughput requirements of typical laser beam positioning systems.

## 8. REFERENCES

1. John Weisz, "Calibrating Laser Trimming Apparatus", U.S. patent 4,918,284.

2. Pierre Brosens, "Scanning Accuracy of the Moving-Iron Galvonometer Scanner", Optical Engineering, vol. 15 no. 2, pp. 95-98, March-April 1975.

3. Gilbert Strang, Linear Algebra and its Applications, Harcourt Brace Jovanovich, San Diego, 1988.

Before proceeding, two metrics needed to understand system accuracy are *edge detection* repeatability and *beam positioning* repeatability. Edge detection repeatability is the measure of how much error is introduced solely by the edge detection mechanism. It is only determinable if the galvonometers are stationary during an edge sense, since any galvonometer motion introduces its own error. Likewise, beam positioning repeatability is the measure of how accurately a galvo can position to the same location.

A positioning system's only capability of ultimate importance is the ability to position a laser blast. Hence, an edge sense technique must lend itself as directly as possible to that end. This concept, as well as those of edge detection repeatability and beam positioning repeatability, are critical to understanding the tradeoffs of the following edge detection methods.

### 4.1 Laser eye scan

This scan is performed by repeatedly pulsing the laser (at non-destructive energies) as one galvonometer is swept over the target area. A photodiode detects the reflected laser energy for each pulse, producing a laser reflectance vs. position profile that is analyzed to detect the edge position. Thus, a good "edge" is any surface feature that exhibits a rapid change in laser reflectivity.

Since the laser beam itself gathers edge location information, the edge position corresponds exactly to laser pulse position. This is the primary advantage of this technique. The disadvantage of this method arises from the movement of the galvo during the scan. Because of this motion, edge detection repeatability cannot be determined; so, for a laser eye scan, beam positioning repeatability and edge detection repeatability are synonymous.

### 4.1 Video eye scan

This edge finding method requires a video processor, or "frame grabber", to digitize a frame from a video camera positioned over the field of view. Reflectance data from the frame are analyzed and edge features determined; edge positions are calculated in pixels.

This method can distinguish edge detection repeatability from beam positioning repeatability, since the galvos are at rest during a scan. Unfortunately, this technique has a serious drawback: the edge position is returned in pixels, and no a priori relationship exists between a pixel position and the position at which a laser blast will appear. This is the so-called *bore-sighting problem*. The problem is compounded if the laser wavelength and the visual illumination wavelength are of significantly different frequencies, since the lens will refract each differently. In this case, an extra step is required in the calibration process to correct between the video frame of reference and the laser blast position over the field.[1]

## 5. POSITIONING ERROR SOURCES

For this discussion, system errors have been partitioned into two groups: those induced by the galvonometers, and those caused by everything else in the system. The second group will be referred to as *system configuration* errors.

Before proceeding, a comment concerning notation. All functions will be denoted with uppercase letters, and all arguments to functions in lowercase letters. A function returning a value in lsbs will have a ˆ above its name; a quantity in lsbs will likewise be denoted. Quantities expressed in distance units will be expressed as a letter only.

Positioning errors are defined as the difference between the nominal beam position and the actual beam position. Determination of the nominal lsb position is given by the equation

$$\hat{N}(x) = \frac{\kappa}{2} + x \cdot \frac{\kappa}{\kappa'} \tag{1}$$

where $x$ is the desired distance, $\kappa$ is the field size in beam positioner lsbs, and $\kappa'$ is the field size in distance units. For example, if the field size is 20000 microns, the lsb value required to position at $x = -1000$ microns is

$$\hat{N}(-1000) = \frac{2^{18}}{2} + (-1000) \cdot \frac{2^{18}}{20000} = 117965.$$

## 5.1 Errors caused by scanning system configuration

*Pincushion Distortion.* This error is caused by successive rotations of the laser beam by the galvo mirrors. We characterize it by examining the displacement of a vector as it is transformed by the rotation matrices that model the beam positioner.

First, an initial unit vector along the $z$ axis is rotated about the $y$ axis, which results in displacement about the $x$ axis:

$$\begin{bmatrix} cos\theta_x & 0 & sin\theta_x \\ 0 & 1 & 0 \\ -sin\theta_x & 0 & cos\theta_x \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} sin\theta_x \\ 0 \\ cos\theta_x \end{bmatrix} \tag{2}$$

Rotation about the $x$ axis follows, effecting displacement along the $y$ axis:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta_y & sin\theta_y \\ 0 & -sin\theta_y & cos\theta_y \end{bmatrix} \cdot \begin{bmatrix} sin\theta_x \\ 0 \\ cos\theta_x \end{bmatrix} = \begin{bmatrix} sin\theta_x \\ sin\theta_y cos\theta_x \\ cos\theta_y cos\theta_x \end{bmatrix} \tag{3}$$

Since the beam is focussed to a plane perpendicular to the $z$ axis, the $z$ component of this vector is made unity, resulting in a final position expression

$$\begin{bmatrix} tan\theta_x / cos\theta_y \\ tan\theta_y \\ 1 \end{bmatrix} \tag{4}$$

This is the expression for position, not error. The linear displacement on a plane for a projection angle $\theta$ and unity distance is $tan\theta$, so the result for the $y$ axis position is expected; pincushion distortion contributes no error along that axis. The x axis, however, has an error of

$$P_x(\theta_x, \theta_y) = tan\theta_x - tan\theta_x / cos\theta_y \tag{5}$$

Pincushion error is dependent on the order in which the galvos are struck, an expected result since three dimensional rotation matrix multiplication is not commutative. All distances are unity in the expression, so a scaling factor based on the distance along the $z$ axis to the focal plane must be present if calculating the exact magnitude of this error. Pictorially, this distortion results in a slight inward bowing along the $x$ axis.

*Tangent Error.* For scanning systems with an $f \cdot tan\theta$ lens, i.e. a lens that projects an image at a distance proportional to the tangent of the entrance angle, an error is caused by the nonlinearity of the tangent function:

$$T(\theta) = \ell \cdot (\theta - tan\theta) \tag{6}$$

where $\ell$ is the focal length of the lens. Scanning systems with an $f \cdot \theta$ lens do not exhibit this behavior.

*Orthogonality Error.* Galvonometers can rarely be mounted so that their mirrors are perfectly orthogonal to one another; this error identifies the angular offset from the first galvo by the second galvo.

*Bore-sight Error.* As mentioned, this error exists in systems using the video eye scan edge detection technique. The error is the difference between the edge detection reference point and the laser positioning reference point.

*Video Eye Scan Error.* This is the repeatability error of the video eye scan edge detection procedure.

J Math Imaging Vis (2009) 35: 155–164
DOI 10.1007/s10851-009-0161-2

# Metrics for 3D Rotations: Comparison and Analysis

**Du Q. Huynh**

Published online: 18 June 2009

**Abstract** 3D rotations arise in many computer vision, computer graphics, and robotics problems and evaluation of the distance between two 3D rotations is often an essential task. This paper presents a detailed analysis of six functions for measuring distance between 3D rotations that have been proposed in the literature. Based on the well-developed theory behind 3D rotations, we demonstrate that five of them are bi-invariant metrics on $SO(3)$ but that only four of them are boundedly equivalent to each other. We conclude that it is both spatially and computationally more efficient to use quaternions for 3D rotations. Lastly, by treating the two rotations as a true and an estimated rotation matrix, we illustrate the geometry associated with iso-error measures.

**Keywords** Matrix Lie group · Lie algebra · Quaternions · 3D rotations · Distance functions

## 1 Introduction

3D rotations are common entities in many computer vision, computer graphics, and robotics problems that need to deal with the 3D world. Typical applications that involve 3D rotations include the interpolation of trajectory of 3D orientations, robot kinematics, flight simulation, structure from motion, 3D pose recovery of objects, and motion capture. The common issues that arise in these applications are how to efficiently represent 3D rotations and how to correctly evaluate the distance between them. If one of the 3D rotation matrices is a true or reference rotation while the other is an estimated one, then it is useful also to identify regions where identical error measures occur. A few functions for distance measures between 3D rotations have been proposed in the computer vision literature; however, there has neither been detailed analysis provided to these functions nor a comparison of them in the context of the group $SO(3)$ to which 3D rotations belong. The contributions of this paper are: (1) to provide this missing information; (2) to analyze and illustrate the iso-error contours of a given reference rotation.

Rotations in 3D space can be represented in various forms. Euler angles are commonly used in robotics applications where, because of constraints in the design of the joints of robot arms, rotations often have to be carried out in a certain order (e.g., [1]; see also [19]). For other applications where such a constraint is absent, Euler angles are less favoured, precisely because the values of these angles are dependent on the order of rotations about the three principal axes. For research in computer vision and computer graphics, 3D rotations are commonly represented as unit quaternions (e.g., [5, 15, 17, 18]), rotation axes and angles (e.g., [8, 21]), or even as the $3 \times 3$ rotation matrices themselves (e.g., [3, 4, 6]). The number of computer vision research papers that involve 3D rotations of any form is far too many to permit a complete list of citations. The references given here are only a very small subset of papers in the literature.

