Planning of Robotic Inspection from Visual Tracking of Manual Surface Finishing Tool

1st Eirik B. Njaastad
Department of Mechanical and Industrial Engineering
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
eirik.njaastad@ ntnu.no

2nd Olav Egeland
Department of Mechanical and Industrial Engineering
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
olav.egeland@ ntnu.no

***Abstract*—A planning system for efficient of robotic inspection of surface tolerances is presented. This is done using visual tracking of a manual polishing tool. The application that is studied is surface finishing of large ship propeller blades. The propeller blade is cast in in NiAl bronze and then adjusted by manual surface finishing. Robotic inspection is used to check the quality of the resulting surface using an high-accuracy RGBD-camera. To avoid time-consuming inspection of the entire propeller blade, the area affected by the manual adjustment is detected with a visual tracking system, which measures the motion of the manual tool using a particle filter and a CAD model of the tool. The main contribution of this work is the strategy for selecting camera views for the inspection robot. The algorithm employs Hotelling's T-squared distribution in a Principal Component Analysis to find efficient viewpoints. The approach is implemented with an industrial robot, a high-accuracy RGBD-camera, and a low-cost RGBD-camera. The system is validated in simulations and experiments, where a surface conditioned ship propeller blade is inspected.**

***Index Terms*—Manufacturing automation, Inspection, Viewpoints, Statistical analysis, Particle filters**

I. Introduction

Optical inspection of surface tolerances for industrial products can be performed by robotic inspection. Then a robot is used to move a camera over the surface of the workpiece. The robot program must be designed so that the relevant parts of the surface is scanned, with the optical sensor positioned sufficiently close to the surface during scanning, so that the required accuracy is achieved. For small batch production with frequent change of product dimensions and product types, it is important that the robotic inspection tasks can be generated efficiently, and preferably by using automated techniques for generating the inspection program. If a CAD model of the workpiece is available, the inspection points can be generated according to surface definition. In some applications, the product may be large and a complete inspection of the geometry may be time consuming. Moreover, it may be necessary to inspect only parts of the workpiece, like weld seams or parts that have undergone manual surface finishing. In this case, it may be advantageous to track the tool paths of the manual operation and then use this as input for the robotic inspection.

The authors wish to thank the OLIVER project and industrial partner Oshaug Metall AS. The project is supported by the Norwegian Research Council under the BIA program, project number 261639.

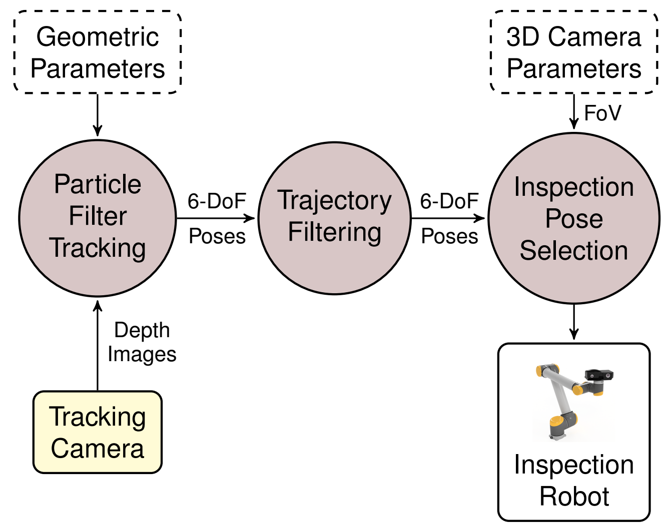


Fig. 1. Overview of the information flow in the proposed approach. Inputs to the tracking process are depth images from an RGBD-camera and a CAD model of the tool to be tracked. In order to generate the camera poses for inspection, information about depth range and field of view of the scanning RGBD-camera is necessary.

Many contact and non-contact methods have successfully been applied in order to track the tool movements of Skilled Workers. It is possible to use contacting motion capturing systems like exoskeletal systems. However, the contacting methods may hinder the human motion or the production process. Noncontacting ways such as vision-based techniques may not hinder human movements [1]. Fiducial markers have been applied successfully in vision-based human-robot interfaces for many applications, including teleoperation [2], where the markers placed on the hand of a human were used to control a robot remotely. It is beneficial to avoid interfering with the process when tracking it, e.g. by avoiding fiducial markers attached to the tool while the worker is performing the process [3].