In terms of storage, each $3 \times 3$ rotation matrix requires the space of 9 floating point numbers, whereas in reality the special orthogonal group is a 3 dimensional object embeddable in $\mathbb{R}^4$. Another concern of the matrix representation is that after several matrix multiplications, round-off errors within computers can result in "rotation" matrices that are

This research was in part supported by a UWA study leave grant.

D.Q. Huynh (✉)
School of Computer Science and Software Engineering,
The University of Western Australia, Nedlands,
WA 6009, Australia
e-mail: du@csse.uwa.edu.au

no longer orthogonal. In that regard, the unit quaternions are preferred as an alternative way of representing 3D rotations. Although round-off errors may also cause any unit quaternion to have a non-unit magnitude, it is more straightforward to renormalize it to unity in comparison with re-orthogonalizing a distorted matrix.

In the following sections, we will first give a brief overview of the special orthogonal group and the unit quaternions. This is then followed by the formal definition of a distance function or metric. Six different functions for 3D rotations will then be studied in turn; their computation complexity will be briefly analyzed. In Sect. 5, a geometrical interpretation of one of the functions will be studied further, this leads to the discussion on iso-error contours. Finally in Sect. 6, we conclude the paper.

## 2 *SO*(3) and Unit Quaternions: An Overview

3D rotations form the so-called Special Orthogonal Group $SO(3)$ of orthogonal matrices with determinant 1. $SO(3)$ is a compact Lie group having the skew-symmetric matrices as its Lie algebra, $so(3)$. This Lie algebra is a non-associative vector space equipped with a binary operation:

$$\begin{aligned}&[\cdot,\cdot]: so(3)\times so(3)\to so(3),\\ &[\mathbf{A},\mathbf{B}]=\mathbf{AB}-\mathbf{BA}\end{aligned} \tag{1}$$

which can easily be seen to be a closed operation in $so(3)$. The Lie algebra $so(3)$ is the tangent space at the identity element of $SO(3)$. The binary operation defined above is known as the *Lie bracket*, which satisfies the following properties:

$$\begin{aligned}&[\mathbf{A},\mathbf{B}]=-[\mathbf{B},\mathbf{A}],\\ &[\mathbf{A},[\mathbf{B},\mathbf{C}]]+[\mathbf{C},[\mathbf{A},\mathbf{B}]]+[\mathbf{B},[\mathbf{C},\mathbf{A}]]=0,\end{aligned} \tag{2}$$

for all $\mathbf{A},\mathbf{B},\mathbf{C}\in so(3)$. It follows immediately from the first property that $[\mathbf{A},\mathbf{A}]=0$.

In the theory of Lie groups, the exponential map is a mapping from the Lie algebra of a Lie group to the group itself. Such a mapping allows one to recapture the group structure from its Lie algebra. As in the general case of matrix Lie groups, the exponential map $\exp: so(3)\to SO(3)$ is simply:

$$\exp(\mathbf{A})=\mathbf{I}+\mathbf{A}+\frac{\mathbf{A}^2}{2!}+\frac{\mathbf{A}^3}{3!}+\cdots. \tag{3}$$

For the special orthogonal group $SO(3)$, the exponential map is surjective but not injective. The failure of injectivity is easily seen by considering a skew-symmetric matrix such as

$$\mathbf{A}=\begin{bmatrix}0&1&1\\-1&0&1\\1&-1&0\end{bmatrix} \tag{4}$$

and noting that $\exp(t\mathbf{A})=\mathbf{I}$ whenever $t$ is an integer multiple of $2\pi/\sqrt{3}$. This example, however, suggests at least some of the degree of non-uniqueness in the exponential parametrization. Indeed, the exponential parametrization of $SO(3)$ corresponds exactly to the rotation axis and rotation angle formulation. Given a rotation with rotation axis $\mathbf{u}=(u_1,u_2,u_3)^\top$ of unit magnitude and rotation angle $\theta$, the $3\times3$ rotation matrix $\mathbf{R}$ can be obtained by the exponential mapping $\exp([\theta\mathbf{u}]_\times)$, where $[\mathbf{a}]_\times$ is chosen to be the matrix defined by $[\mathbf{a}]_\times\mathbf{b}=\mathbf{a}\times\mathbf{b}$. For example, the matrix in (4) corresponds to $[(-1,1,-1)]_\times$. The group $SO(3)$ is covered by one-parameter groups (in fact circles) of the form $\{\exp([\theta\mathbf{u}]_\times):\theta\in[-\pi,\pi)\}$ and this representation is almost unique, since a rotation uniquely specifies its (unit vector) rotation axis up to a multiplication by ±1, and once this is fixed, the angle is specified up to a multiple of $2\pi$.

It is straightforward to show that the Rodrigues formula (see, e.g., [11]) for a rotation matrix $\mathbf{R}$ as defined below

$$\mathbf{R}=\cos\theta\,\mathbf{I}+\sin\theta\,[\mathbf{u}]_\times+(1-\cos\theta)\mathbf{u}\mathbf{u}^\top, \tag{5}$$

is just a simplification of the exponential map given in (3) being applied to $[\theta\mathbf{u}]_\times$. Given a $3\times3$ rotation matrix $\mathbf{R}$, the inverse of the exponential map provides *a* rotation angle/axis description of the rotation. Thus, $\log(\mathbf{R})$ is the skew-symmetric matrix containing information about the rotation axis and angle. Although the inverse process requires a choice of rotation axis between the two alternatives, it is a straightforward procedure to retrieve the rotation axis and angle (see Appendix A).

As a unit quaternion, the same 3D rotation matrix $\exp([\theta\mathbf{u}]_\times)\in SO(3)$ can be written as $\mathbf{q}\doteq(q_0,q_1,q_2,q_3)^\top\equiv(q_0,\tilde{\mathbf{q}})^\top$
$=\pm(\cos\frac{\theta}{2},\mathbf{u}^\top\sin\frac{\theta}{2})^\top$. The unit quaternions are a one-to-one parametrization of the Special Unitary group $SU(2)$ and the $\mathbf{q}$ above can be written as a $2\times2$ unitary matrix:

$$\begin{bmatrix}q_0+iq_1&q_2+iq_3\\-q_2+iq_3&q_0-iq_1\end{bmatrix}. \tag{6}$$

$SU(2)$ provides a double-covering of $SO(3)$; that is, each rotation matrix in $SO(3)$ corresponds to two members of $SU(2)$. The group homomorphism from $SU(2)$ to $SO(3)$ has a two element kernel and this corresponds to the ambiguity in the choice of rotation axis in the logarithmic map. In this regard, we note that $SU(2)$ and $SO(3)$ have the same Lie algebra. The Lie algebra $su(2)$, which consists of the $2\times2$ skew-hermitian matrices, is isomorphic as a Lie algebra to $so(3)$. Consider representing the rotation axis $\mathbf{u}=(u_1,u_2,u_3)^\top$ of unit magnitude and the rotation angle $\theta$ as a $2\times2$ skew-hermitian matrix, $\mathbf{A}$, as follows:

$$\mathbf{A}=\begin{bmatrix}i\theta u_1&\theta u_2+i\theta u_3\\-\theta u_2+i\theta u_3&-i\theta u_1\end{bmatrix}$$

$$= \theta \begin{bmatrix} iu_1 & u_2 + iu_3 \\ -u_2 + iu_3 & -iu_1 \end{bmatrix} \equiv \theta \tilde{\mathbf{A}}. \tag{7}$$

By applying the exponential mapping to $\mathbf{A}$ and noting that $\mathbf{A}^2 = -\theta^2 \mathbf{I}$, we obtain a different version of the Rodrigues formula:

$$\exp : su(2) \to SU(2),$$

$$\begin{aligned} \exp(\mathbf{A}) &= \cos\theta\, \mathbf{I} + \frac{\sin\theta}{\theta} \mathbf{A} \\ &= \cos\theta\, \mathbf{I} + \sin\theta\, \tilde{\mathbf{A}}. \end{aligned} \tag{8}$$

It is straightforward to verify that $\exp(\mathbf{A}) \in SU(2)$ and that the quaternions are simply a parametrization of $SU(2)$. The exponential map from $su(2)$ to $SU(2)$ does not suffer from the ambiguity present in the orthogonal case.

## 3 Distance Function or Metric

### 3.1 Definition

For definiteness, we give the usual definition of *distance function* or *metric*. Let $S$ be some space between whose elements we are interested in knowing distances. A *distance function* or *metric* is a map $\Phi : S \times S \to \mathbb{R}^+$ satisfying the usual axioms for metrics:

(i) $\Phi(x, y) = 0 \Leftrightarrow x = y$;
(ii) $\Phi(x, y) = \Phi(y, x)$ for $x, y \in S$;
(iii) $\Phi(x, z) \le \Phi(x, y) + \Phi(y, z)$ for $x, y, z \in S$.

We note that there are situations where our intuitive concept of "distance" may not satisfy these axioms.

There are two other properties of distance functions that are important in the context of $SO(3)$. The first is that the distance function defines and respects the topology of $SO(3)$. To state precisely what this means, we note that the obvious definition of $\mathbf{R}_n \to \mathbf{R}$ as $n \to \infty$ for $\mathbf{R}_n, \mathbf{R} \in SO(3)$ is that the matrix entries of the $\mathbf{R}_n$ should converge to the matrix entries of $\mathbf{R}$ as real numbers. We say that a distance function $\Phi$ respects the topology of $SO(3)$ provided

$$\Phi(\mathbf{R}_n, \mathbf{R}) \to 0 \quad \Leftrightarrow \quad \mathbf{R}_n \to \mathbf{R}. \tag{9}$$

Since $SO(3)$ is compact, it is sufficient [12] that

$$\mathbf{R}_n \to \mathbf{R} \quad \Rightarrow \quad \Phi(\mathbf{R}_n, \mathbf{R}) \to 0. \tag{10}$$

It is essential that this is the case for all distance functions defined on $SO(3)$.

Since $SO(3)$ is a group, we can ask one other property of interest. We say that a distance function $\Phi$ is *left, (resp. right) invariant* if

$$\Phi(\mathbf{R}_1\mathbf{R}_2, \mathbf{R}_1\mathbf{R}_3) = \Phi(\mathbf{R}_2, \mathbf{R}_3), \tag{11}$$

$$\left(\text{resp. } \Phi(\mathbf{R}_2\mathbf{R}_1, \mathbf{R}_3\mathbf{R}_1) = \Phi(\mathbf{R}_2, \mathbf{R}_3)\right) \tag{12}$$

for $\mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3 \in SO(3)$. A distance function is *bi-invariant* if it is both left and right invariant.

Two distance functions $\Phi$ and $\Psi$ are said to be *boundedly equivalent* if there are positive real numbers $a$ and $b$ such that

$$a\Phi(\mathbf{R}_1, \mathbf{R}_2) \le \Psi(\mathbf{R}_1, \mathbf{R}_2) \le b\Phi(\mathbf{R}_1, \mathbf{R}_2) \tag{13}$$

for all $\mathbf{R}_1$, $\mathbf{R}_2 \in SO(3)$. We also define *functional equivalence* between $\Phi$ and $\Psi$ to mean that there exists a positive continuous strictly increasing function $h$ such that

$$h \circ \Phi = \Psi. \tag{14}$$

It is easy to see that if one distance function respects the topology of $SO(3)$ and another is either boundedly or functionally equivalent to it then it too respects the topology. Also, if $\Phi$ is a distance function satisfying

$$c\Phi(\mathbf{R}_1, \mathbf{R}_2) \le \Phi(\mathbf{R}\mathbf{R}_1, \mathbf{R}\mathbf{R}_2) \le d\Phi(\mathbf{R}_1, \mathbf{R}_2) \tag{15}$$

for all $\mathbf{R}_1, \mathbf{R}_2, \mathbf{R} \in SO(3)$ and for some positive real numbers $c$ and $d$, then it is equivalent to a left-invariant one. The invariant distance function is

$$\Phi_L(\mathbf{R}_1, \mathbf{R}_2) = \int \Phi(\mathbf{R}\mathbf{R}_1, \mathbf{R}\mathbf{R}_2)\, d\mathbf{R} \tag{16}$$

for all $\mathbf{R}_1, \mathbf{R}_2 \in SO(3)$ where the integral is with respect to the (left) Haar measure on the group. We note that this measure is left invariant and it is this property which ensures left invariance of the distance function. A similar result holds for right and bi-invariant distance functions. Note that, in this case, since $SO(3)$ is a compact group its left and right Haar measures are the same.

Finally, two metrics are said to be *topologically equivalent* if they give the same convergent sequences. Both *functional* and *bounded equivalence* individually imply *topological equivalence*. Our assumption that metrics respect the topology means that they are topologically equivalent to each other.

Several functions for measuring the *distance* between two 3D rotations for various applications have been reported in the literature. Some of these functions are defined in terms of Euler angles or quaternions, while others involve the rotation matrices. We will discuss each of them in turn below. We assume that the two rotation matrices whose distance is of interest are $\mathbf{R}_1 = \exp([\theta_1 \mathbf{u}]_\times)$ and $\mathbf{R}_2 = \exp([\theta_2 \mathbf{v}]_\times)$, where $\theta_1$ and $\theta_2$ are the rotation angles, and $\mathbf{u}$ and $\mathbf{v}$ the rotation axes of unit magnitude. Their corresponding unit quaternions can be written as $\mathbf{q}_1 = \pm(\cos\frac{\theta_1}{2}, \mathbf{u}^\top \sin\frac{\theta_1}{2})^\top$ and $\mathbf{q}_2 = \pm(\cos\frac{\theta_2}{2}, \mathbf{v}^\top \sin\frac{\theta_2}{2})^\top$, respectively. All rotation axes are assumed to be of unit magnitude from here on.

### 3.2 Euclidean Distance between the Euler Angles

Let $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$ be two sets of Euler angles. Then

$$\begin{aligned}&\Phi_1 : E \times E \to \mathbb{R}^+,\\&\Phi_1((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2))\\&\quad = \sqrt{d(\alpha_1, \alpha_2)^2 + d(\beta_1, \beta_2)^2 + d(\gamma_1, \gamma_2)^2},\end{aligned} \tag{17}$$

where $E \subset \mathbb{R}^3$ is an appropriate domain for the three Euler angles (see the discussion that follows) and $d(a, b) = \min\{|a - b|, 2\pi - |a - b|\}$ denotes the normalized difference between the two angles so that $0 \leq d(.,.) \leq \pi$. The range of values of $\Phi_1$ is $[0, \pi\sqrt{3}]$. This function was discussed in [9] in the context of rotation matrix sampling. Unfortunately, it is not a distance function on $SO(3)$ since it depends on a representation that is not unique. For example, the Euler angles $(\alpha, \beta, \gamma) = (\pi, \pi, 0)$ can represent the same rotation as $(\alpha, \beta, \gamma) = (0, 0, \pi)$ under a particular order of rotations, yet their distance is non-zero. To overcome the problem of ambiguous representation, we can impose the following conditions on the Euler angles (and thus the domain $E$): $\alpha, \gamma \in [-\pi, \pi)$; $\beta \in [-\pi/2, \pi/2)$. Under this representation, $\Phi_1$ is a metric on $SO(3)$.

### 3.3 Norm of the Difference of Quaternions

Ravani and Roth [16] define the distance between two rotations as the Euclidean distance between two unit quaternions. As unit quaternions $\mathbf{q}$ and $-\mathbf{q}$ denote the same rotation, we can define the following function, which takes into account the ambiguity in quaternion representation:

$$\begin{aligned}&\Phi_2 : S^3 \times S^3 \to \mathbb{R}^+,\\&\Phi_2(\mathbf{q}_1, \mathbf{q}_2) = \min\{\|\mathbf{q}_1 - \mathbf{q}_2\|, \|\mathbf{q}_1 + \mathbf{q}_2\|\},\end{aligned} \tag{18}$$

where $\|\cdot\|$ denotes the Euclidean norm (or 2-norm) and $S^3 = \{\mathbf{q} \in \mathbb{R}^4 \mid \|\mathbf{q}\|^2 = 1\}$. We can easily verify that in the $S^3$ space, the $\Phi_2$ function above satisfies all axioms except for Axiom (i) since $\Phi_2(\mathbf{q}, -\mathbf{q}) = 0 \not\Rightarrow \mathbf{q} = -\mathbf{q}$ (as vectors in $\mathbb{R}^4$). This means that $\Phi_2$ is a *pseudometric* [7] rather than a metric on the unit quaternions. However, as the mapping from unit quaternions to $SO(3)$ is 2-to-1, the pseudometric on the unit quaternions becomes a metric on 3D rotations because we are identifying points with zero distance apart.

The $\Phi_2$ metric gives values in the range $[0, \sqrt{2}]$.

### 3.4 Inner Product of Unit Quaternions

A similar function that involves unit quaternions is given by:

$$\begin{aligned}&\Phi'_3 : S^3 \times S^3 \to \mathbb{R}^+,\\&\Phi'_3(\mathbf{q}_1, \mathbf{q}_2) = \min\{\arccos(\mathbf{q}_1 \cdot \mathbf{q}_2), \pi - \arccos(\mathbf{q}_1 \cdot \mathbf{q}_2)\},\end{aligned}$$

where $\cdot$ denotes the inner (or dot) product of vectors (not the quaternion multiplication, which produces another quaternion). This function was used by Wunsch et al. [20] for 3D object pose estimation. As in (18), the ambiguity in sign of unit quaternions must be taken into consideration. So, $\Phi'_3$ can be replaced by the following computationally more efficient function:

$$\begin{aligned}&\Phi_3 : S^3 \times S^3 \to \mathbb{R}^+,\\&\Phi_3(\mathbf{q}_1, \mathbf{q}_2) = \arccos(|\mathbf{q}_1 \cdot \mathbf{q}_2|).\end{aligned} \tag{19}$$

Since it is necessary that $\Phi_3$ is a non-negative function, we restrict the angles returned by arccos to be in the first quadrant, i.e., the range of values mapped by $\Phi_3$ is $[0, \pi/2]$ (radians).