Optical 3D digitizing systems allows for inspecting complex shapes in a short time. One of the main related problems is to determine the sensor position in order to achieve the best measurement accuracy using a minimal number of views. The challenge of automatic determination of view poses has been widely studied in robotics and computer vision. CAD-based approaches originate from Coordinate Measuring Machines (CMM), but has been extended to 6 DoF robotic arms coupled with 3D optical scanners. A comprehensive review of proposed and employed methodologies and systems can be found in [4], [5], [6]. In computer vision literature, the problem of determining the best camera view poses when no CAD data is available is known as Next Best View (NBV) planning [7]. NBV is used to minimize the number of required camera views to acquire a complete 3D model, or to ensure that the view poses selected are as close as possible to the optimal view poses.

Capturing of surface finishing processes and other complex behaviors from human demonstration is a research topic of consistent relevance [8], [9], [10], [11], [12]. Automatic inspection of a processed surface based on the learned trajectories is however not much described in the literature.

In this paper, we propose to track the trajectories of the manual tool, and then to use the captured tool trajectories as the basis for automated inspection planning. A demonstration setup is described (Figure 1), where a simple RGBD-camera is used for tracking a tool used in manual surface finishing. A particle filter is used for tracking a CAD model of the tool, making the tracker sufficiently robust to handle occlusions and cluttered environments. Based on the recorded time history of motions from the particle filter, robotic inspection is performed where the robot moves a high-accuracy RGBD-camera over the relevant parts of the workpiece. This is done using statistical properties of the assumed surface. The system is implemented in simulations and experiments, and the performance is investigated.

The paper is organized as follows: In Section II the developed motion tracking and inspection system is presented. A computational analysis of the algorithm performance takes place in Section III. Simulated and experimental verification of the overall scheme is described in Section IV, followed by a summary of the approach and proposals for further research in Section V.

II. Motion Tracking and View Planning

The goal of the approach is to enable the inspection robot to execute a scanning program based on the tool trajectories learned from a Skilled Worker. The resulting trajectories are assumed to follow the surface of the object to inspect. The robot will thus only inspect the parts of the surface which the worker has performed any processing on.

We employ a Rao-Blackwellized particle filter for tracking the tool movements [13]. A demonstration consists of a trajectory T and a set of 6-dimensional references R. The trajectory $T = \{t_0, \ldots, t_{n-1}\}$ represents the $n \in \mathbb{N}$ samples (via-points) of the tool center point t_i as a unit quaternion and translation vector.

A. System Overview

An overview of the proposed approach is shown in Figure 1. We start by capturing the tool trajectories with the particle filter. The input to the particle filter is depth images from a low accuracy RGBD-camera at 30 Hz, combined with a CAD model of the tool to be tracked. After various filtering of the captured tool trajectories, the scanning view poses are selected on the basis of a set of basic camera parameters: Its field of view and optimal scanning distance.

The cameras and robot are calibrated w.r.t each other using an offline calibration procedure.

B. Particle Filter Tracking

The Rao-Blackwellized Particle Filter (RBPF) [14] improves the performance of particle filtering by sampling over a subspace of the probability distribution of the state. The method is based on the assumption that it is possible to evaluate some of the filtering equations analytically and the others with a particle filter instead of computing everything with pure sampling.

C. Trajectory Filtering

Many surface treatment processes consist of repetitive movements over the same surface patch. The tracked trajectory would then be too detailed and ineffective as input for the inspection robot. A learned process trajectory could potentially contain several thousand recorded tool poses. It would be a highly time-consuming task for the inspecting robot to visit and scan all of them. In many cases, the camera is able to cover large parts of the tracked path simultaneously. It is thus necessary to use filtering methods in order to streamline the inspection.

Our main strategy for filtering the view poses is to divide them into a voxel grid structure, effectively grouping nearby view poses and replacing them with their centroid. This ensures that repetitive tool movements are filtered into essential view poses. By adjusting the voxel size, the resolution of the following view selection can be adjusted. Increased grid resolution comes with a computational cost.

Before the voxel grid filtering step, we do a simple Gaussian smoothing of the recorded data, in both the forward and reverse directions.