Alternatively, the inverse cosine function above can be eliminated by defining

$$\begin{aligned}&\Phi_4 : S^3 \times S^3 \to \mathbb{R}^+,\\&\Phi_4(\mathbf{q}_1, \mathbf{q}_2) = 1 - |\mathbf{q}_1 \cdot \mathbf{q}_2|.\end{aligned} \tag{20}$$

This function was used in [9] for the distance measure between two Euclidean transformations. Function $\Phi_4$ give values in the range [0, 1].

Following the same argument as that for $\Phi_2$, we can conclude that both $\Phi_3$ and $\Phi_4$ are pseudometrics on the unit quaternions but are metrics on $SO(3)$. Up to this point, we have not discussed whether $\Phi_2$, $\Phi_3$, and $\Phi_4$ are bi-invariant metrics on $SO(3)$. We defer the proof for this until later in the paper.

### 3.5 Deviation from the Identity Matrix

Larochelle et al. [10] use polar decomposition to approximate elements of the Euclidean group $SE(n-1)$ with elements of the special orthogonal group $SO(n)$ and then employ the metric $d(\mathbf{A}_1, \mathbf{A}_2) = \|\mathbf{I} - \mathbf{A}_1\mathbf{A}_2^\top\|_F$ (where $\mathbf{A}_1, \mathbf{A}_2 \in SO(n)$ and $\|\cdot\|_F$ denotes the Frobenius norm of the matrix) as a distance measure between two rigid body displacements. For the specific case where $n = 3$, we have

$$\begin{aligned}&\Phi_5 : SO(3) \times SO(3) \to \mathbb{R}^+,\\&\Phi_5(\mathbf{R}_1, \mathbf{R}_2) = \|\mathbf{I} - \mathbf{R}_1\mathbf{R}_2^\top\|_F,\end{aligned} \tag{21}$$

which gives values in the range $[0, 2\sqrt{2}]$. An alternative is to replace the Frobenius norm above by the 2-norm to reduce the range of values to $[0, 2]$ instead.

One can verify that $\Phi_5$ is a metric on $SO(3)$ although the proof to show that the function satisfies the triangle inequality condition involves some messy algebra.

### 3.6 Geodesic on the Unit Sphere

Since $SO(3)$ is a compact Lie group it has a natural Riemannian metric; that is, an inner product on its tangent space at every point. At the identity, this tangent space is $so(3)$, i.e., the skew-symmetric matrices, as we have mentioned. The inner product on $so(3)$ is given by

$$\langle \mathbf{S}_1, \mathbf{S}_2 \rangle = \frac{1}{2}\text{trace}\left(\mathbf{S}_1^\top \mathbf{S}_2\right), \tag{22}$$

for $\mathbf{S}_1, \mathbf{S}_2 \in so(3)$.

The inner product in the Riemannian structure provides an "infinitesimal" version of length on the tangent vectors, and so the length of a curve can be obtained by integration along the curve. Then the concept of a shortest path between two points on the group, a geodesic, follows. In fact, it is enough to describe the shortest path from the identity of the group to another point which we can write as $\exp(\mathbf{S})$, where $\mathbf{S} \in so(3)$. This shortest path can be shown to be of the form $\exp(t\mathbf{S})$, where $0 \le t \le 1$. We can use this to define a metric on the group by making the distance between two points the length of the geodesic between them. This metric is the one considered by Park and Ravani [13, 14]:

$$\begin{aligned} &\Phi_6 : SO(3) \times SO(3) \to \mathbb{R}^+, \\ &\Phi_6(\mathbf{R}_1, \mathbf{R}_2) = \|\log(\mathbf{R}_1\mathbf{R}_2^\top)\|, \end{aligned} \tag{23}$$

where, as described above, $\log(\mathbf{R})$ gives the skew-symmetric matrix that embodies the rotation axis and angle of the rotation matrix $\mathbf{R}$. The $\|.\|$ above therefore gives the magnitude of the rotation angle. This $\Phi_6$ function is a bi-invariant metric on $SO(3)$. The proof for this will be given later in the paper. The metric gives values in the range $[0, \pi)$.

It is obvious that both $\Phi_5$ and $\Phi_6$ attempt to find the amount of rotation required to bring $\mathbf{R}_1$ to align with $\mathbf{R}_2$, i.e., to find $\mathbf{R}$ such that $\mathbf{R}_1 = \mathbf{R}\mathbf{R}_2$, thus $\mathbf{R} = \mathbf{R}_1\mathbf{R}_2^\top$.

## 4 Comparison of the Metrics

One can verify that all the functions $\Phi_i$, for $i = 1, \ldots, 6$, defined above are metrics on $SO(3)$, although the proof for the triangle inequality condition may not be straightforward for some of them (a proof to show that $\Phi_6$ is a metric is given in Appendix B). It should be noted also that the $\Phi_1$ function does not truly reflect the 'distance' of two rotations. That is, two nearby rotations may have a large $\Phi_1$ value, while two distant rotations may have a smaller $\Phi_1$ value. For this reason, when a distance measure between two rotations is sought, any of the $\Phi_2$ to $\Phi_6$ metrics should be used instead. From here on, the comparison will be focused only on these five metrics.

### 4.1 Bounded Equivalence

The metrics $\Phi_i$, for $i = 2, \ldots, 6$, produce values in different ranges and of different units: $\Phi_3$ and $\Phi_6$ are in radians while the other three are dimensionless. The difference in units is not an issue as a change of unit merely results in a scale change to the metric being considered. Furthermore, the relationships among these metrics are clearly non-linear, except for $\Phi_3$ and $\Phi_6$. To see that $\Phi_3$ and $\Phi_6$ have a linear relationship, consider the computation of the rotation angle of $\mathbf{R}_1\mathbf{R}_2^\top$ in the definition of $\Phi_6$. As unit quaternions, $\mathbf{R}_1$ and $\mathbf{R}_2^\top$ can be, respectively, represented by $\mathbf{q}_1 = \pm(\cos\frac{\theta_1}{2}, \mathbf{u}^\top \sin\frac{\theta_1}{2})^\top$ and $\bar{\mathbf{q}}_2 = \pm(\cos\frac{\theta_2}{2}, -\mathbf{v}^\top \sin\frac{\theta_2}{2})^\top$, where the overhead bar denotes quaternion conjugate. The cosine of half of the rotation angle of $\mathbf{R}_1\mathbf{R}_2^\top$ can be found from the first component of product $|\mathbf{q}_1\bar{\mathbf{q}}_2|$, which, by definition of quaternion multiplication, is $|\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} + \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}(\mathbf{u}\cdot\mathbf{v})|$. This term is identical to the absolute value of the dot product of the two 4-vector $\mathbf{q}_1$ and $\mathbf{q}_2$ computed in $\Phi_3$, i.e., $\cos(\frac{\theta}{2}) = |\mathbf{q}_1 \cdot \mathbf{q}_2|$. Thus, $\Phi_3$ returns half of the rotation angle of $\mathbf{R}_1\mathbf{R}_2^\top$ from $\Phi_6$. From the definition given in Sect. 3.1, metrics $\Phi_3$ and $\Phi_6$ are boundedly equivalent.

To prove that two metrics $\Phi_i$ and $\Phi_j$ are boundedly equivalent (see (13)), it suffices, because $SO(3)$ is compact, to show that $\Phi_i/\Phi_j$ is bounded near the origin. In particular, the metrics $\Phi_2$ and $\Phi_5$ can be rewritten as

$$\Phi_2(\mathbf{q}_1, \mathbf{q}_2) = \sqrt{2(1 - |\mathbf{q}_1 \cdot \mathbf{q}_2|)}, \tag{24}$$

$$\begin{aligned} \Phi_5(\mathbf{R}_1, \mathbf{R}_2) &= \sqrt{\text{trace}\left((\mathbf{I} - \mathbf{R}_1\mathbf{R}_2^\top)^\top(\mathbf{I} - \mathbf{R}_1\mathbf{R}_2^\top)\right)} \\ &= \sqrt{2\,\text{trace}\,(\mathbf{I}) - 2\,\text{trace}\left(\mathbf{R}_1\mathbf{R}_2^\top\right)} \\ &= \sqrt{2\left(3 - \text{trace}\left(\mathbf{R}_1\mathbf{R}_2^\top\right)\right)}. \end{aligned} \tag{25}$$

Let $\mathbf{R}_1\mathbf{R}_2^\top = \exp([\theta\mathbf{u}]_\times)$. Then $\text{trace}\left(\mathbf{R}_1\mathbf{R}_2^\top\right) = 1 + 2\cos(\theta)$ (see Appendix A). Substituting this into (25) gives

$$\Phi_5(\mathbf{R}_1, \mathbf{R}_2) = 2\sqrt{1 - \cos(\theta)} = 2\sqrt{2 - 2\cos^2(\theta/2)}. \tag{26}$$

From the discussion given above, we have $\cos(\theta/2) = |\mathbf{q}_1 \cdot \mathbf{q}_2|$. This gives us an alternative distance function defined in terms of unit quaternions:

$$\begin{aligned} &\Phi_5' : S^3 \times S^3 \to \mathbb{R}^+, \\ &\Phi_5'(\mathbf{q}_1, \mathbf{q}_2) = 2\sqrt{2(1 - |\mathbf{q}_1 \cdot \mathbf{q}_2|^2)}. \end{aligned} \tag{27}$$

By taking $\Phi_3$ as the reference metric and applying L'Hôpital's rule, it can be shown that

$$\lim_{|\mathbf{q}_1\cdot\mathbf{q}_2|\to 1} \frac{\Phi_3}{\Phi_2} = \lim_{|\mathbf{q}_1\cdot\mathbf{q}_2|\to 1} \frac{\arccos(|\mathbf{q}_1 \cdot \mathbf{q}_2|)}{\sqrt{2(1 - |\mathbf{q}_1 \cdot \mathbf{q}_2|)}} = 1, \tag{28}$$

$$\lim_{|\mathbf{q}_1\cdot\mathbf{q}_2|\to 1}\frac{\Phi_3}{\Phi_4}=\lim_{|\mathbf{q}_1\cdot\mathbf{q}_2|\to 1}\frac{\arccos(|\mathbf{q}_1\cdot\mathbf{q}_2|)}{1-|\mathbf{q}_1\cdot\mathbf{q}_2|}=\infty, \tag{29}$$

$$\begin{aligned}\lim_{|\mathbf{q}_1\cdot\mathbf{q}_2|\to 1}\frac{\Phi_3}{\Phi_5}&=\lim_{|\mathbf{q}_1\cdot\mathbf{q}_2|\to 1}\frac{\Phi_3}{\Phi_5'}\\&=\lim_{|\mathbf{q}_1\cdot\mathbf{q}_2|\to 1}\frac{\arccos(|\mathbf{q}_1\cdot\mathbf{q}_2|)}{2\sqrt{2(1-|\mathbf{q}_1\cdot\mathbf{q}_2|^2)}}=\frac{1}{2\sqrt{2}}.\end{aligned} \tag{30}$$

Thus, $\Phi_2$, $\Phi_3$, $\Phi_5$, and $\Phi_6$ are boundedly equivalent metrics, while $\Phi_4$ is not boundedly equivalent to any other metrics.

## 4.2 Functional Equivalence

Figure 1 shows the relationships among the five distance functions. The figure was generated using many randomly simulated 3D rotations. The linear relationship between $\Phi_3$ and $\Phi_6$ is evident from the straight green line in the figure. It is also clear that each of the mappings from $\Phi_i$ to $\Phi_j$, for $i, j = 2, \dots, 6, i \neq j$, is a bijection and monotonic increasing. That is, these five metrics are functionally equivalent to each other. Again, taking $\Phi_3$ as the reference metric, one can easily derive that the positive, strictly increasing functions $h$ mapping from $\Phi_3$ to the other four metrics are:

$$\Phi_2 = h_{32}(\Phi_3) \equiv \sqrt{2\,(1-\cos\Phi_3)}, \tag{31}$$

$$\Phi_4 = h_{34}(\Phi_3) \equiv 1-\cos\Phi_3, \tag{32}$$

$$\Phi_5 = h_{35}(\Phi_3) \equiv 2\sqrt{2}\sin\Phi_3, \tag{33}$$

$$\Phi_6 = h_{36}(\Phi_3) \equiv 2\Phi_3. \tag{34}$$

It is interesting to note that the curve for $\Phi_4$ versus $\Phi_6$ approaches the origin approximately quadratically. This indicates that $\Phi_4(\mathbf{q}', \mathbf{q}) \to 0$ at a much slower rate than the other metrics when the rotation represented by $\mathbf{q}'$ approaches the rotation represented by $\mathbf{q}$. This confirms that $\Phi_4$ is not boundedly equivalent to other metrics, as demonstrated earlier.

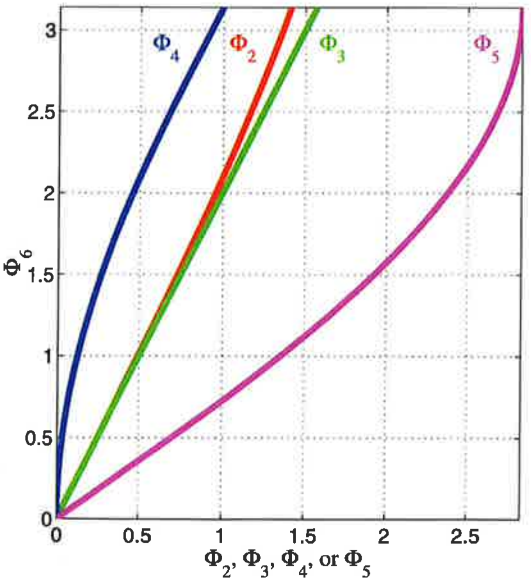


**Fig. 1** Metrics $\Phi_i$, for $i = 2, \dots, 5$ versus $\Phi_6$

If we have a sequence of matrices $\mathbf{R}_n$ such that $\mathbf{R}_n \to \mathbf{R}$ as $n \to \infty$ (Sect. 3.1, (10)), then $\mathbf{R}_n^\top \to \mathbf{R}^\top$ and $\mathbf{R}\mathbf{R}_n^\top \to \mathbf{R}\mathbf{R}^\top = \mathbf{I}$. Using the metric $\Phi_5$ we have

$$\begin{aligned}\Phi_5(\mathbf{R}, \mathbf{R}_n) &= \Phi_5(\mathbf{R}_n, \mathbf{R})\\&= \|\mathbf{I}-\mathbf{R}_n\mathbf{R}^\top\|_F\\&\to \|\mathbf{I}-\mathbf{R}\mathbf{R}^\top\|_F\\&= 0.\end{aligned} \tag{35}$$

Conversely, if $\Phi_5(\mathbf{R}_n, \mathbf{R}) = \Phi_5(\mathbf{R}, \mathbf{R}_n) \to 0$ as $n \to \infty$, then it is necessary that $\mathbf{R}\mathbf{R}_n^\top \to \mathbf{I}$, i.e., $\mathbf{R}_n \to \mathbf{R}$. We can therefore conclude that $\Phi_5$ respects the topology of $SO(3)$, and so do all the metrics $\Phi_2$, $\Phi_3$, $\Phi_4$, and $\Phi_6$, for their functional equivalence to $\Phi_5$.

Table 1 summarizes the computational complexity of these distance functions. Among them, $\Phi_5$ is the most computationally expensive metric. However, the computation work can be significantly reduced if unit quaternions are used instead. Similarly, computation work for function $\Phi_6$

**Table 1** Summary of the amount of computations required for each of the distance functions $\Phi_i$, $i = 2, \dots, 6$. The dot product of two quaternions requires 4 multiplications; the product of two $3 \times 3$ rotation matrices require 27 multiplications; the Euclidean norm of a 4-vector requires 4 multiplications; the $\|.\|_F$ norm requires 9 multiplications; to obtain the absolute or minimum value, 1 comparison is required

| Function | Range | Computations required |
|---|---|---|
| $\Phi_2$ | $[0, \sqrt{2}]$ | 8 multiplications<br>1 comparison |
| $\Phi_3$ | $[0, \pi/2]$ | 4 multiplications<br>1 arccos<br>1 comparison |
| $\Phi_4$ | $[0, 1]$ | 4 multiplications<br>1 comparison |
| $\Phi_5$ | $[0, 2\sqrt{2}]$ | 36 multiplications<br>1 square root<br>Using quaternions (27):<br>7 multiplications<br>1 square root |
| $\Phi_6$ | $[0, \pi]$ | 30 multiplications<br>1 square root<br>1 arccos<br>Using quaternion (same as $\Phi_3$):<br>4 multiplications<br>1 arccos<br>1 comparison |

can be reduced if rotations are represented as unit quaternions so that the $\Phi_3$ metric can be employed.

## 5 Further Analysis of $\Phi_6$ and Iso-error Contours

The $\Phi_6$ metric is geometrically more meaningful than the other metrics, as demonstrated from further analysis of the metric given in this section.

**Theorem 1** *Let* $\mathbf{R}_1$ *and* $\mathbf{R}_2$ *be two 3D rotations. Let* $\mathbf{p}$ *be any arbitrary point on the surface of the unit sphere. Then*

(i) $\Phi_6(\mathbf{R}_1, \mathbf{R}_2) = \max_{\mathbf{p}} \arccos(\mathbf{R}_1\mathbf{p} \cdot \mathbf{R}_2\mathbf{p})$.
(ii) $\hat{\mathbf{p}} = \operatorname{argmax}_{\mathbf{p}} \arccos(\mathbf{R}_1\mathbf{p} \cdot \mathbf{R}_2\mathbf{p}) \Leftrightarrow \hat{\mathbf{p}} \perp$ *rotation axis of* $\mathbf{R}_1^\top\mathbf{R}_2$.