D. Camera Orientation by Averaging Quaternions

The distribution of the recorded poses into a voxel grid does not comprise determining a camera orientation for the camera viewpoints. For each pose present in the voxel grid, the camera orientation must be chosen.

We use the average of the local group of viewpoints for each of the poses in order to account for more of the local orientational information. We employ a quaternion based fast averaging technique as proposed in [15]. The local group is determined by doing a linear k-nearest neighbor search for each of the poses present in the voxel grid.

Given $\boldsymbol{q}_i$, a set of quaternions, we form the weighted dot product matrix:

$$\boldsymbol{B} = \frac{1}{n_q} \sum_{i=1}^{n_q} w_i^q (\boldsymbol{q}_i^T \cdot \boldsymbol{q}_i) \tag{1}$$

where n_q is the number of poses in the local group, and w_i^q is the associated weights, given the pose $\boldsymbol{q}_i$. The mean quaternion

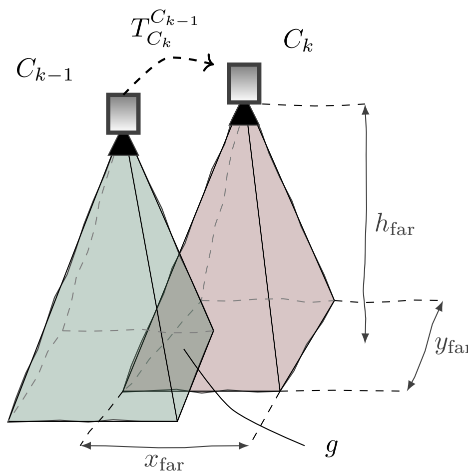


Fig. 2. The volumetric overlap (g) between adjacent camera poses. The transformation between the camera poses C_k and C_{k-1} is $T_{C_k}^{C_{k-1}}$. The camera views are modeled as pyramids with base sides y_{far}, x_{far}, and height h_{far}.

$\boldsymbol{q}_{avg}$ is given by the eigenvector $\boldsymbol{e}_{max}$ corresponding to the maximum eigenvalue of $\boldsymbol{B}$, λ_{max}.

E. Hotelling's T-squared Distribution

We employ a multivariate statistical distribution in order to select the most relevant view poses for inspection. The Hotelling's T^2 values represent a measure of the variation in each sample within the model [16]. It indicates how far each sample is from the center of the model.

The statistics are calculated for a Principal component analysis model (PCA), which is a well-established technique for unsupervised dimensionality reduction [17]. PCA is used for data compression and information extraction.

The T^2 value for the i^{th} observation is defined as:

$$T^2 = \sum_{a=1}^{a=A} \left(\frac{t_{i,a}}{s_a} \right)^2 \tag{2}$$

where the s_a^2 values are constants, and represents the variances of each component. A is the PCA components, with score $t_{i,a}$.

After calculating the T^2, we sort the view poses in descending order and use the resulting list as input for checking volumetric overlap between adjacent camera views.

F. Camera Volumetric Overlap Calculation

In order to reduce the number of camera views such that there is only a minimal camera overlap, we iterate through all view pose candidates generated from the statistical selection step. An illustration of the geometric method of comparison is shown in Figure 2. The camera views are modeled as pyramids with sides and height corresponding to the field of view and optimal scanning distance of the inspecting RGBD-camera.

Camera view overlap (fraction of superimposed voxels) is determined using a Jaccard similarity coefficient between adjacent camera views in voxel space, where each attribute of C_k and C_{k-1} can either be 0 or 1.

$$\begin{aligned} J(C_k, C_{k-1}) &= \frac{|C_k \cap C_{k-1}|}{|C_k \cup C_{k-1}|} \\ &= \frac{|C_k \cap C_{k-1}|}{|C_k| + |C_{k-1}| - |C_k \cap C_{k-1}|} \\ &= \frac{\sum_{i=1}^{n}(C_k[i] \cdot C_{k-1}[i])}{\sum_{i=1}^{n}(C_k[i] + \sum_{i=1}^{n}(C_{k-1}[i] - \sum_{i=1}^{n}(C_k[i] \cdot C_{k-1}[i])} \end{aligned} \tag{3}$$

The similarity coefficient is defined $0 \leq J(C_k, C_{k-1}) \leq 1$.