*Proof* The proofs for (i) and (ii) can be combined. Consider $\mathbf{R}_1\mathbf{p} \cdot \mathbf{R}_2\mathbf{p}$. We have $\mathbf{R}_1\mathbf{p} \cdot \mathbf{R}_2\mathbf{p} = \mathbf{p}^\top(\mathbf{R}_1^\top\mathbf{R}_2)\mathbf{p}$. Let $\mathbf{R}_1^\top\mathbf{R}_2 = \exp([\mathbf{w}\beta]_\times)$ for some rotation angle $\beta$ and rotation axis $\mathbf{w}$. By treating $\mathbf{p}$ as a vector of the unit sphere and expressing $\mathbf{p}$ as the sum of two component vectors, one parallel and another orthogonal to $\mathbf{w}$, we see that clearly, if $\mathbf{p}$ is parallel to the rotation axis $\mathbf{w}$, it is invariant under the rotation, so $\mathbf{p}^\top \exp([\mathbf{w}\beta]_\times)\mathbf{p} = \mathbf{p}^\top\mathbf{p} = 1$ and $\arccos(\mathbf{R}_1\mathbf{p} \cdot \mathbf{R}_2\mathbf{p})$ attains the minimum value 0. If $\mathbf{p}$ is orthogonal to $\mathbf{w}$, then $\mathbf{p}^\top \exp([\mathbf{w}\beta]_\times)\mathbf{p} = \cos\beta$ and $\arccos(\mathbf{R}_1\mathbf{p} \cdot \mathbf{R}_2\mathbf{p})$ attains its maximum value $\beta$. Following the same argument, we can conclude that if $\arccos(\mathbf{R}_1\hat{\mathbf{p}} \cdot \mathbf{R}_2\hat{\mathbf{p}}) = \beta$, for any $\hat{\mathbf{p}}$, then it is necessary that $\hat{\mathbf{p}}$ is orthogonal to the rotation axis of $\mathbf{R}_1^\top\mathbf{R}_2$.

What remains to be proven is $\Phi_6(\mathbf{R}_1, \mathbf{R}_2) = \beta$. This can be easily seen by considering the dot product of any two unit quaternions $\mathbf{q}_1$ and $\mathbf{q}_2$ associated with $\mathbf{R}_1$ and $\mathbf{R}_2$. Since $|\bar{\mathbf{q}}_1 \cdot \mathbf{q}_2|$ and $|\mathbf{q}_1 \cdot \bar{\mathbf{q}}_2|$ are, respectively, the cosine of half of the rotation angle $\beta$ of $\mathbf{R}_1^\top\mathbf{R}_2$ and the cosine of half of the rotation angle of $\mathbf{R}_1\mathbf{R}_2^\top$, the equality of $|\bar{\mathbf{q}}_1 \cdot \mathbf{q}_2|$ and $|\bar{\mathbf{q}}_1 \cdot \mathbf{q}_2|$ means that $\Phi_6(\mathbf{R}_1, \mathbf{R}_2) = \beta = \max_{\mathbf{p}} \arccos(\mathbf{R}_1\mathbf{p} \cdot \mathbf{R}_2\mathbf{p})$. □

The above theorem shows that $\Phi_6$ gives the maximum angular measure of separation of points transformed by the two rotations. The following corollary is immediate.

**Corollary 1** *Given any two 3D rotations* $\mathbf{R}_1$ *and* $\mathbf{R}_2$, *all the rotations* $\mathbf{R}_1\mathbf{R}_2^\top$, $\mathbf{R}_1^\top\mathbf{R}_2$, $\mathbf{R}_2\mathbf{R}_1^\top$, *and* $\mathbf{R}_2^\top\mathbf{R}_1$ *have the same rotation angle.*

**Theorem 2** $\Phi_6$ *is a bi-invariant metric on* $SO(3)$.

*Proof* It is easy to verify that $\Phi_6$ is right-invariant as for any rotation matrices $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{R}$, we have

$$\begin{aligned}\Phi_6(\mathbf{R}_1\mathbf{R}, \mathbf{R}_2\mathbf{R}) &= \|\log(\mathbf{R}_1\mathbf{R}(\mathbf{R}_2\mathbf{R})^\top)\| \\ &= \|\log(\mathbf{R}_1\mathbf{R}\mathbf{R}^\top\mathbf{R}_2^\top)\| \\ &= \|\log(\mathbf{R}_1\mathbf{R}_2^\top)\| = \Phi_6(\mathbf{R}_1, \mathbf{R}_2).\end{aligned}$$

To show that $\Phi_6$ is left-invariant also, we may apply Corollary 1 above. Given the same three arbitrary rotation matrices, we have

$$\begin{aligned}\Phi_6(\mathbf{R}\mathbf{R}_1, \mathbf{R}\mathbf{R}_2) &= \|\log(\mathbf{R}\mathbf{R}_1(\mathbf{R}\mathbf{R}_2)^\top)\| \\ &= \|\log((\mathbf{R}\mathbf{R}_2)^\top\mathbf{R}\mathbf{R}_1)\| \\ &= \|\log(\mathbf{R}_2^\top\mathbf{R}^\top\mathbf{R}\mathbf{R}_1)\| \\ &= \|\log(\mathbf{R}_2^\top\mathbf{R}_1)\| \\ &= \|\log(\mathbf{R}_1\mathbf{R}_2^\top)\| = \Phi_6(\mathbf{R}_1, \mathbf{R}_2).\end{aligned}$$

$\Phi_6$ is thus a bi-invariant metric on $SO(3)$. □

The functional equivalence of the metrics $\Phi_i$, for $i = 2, \ldots, 6$, means that all of these five metrics are bi-invariant.

**Theorem 3** *Let* $\mathbf{R}_0 = \exp([\theta_0\mathbf{u}]_\times)$, $\mathbf{R}_1 = \exp([\theta_1\mathbf{v}]_\times)$, *and* $\mathbf{R}_2 = \exp([\theta_1\mathbf{w}]_\times)$. *Then* $\Phi_6(\mathbf{R}_0, \mathbf{R}_1) = \Phi_6(\mathbf{R}_0, \mathbf{R}_2)$ *iff* $\mathbf{u} \cdot \mathbf{v} = \mathbf{u} \cdot \mathbf{w}$.

*Proof* The three rotations can be written in unit quaternions as follows: $\mathbf{q}_0 = (\cos\frac{\theta_0}{2}, \mathbf{u}^\top \sin\frac{\theta_0}{2})^\top$, $\mathbf{q}_1 = (\cos\frac{\theta_1}{2}, \mathbf{v}^\top \sin\frac{\theta_1}{2})^\top$, $\mathbf{q}_2 = (\cos\frac{\theta_1}{2}, \mathbf{w}^\top \sin\frac{\theta_1}{2})^\top$. We have

$$\begin{aligned}&\Phi_6(\mathbf{R}_0, \mathbf{R}_1) = \Phi_6(\mathbf{R}_0, \mathbf{R}_2) \\ &\Leftrightarrow \quad \mathbf{q}_1\bar{\mathbf{q}}_0 \text{ and } \mathbf{q}_2\bar{\mathbf{q}}_0 \text{ have the same rotation angle} \\ &\Leftrightarrow \quad \mathbf{q}_1 \cdot \bar{\mathbf{q}}_0 = \mathbf{q}_2 \cdot \bar{\mathbf{q}}_0 \\ &\Leftrightarrow \quad \cos\frac{\theta_1}{2}\cos\frac{\theta_0}{2} + \sin\frac{\theta_1}{2}\sin\frac{\theta_0}{2}(\mathbf{u}\cdot\mathbf{v}) \\ &\qquad = \cos\frac{\theta_1}{2}\cos\frac{\theta_0}{2} + \sin\frac{\theta_1}{2}\sin\frac{\theta_0}{2}(\mathbf{u}\cdot\mathbf{w}) \\ &\Leftrightarrow \quad \mathbf{u}\cdot\mathbf{v} = \mathbf{u}\cdot\mathbf{w}.\end{aligned}$$