View pose candidates with a similarity coefficient over a certain threshold are disregarded. Finally, a list of reasonable camera view poses for inspection emerges.

III. Computational Analysis

A simple but scalable surface finishing scenario is used for evaluating the computational capabilities of the proposed algorithm. The test scenario consists of a path where the movements follow an equilateral triangular pattern. It has a total height of $500\,\mathrm{mm}$, a total width of $1200\,\mathrm{mm}$, and it is arranged in a plane as shown in Figure 3. The surface corresponds to an area of $0.6\,\mathrm{m}^2$, filled with a variable number of equilateral triangles. This effectively corresponds to different trajectory lengths and resolutions over an equal area, and thus a varying number of view poses.

The analysis is performed by measuring the time spent on each part of the algorithm when varying the number of input view poses and the resolution of the voxel grid. Simulations were performed on a computer with a $3.6\,\mathrm{GHz}$ CPU running Windows 10 and implementing the view selection algorithm in a single-thread MATLAB program. The tracking part of the proposed system is hence not evaluated in this test.

The simulations were performed by assuming a camera with optimal scanning distance of $700\,\mathrm{mm}$, and a field of view image area ($y_{\mathrm{far}} \times x_{\mathrm{far}}$) of $430\,\mathrm{mm} \times 270\,\mathrm{mm}$.

The resulting time consumption for various situations is shown in Figure 4. The total running time takes a linear behavior around 1000 input trajectory points (Figure 4a), whereas increasing voxel grid resolution (decreasing voxel size) results in exponential behavior when approaching zero (Figure 4b).

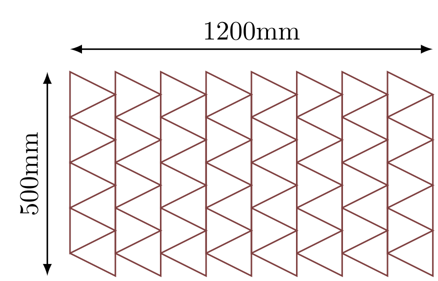
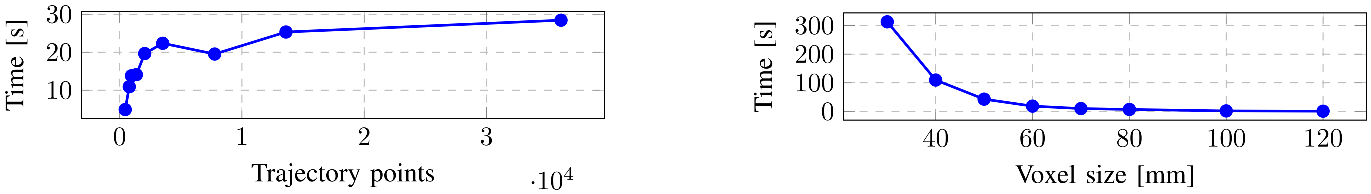


Fig. 3. The triangular pattern used in various resolutions for analyzing computational characteristics of the proposed algorithm.



(a) Execution time as a function of the number of input points.

(b) Execution time as a function of voxel grid filtering resolution.

Fig. 4. Analysis of computational characteristics of the proposed algorithm. The total algorithm running time is considered, where voxel grid filtering, camera orientation selection, and the final statistical and geometrical camera view selection step is performed. In (a), the number of considered points is varied, while the voxel size is varied in (b).

IV. Experimental Results

We tested our approach in an experimental setup using a Universal Robots UR10 robot equipped with a Zivid high-accuracy RGBD-camera for inspection. Another RGBD-camera (Microsoft Kinect v2) was used to track the motions of an angle grinder using the particle filter and a CAD model of the angle grinder. The trajectory was obtained while the Skilled Worker performed surface polishing of a ship propeller blade as shown in Figure 5a.

This experiment was conducted in order to evaluate the qualitative results of the approach. The motions performed by the Skilled Worker in the surface conditioning process is characterized by repetitive, alternating motions over the double-chambered surface. The trajectory shown in Figure 5c is the raw tool trajectory captured with the tracking camera and particle filter. Figure 5d shows the resulting scanning camera views used by the inspection robot to capture the point cloud of the ship propeller blade shown in Figure 5b.