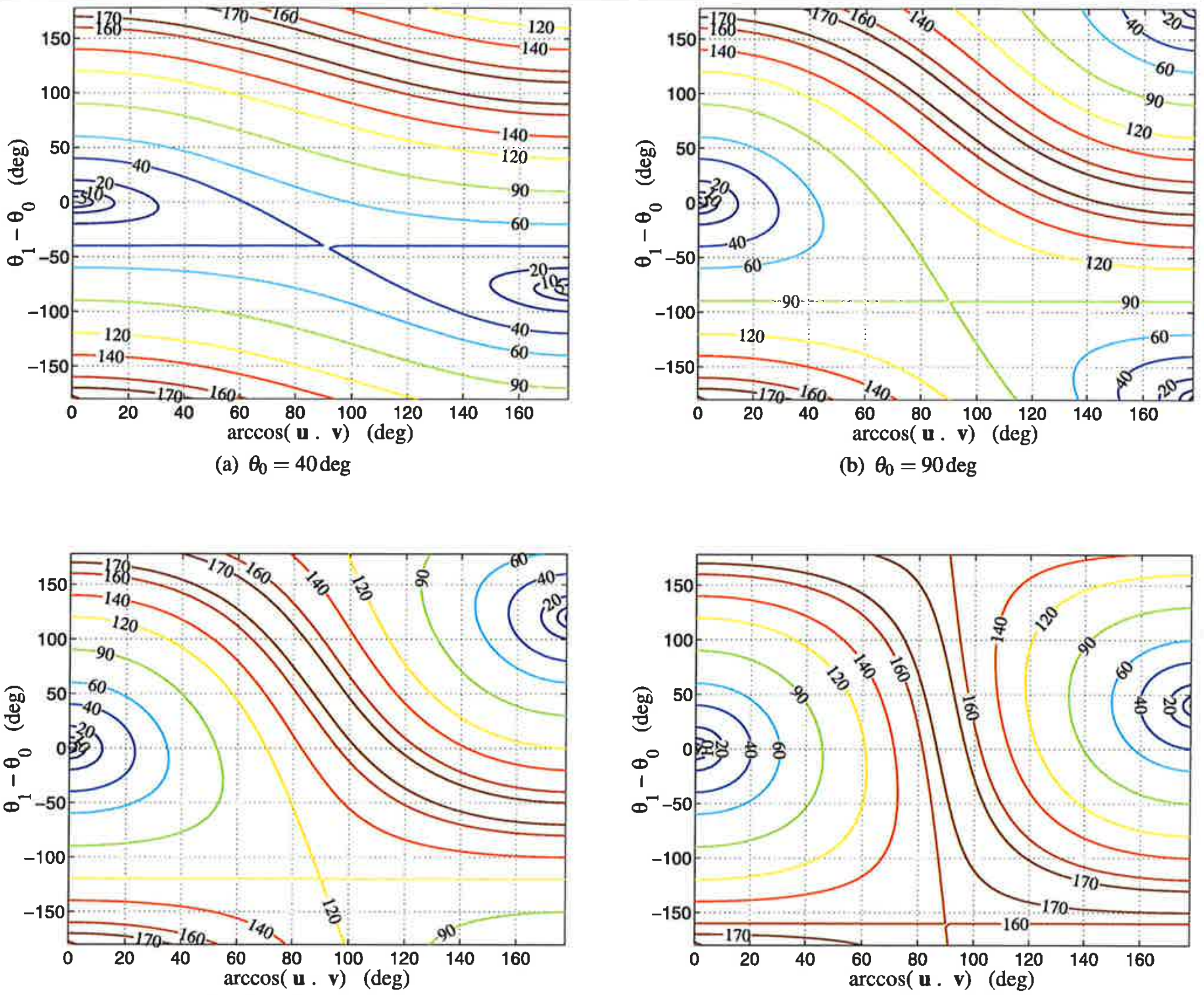
□

**Corollary 2** *The locus of all the rotation matrices,* $\mathbf{R}$, *which have the same rotation angle and are equidistant from a reference rotation* $\mathbf{R}_0 = \exp([\theta_0\mathbf{w}]_\times)$, *is a cone with central axis* $\mathbf{w}$.

In plain English, Theorem 3 and Corollary 2 state that if two different rotations have the same rotation angle and are equidistant from a reference rotation, then their rotation axes must deviate by the same amount from the rotation axis of the reference rotation.

### 5.1 Iso-error Contours

Consider the rotations $\mathbf{R}_0 = \exp([\theta_0\mathbf{u}]_\times)$ and $\mathbf{R}_1 = \exp([\theta_1\mathbf{v}]_\times)$ given in Theorem 3 again. If $\mathbf{R}_0$ is the true rotation and $\mathbf{R}_1$ is an estimated rotation, then the distance measure $\Phi_6(\mathbf{R}_0, \mathbf{R}_1)$ can be taken as an error measure of $\mathbf{R}_1$.



(a) $\theta_0 = 40$deg

(b) $\theta_0 = 90$deg

(c) $\theta_0 = 120$deg

(d) $\theta_0 = 160$deg

**Fig. 2** Error-contour plots of a 3D rotation $\mathbf{R}_1 = \exp([\theta_1 \mathbf{v}]_\times)$ for different values of the rotation angle $\theta_0$ of the true rotation, $\mathbf{R}_0 = \exp([\theta_0 \mathbf{u}]_\times)$. In each plot, the *horizontal axis* corresponds to arccos$(\mathbf{u} \cdot \mathbf{v})$, i.e., the angle of deviation between the two rotation axes of $\mathbf{R}_0$ and $\mathbf{R}_1$; the *vertical axis* corresponds to the difference of the rotation angles, i.e., $\theta_1 - \theta_0$. Both axes are in degrees

Of interest is then the geometry of the set of rotations that are on the iso-error contours. From Theorem 3 and Corollary 2, we see that $\mathbf{R}_1$ is on the iso-error contour with all the matrices in the set $\{\mathbf{R} = \exp([\theta_1 \mathbf{w}]_\times) \mid \mathbf{u} \cdot \mathbf{v} = \mathbf{u} \cdot \mathbf{w}\}$. However, there are other rotation matrices outside this set having the same error measure from $\mathbf{R}_0$ also. If we start altering the direction of the rotation axis $\mathbf{v}$ to increase its angle of deviation with $\mathbf{u}$ (this corresponds to decreasing $\mathbf{u} \cdot \mathbf{v}$), then it would be necessary that the difference between $\theta_0$ and $\theta_1$ be adjusted in order to have $\mathbf{R}_1$ still remain on the same error contour. The induced change to the difference between $\theta_0$ and $\theta_1$ by the angle of deviation is non-linear and can not be expressed in an explicit form. Furthermore, if the angle of deviation is too large, then it is possible that $\mathbf{R}_1$ would move to a different error contour.

The various error contour plots shown in Fig. 2 illustrate the relationship between the angle of deviation, arccos$(\mathbf{u} \cdot \mathbf{v})$, and the difference between $\theta_0$ and $\theta_1$ for the rotations $\mathbf{R}_0$ and $\mathbf{R}_1$ described above. This relationship varies depending on the rotation angle $\theta_0$ of the true rotation. For illustration purpose, the angles in Fig. 2 are all in degrees. We only need to consider cases where $\theta_0 \in [0, 180)$ deg, as those cases outside this range correspond to flipping the rotation axis to the opposite direction. In Fig. 2, if $\theta_1 - \theta_0$ results in $\theta_1$ outside the range $[0, 180)$ deg, similar interpretation can be applied. In all of the plots in Fig. 2, we can see some contours are perfectly horizontal. They correspond to the case where $\mathbf{R}_1 = \mathbf{I}$.

For instance, in Fig. 2(a) where $\theta_0 = 40$ deg, the straight horizontal contour corresponds to $\theta_1 - \theta_0 = -40$ deg, i.e., $\theta_1 = 0$ deg, so the error measure is a constant 40 deg regardless of the direction of the rotation axis $\mathbf{v}$ of $\mathbf{R}_1$. In Fig. 2, it is evident that for a given rotation angle $\theta_0$, the error contours that are less than $\theta_0$ are closed loops while those that are larger than $\theta_0$ are open. The vertical line (not shown in the figure) where $\arccos(\mathbf{u} \cdot \mathbf{v}) = 90$ deg and the horizontal straight contour form a pair of lines of reflection for all the contour curves.

Each plot in Fig. 2 clearly has two valleys where the errors are small. The iso-error contour plots can therefore be used as a visualization aid to help identify the shortest path for updating the estimated $\theta_1$ angle and $\mathbf{u}$ vector for error minimization. These plots were analyzed based on the assumption that no constraints are imposed on the rotations. In many computer vision and robotics problems, such as motion capture and articulated kinematic robot arms, there are often limits associated with the rotations on each joint or the 3D orientations of the recovered poses. For instance, human subjects cannot bend their elbows forward by more than 75 deg or backward by more than 10 deg. In other words, there are infeasible regions in the space of the elbow and other joint angles of the human subjects. As a result, the iso-error contours for the recovered rotations at these joints would be disjoint curve segments.

A motivation behind the study of iso-error contours above is to find ways to reduce the search space for the estimated rotation and to make applications, such as markerless motion capture, more efficient. We note that one of the approaches commonly adopted in markerless motion capture is particle filter based (see, e.g., [2]). The particle filter is known to be an expensive visual tracking process where a large number of particles must be used in order for the tracking to be successful. The high dimensionality of the joint configuration further escalates the number of particles required. If it is known that the estimated rotation of a particular joint is more likely to have an error of, say, $\epsilon$ deg, then, knowledge of the iso-error contours of the 3D rotation of the joint is useful for reducing the sampling region for the rotation and consequently fewer particles would be necessary.

## 6 Conclusion

3D rotations are common entities that arise in many computer vision, graphics, and robotics problems. We have reported six functions for estimating the *distance* between two given rotations represented in various forms. Based on the well-founded theory behind 3D rotations, our detailed comparison on these functions shows that five of these functions are bi-invariant metrics, all of which respect the topology on $SO(3)$; however, only four of them are boundedly equivalent to each other. From our brief analysis on the computational complexity, we conclude that it is both spatially and computationally more efficient to use the unit quaternions for 3D rotations. We have also illustrated the geometry associated with iso-error contours and discuss its possible applications in some common computer vision problems.