The point cloud captured from the qualitative experiment confirms the quantitative results. The proposed inspection camera views have high coverage of the captured trajectory and underlying ship propeller surface.

V. Summary & Conclusions

This paper presents a new approach to generate meaningful camera views for inspecting the results of manual surface finishing processes. The approach is based on the assumption that the tool trajectories of the Skilled Worker represent the most relevant regions of the underlying surface of the workpiece, namely the parts where a process has been performed.

The approach is also suitable for generating meaningful view poses for an inspection robot in the case where robots or CNC machines perform an industrial process and sparse or no CAD-data is available, or when inspecting workpieces processed by a robot programmed by online teach- or lead-through methods.

In contrast to existing work, the algorithm expects a tool trajectory and is capable of adapting the orientation of the robot. In order to select meaningful camera views, camera parameters such as camera field of view and optimal scanning distance are necessary input to the algorithm. The algorithm is not aware of what process it has been shown. It does not recognize, that for example a surface polishing task has been demonstrated. There is neither deduction of the actions nor a priori knowledge.

We provided qualitative results by simulating various tool paths and quantitatively results by recording data from a human Skilled Worker. The approach can be easily extended to other applications, such as robotic welding, machining, or painting.

While the different operations described in this paper have been functionally implemented and successfully experimented with, the integration into a complete process cycle remains to be done. Collision avoidance could be added to the system, taking into account the complete robot model and the continuous stream of information originating while inspecting the workpiece. Further work may also focus on improving the scanning execution time and integration of the approach into our ship propeller inspection system [18].

References

[1] G. Du and P. Zhang, "A markerless human-robot interface using particle filter and kalman filter for dual robots.," *IEEE Trans. Industrial Electronics*, vol. 62, no. 4, pp. 2257–2264, 2015.

[2] J. Kofman, X. Wu, T. J. Luu, and S. Verma, "Teleoperation of a robot manipulator using a vision-based human-robot interface," *IEEE transactions on industrial electronics*, vol. 52, no. 5, pp. 1206–1219, 2005.

[3] R. Dillmann, T. Asfour, M. Do, R. Jäkel, A. Kasper, P. Azad, A. Ude, S. R. Schmidt-Rohr, and M. Lösch, "Advances in robot programming by demonstration," *KI-Künstliche Intelligenz*, vol. 24, no. 4, pp. 295–303, 2010.

[4] G. Sansoni, M. Trebeschi, and F. Docchio, "State-of-the-art and applications of 3d imaging sensors in industry, cultural heritage, medicine, and criminal investigation," *Sensors*, vol. 9, no. 1, pp. 568–601, 2009.

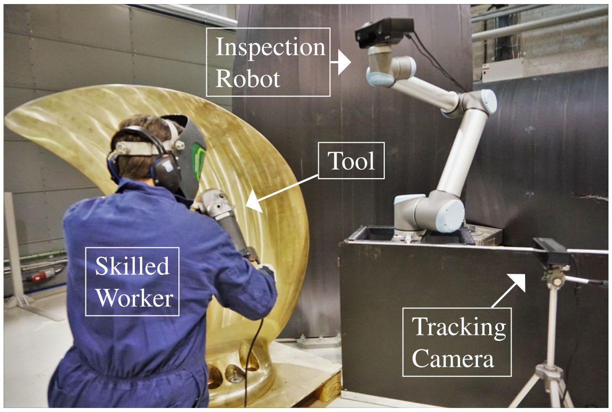
[5] F. Zhao, X. Xu, and S. Q. Xie, "Computer-aided inspection planningthe state of the art," *Computers in industry*, vol. 60, no. 7, pp. 453–466, 2009.

[6] Y. Li and P. Gu, "Free-form surface inspection techniques state of the art review," *Computer-Aided Design*, vol. 36, no. 13, pp. 1395–1417, 2004.

[7] C. Connolly, "The determination of next best views," in *Robotics and automation. Proceedings. 1985 IEEE international conference on*, vol. 2, pp. 432–435, IEEE, 1985.

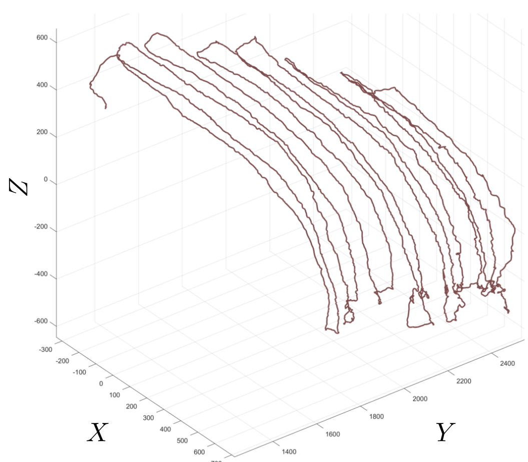
[8] W. Ng, H. Chan, W. K. Teo, and I.-M. Chen, "Programming robotic tool-path and tool-orientations for conformance grinding based on human demonstration," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pp. 1246–1253, IEEE, 2016.

[9] B. Nemec, K. Yasuda, N. Mullennix, N. Likar, and A. Ude, "Learning by demonstration and adaptation of finishing operations using virtual mechanism approach," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7219–7225, IEEE, 2018.

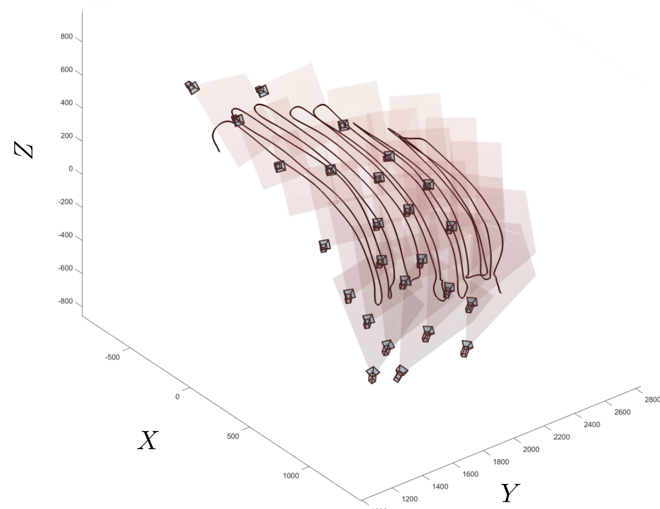


(a) Capturing the tool paths of a Skilled Worker polishing a ship propeller blade.

(b) The resulting point cloud after inspection of the propeller blade surface.



(c) Raw tool trajectories.



(d) Visualization of the automatically generated inspection view poses.

Fig. 5. Evaluation test-case: Capturing a surface finishing process. In (a), a Skilled Worker is polishing a propeller blade surface using an angle grinder which is tracked by an RGBD-camera and particle filter approach. The resulting tracked raw trajectory from the process is shown in (c). In (d), the view poses generated by our approach is shown. The resulting point cloud from the performed inspection is shown in (b).

[10] C. W. Ng, K. H. Chan, W. K. Teo, and I.-M. Chen, "A method for capturing the tacit knowledge in the surface finishing skill by demonstration for programming a robot," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 1374–1379, IEEE, 2014.

[11] E. KALT, R. MONFARED, and M. JACKSON, "Towards an automated polishing system- capturing manual polishing operations," *International Journal of Research in Engineering and Technology*, vol. 5, no. 07, pp. 182–192, 2016.

[12] A. Irish, I. Mantegh, and F. Janabi-Sharifi, "A pbd approach for learning pseudo-periodic robot trajectories over curved surfaces," in *Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on*, pp. 1425–1432, IEEE, 2010.

[13] M. Wüthrich, P. Pastor, M. Kalakrishnan, J. Bohg, and S. Schaal, "Probabilistic object tracking using a range camera," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 3195–3202, IEEE, 2013.

[14] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of Nonlinear Filtering*, vol. 12, pp. 656–704, 2009.

[15] F. L. Markley, Y. Cheng, J. L. Crassidis, and Y. Oshman, "Averaging quaternions," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 4, pp. 1193–1197, 2007.

[16] H. Hotelling, "The generalization of student's ratio," *Ann. Math. Statist.*, vol. 2, pp. 360–378, 08 1931.

[17] K. Pearson, "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[18] E. B. Njaastad, N. H. Munthe-Kaas, and O. Egeland, "Robotic autoscanning of highly skewed ship propeller blades," *IFAC-PapersOnLine*, in press.