**Acknowledgements** The author would like to thank Bill Moran for many valuable discussions on the theory of Lie groups and the anonymous reviewers for their suggestions for improving the manuscript.

## Appendix A: Extracting u and $\theta$ from R

If $\mathbf{R}$ has rotation axis $\mathbf{u}$ and rotation angle $\theta$, then $\mathbf{R}^\top$ has rotation axis $\mathbf{u}$ and rotation angle $-\theta$. From Rodrigues formula, we have

$$\mathbf{R} = \cos\theta\, \mathbf{I} + \sin\theta\, [\mathbf{u}]_\times + (1 - \cos\theta)\mathbf{u}\mathbf{u}^\top, \tag{36}$$

$$\mathbf{R}^\top = \cos\theta\, \mathbf{I} - \sin\theta\, [\mathbf{u}]_\times + (1 - \cos\theta)\mathbf{u}\mathbf{u}^\top. \tag{37}$$

Subtracting these two equations gives $\mathbf{R} - \mathbf{R}^\top = 2\sin\theta[\mathbf{u}]_\times$. This is a skew-symmetric matrix, from which the rotation axis can be easily extracted and normalized to unit magnitude. From the normalization, we implicitly assume that $\sin\theta \geq 0$, thus restricting $0 \leq \theta \leq \pi$. If $\mathbf{R} - \mathbf{R}^\top = \mathbf{0}$, then (i) $\mathbf{R}$ must be the identity matrix, in which case, $\theta$ can be set to 0 and $\mathbf{u}$ can be any arbitrary vector; (ii) $\theta$ must be equal to $\pi$, in which case, $\mathbf{R} + \mathbf{I} = 2\mathbf{u}\mathbf{u}^\top$ is a rank-1 matrix and $\mathbf{u}$ can be obtained by normalizing any column of $\mathbf{R} + \mathbf{I}$.

The three eigenvalues of $\mathbf{R}$ are 1 and $\exp(\pm i\theta)$. Thus, $\mathrm{tr}(\mathbf{R}) = 1 + 2\cos(\theta)$. It follows that $\theta = \arccos((\mathrm{tr}(\mathbf{R}) - 1)/2)$.

Computing the rotation axis and angle this way will ensure that the rotation angle is always positive and in the range $[0, \pi]$. The sign of the rotation axis is correctly determined from the normalization to suit the sign of the rotation angle.

## Appendix B: Proof that $\Phi_6$ is a Metric

To prove that $\Phi_6$ is a metric, we show that it satisfies the three axioms listed in Sect. 3.1. Since $\Phi_6 = 2\Phi_3$ as shown in (34), it is useful to use $\Phi_3$ in the proof when convenient.

Let $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{R}_3$ be three arbitrary rotation matrices in $SO(3)$ and let their corresponding unit quaternions be denoted by $\mathbf{q}_1$, $\mathbf{q}_2$, and $\mathbf{q}_3$. Suppose that $\mathbf{R}_1\mathbf{R}_2^\top = \exp([\theta\mathbf{u}]_\times)$, where $\theta$ is the rotation angle and $\mathbf{u}$ is the unit vector representing the rotation axis. We have

$$\begin{aligned} &\Phi_6(\mathbf{R}_1, \mathbf{R}_2) = 0 \\ &\quad \Leftrightarrow \quad \|\log(\mathbf{R}_1\mathbf{R}_2^\top)\| = 0 \end{aligned}$$

$$\begin{aligned}
&\Leftrightarrow \quad \| [\theta \mathbf{u}]_\times \| = 0 \\
&\Leftrightarrow \quad \theta = 0 \ldots \Phi_6 \text{ returns values in the range } [0, \pi) \\
&\Leftrightarrow \quad \mathbf{R}_1 \mathbf{R}_2^\top = \mathbf{I} \ldots \text{ obvious from Rodrigues formula (36)} \\
&\Leftrightarrow \quad \mathbf{R}_1 = \mathbf{R}_2.
\end{aligned}$$

$\Phi_6$ thus satisfies Axiom (i). The proof for satisfying Axiom (ii) is even more trivial:

$$\begin{aligned}
\Phi_6(\mathbf{R}_1, \mathbf{R}_2) &= \| \log(\mathbf{R}_1 \mathbf{R}_2^\top) \| \\
&= \| \log(\mathbf{R}_2 \mathbf{R}_1^\top) \| \quad \ldots \text{Corollary 1} \\
&= \Phi_6(\mathbf{R}_2, \mathbf{R}_1).
\end{aligned}$$

To show that the triangle inequality is also satisfied, it is easier to use $\Phi_3$ instead:

$$\begin{aligned}
&\Phi_6(\mathbf{R}_1, \mathbf{R}_3) = 2\Phi_3(\mathbf{q}_1, \mathbf{q}_3) \\
&\quad = 2 \arccos(|\mathbf{q}_1 \cdot \mathbf{q}_3|) \\
&\quad \le 2 \arccos(|\mathbf{q}_1 \cdot \mathbf{q}_2|) + 2 \arccos(|\mathbf{q}_2 \cdot \mathbf{q}_3|) \\
&\qquad \ldots \text{holds for angles among unit vectors} \\
&\quad = 2\Phi_3(\mathbf{q}_1, \mathbf{q}_2) + 2\Phi_3(\mathbf{q}_2, \mathbf{q}_3) \\
&\quad = \Phi_6(\mathbf{R}_1, \mathbf{R}_2) + \Phi_6(\mathbf{R}_2, \mathbf{R}_3).
\end{aligned}$$

## References

1. Craig, J.J.: Introduction to Robotics: Mechanics and Control. Addison Wesley, Reading (1986)
2. Deutscher, J., Davison, A.J., Reid, I.: Automatic partitioning of high dimensional search spaces associated with articulated body motion capture. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 669–676 (2001)
3. Faugeras, O.D., Toscani, G.: The calibration problem for stereo. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 15–20. Miami Beach, Florida, USA (1986)
4. Heeger, D.J., Jepson, A.: Simple method for computing 3D motion and depth. In: Proc. International Conference on Computer Vision, pp. 96–100 (1990)
5. Horn, B.K.P.: Closed form solution of absolute orientation using unit quaternions. J. Opt. Soc. Am. **4**(4), 629–642 (1987)
6. Huynh, D.Q., Heyden, A.: Scene point constraints in camera autocalibration: an implementational perspective. Image Vis. Comput. **23**(8), 747–760 (2005)
7. Kelly, J.L.: General Topology. Van Nostrand, New York (1955)
8. Knight, J., Reid, I.: Automated alignment of robotic pan-tilt camera units using vision. Int. J. Comput. Vis. **68**(3), 219–237 (2006)
9. Kuffner, J.J.: Effective sampling and distance metrics for 3D rigid body path planning. In: Proc. International Conference on Robotics and Automation (2004)
10. Larochelle, P.M., Murray, A.P., Angeles, J., A distance metric for finite sets of rigid-body displacement in the polar decomposition. ASME J. Mech. Des. **129**, 883–886 (2007)
11. McCarthy, J.M.: An Introduction to Theoretical Kinematics. MIT Press, Cambridge (1990)
12. Munkres, J.R.: Topology: A First Course. Prentice-Hall, New York (1975)
13. Park, F.C.: Distance metrics on the rigid-body motions with applications to mechanism design. ASME J. Mech. Des. **117**, 48–54 (1995)
14. Park, F.C., Ravani, B.: Smooth invariant interpolation of rotations. ACM Trans. Graph. **16**(3), 277–295 (1997)
15. Pervin, E., Webb, J.: Quaternions in computer vision and robotics. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 382–383. Los Alamitos, CA (1983)
16. Ravani, B., Roth, B.: Motion synthesis using kinematic mappings. ASME J. Mech. Transm. Autom. Des. **105**, 460–467 (1983)
17. Shoemake, K.: Animating rotation with quaternion curves. SIGGRAPH **19**(3), 245–254 (1985)
18. Watt, A., Watt, M.: Advanced Animation and Rendering Techniques. Addison Wesley, Reading (1992)
19. Wolfram MathWorld. http://mathworld.wolfram.com/EulerAngles.html
20. Wunsch, P., Winkler, S., Hirzinger, G.: Real-time pose estimation of 3D objects from camera images using neural networks. In: Proc. International Conference on Robotics and Automation, vol. 4, pp. 3232–3237 (1997)
21. Zisserman, A., Beardsley, P.A., Reid, I.: Metric calibration of a stereo rig. In: Proc. IEEE Workshop on Representations of Visual Scenes, pp. 93–100. Boston, USA (1995)

**Du Q. Huynh** is an Associate Professor at the School of Computer Science and Software Engineering, The University of Western Australia. She obtained her PhD in Computer Vision in 1994 at the same university. Since then, she has worked for the Australian Cooperative Research Centre for Sensor Signal and Information Processing (CSSIP) and Murdoch University. She has been a visiting scholar at Chinese University of Hong Kong, Malmö University, Gunma University, and The University of Melbourne. Associate Professor Huynh is currently a recipient of a Discovery Project grant and two Linkage Project grants funded by the Australian Research Council. Her research interests include shape from motion, multiple view geometry, video image processing, visual tracking, and signal processing.