Proceedings 2021 ASME International Manufacturing Science and Engineering Conference
MSEC2021
June 21-25, University of Cincinnati, USA

MSEC2021-63892

AUTO-CALIBRATION FOR VISION-BASED 6-D SENSING SYSTEM TO SUPPORT MONITORING AND HEALTH MANAGEMENT FOR INDUSTRIAL ROBOTS

Guixiu Qiao
National Institute of Standards and Technology
Gaithersburg, Maryland, USA

Guangkun Li
Johns Hopkins University Applied Physics
Laboratory, Maryland, USA

KEY WORDS

Advanced Sensing, Vision-based System Calibration, Prognostics and Health Management (PHM), Industrial Robot, Accuracy Degradation

ABSTRACT

Industrial robots play important roles in manufacturing automation for smart manufacturing. Some high-precision applications, for example, robot drilling, robot machining, robot high-precision assembly, and robot inspection, require higher robot accuracy compared with traditional part handling operations. The monitoring and assessment of robot accuracy degradation become critical for these applications. A novel vision-based sensing system for 6-D measurement (six-dimensional x, y, z, yaw, pitch, and roll) is developed at the National Institute of Standards and Technology (NIST) to measure the dynamic high accuracy movement of a robot arm. The measured 6-D information is used for robot accuracy degradation assessment and improvement. This paper presents an automatic calibration method for a vision-based 6-D sensing system. The stereo calibration is separated from the distortion calibration to speed up the on-site adjustment. Optimization algorithms are developed to achieve high calibration accuracy. The vision-based 6-D sensing system is used on a Universal Robots (UR5) to demonstrate the feasibility of using the system to assess the robot's accuracy degradation.

INTRODUCTION

Industrial robot systems are very complex, containing sub-systems and components that interact within manufacturing work cells. It is a challenge to determine their specific influences on performance when an unexpected break down happens in a work cell [1, 2]. There is increasing interest to enable the ability to leverage prognostic data to monitor the system's health status [3, 4]. Data analysis may generate actionable intelligence in maintenance plan optimization. The monitoring of robot health conditions may also help to detect potential faults and failures. It helps to eliminate the unexpected maintenance shutdowns that are expensive.

To monitor the health condition of an industrial robot, low-level data can be extracted from robot controllers. Information from controllers may include joint current, positions, velocities, accelerations, etc. The controller's low-level data can be used to capture and monitor the abnormal changes of signals [5]. But this method is suitable for repeating operations, where signals usually have fixed patterns at normal conditions. Moreover, some non-geometric errors, for example, squareness errors between axes and deflections of the structure cannot be reflected from joint data. An individual joint's data cannot determine the overall robot's health condition, that is, the accuracy degradation of the robot's tool center position (TCP). To monitor a robot's absolute accuracy changes, a sensor that can measure the 6-D information of the robot TCP is needed.

Many sensing systems in the market can measure 6-D information, including laser trackers [6], theodolite measurement devices [7], probing coordinate measuring machines (CMM) [8, 9], and optical tracking systems [10, 11]. Laser trackers and theodolites have the line-of-sight issue for moving objects. CMM measurement is slow, not suitable to measure the robot arm's dynamic movements. Also, laser trackers and CMM measurements are expensive. Optical tracking systems are vision-based instruments. The major challenge to optical tracking lies in measurement accuracy. Since they use infrared (IR) cameras and reflective spheres as targets, ambient light has a strong influence on measurement uncertainties [10]. A novel vision-based 6-D sensing system (patent pending) has been developed at the National Institute of Standards and Technology (NIST). The system contains a smart target (Fig. 1a)) and a vision-based measurement instrument (Fig 1b)). The smart target has three-color light pipes. It is motor-driven and can constantly rotate toward the measurement instrument. There is an orientation sensor mounted on the elevation shaft. An initial pose is defined by the user. When the smart target rotates away from the original pose, motors on the azimuth and elevation shaft rotate the elevation shaft back to its

a) Smart target (mounted on the end effector of UR5)

b) Vision-based measurement instrument

Figure 1. Vision-based 6-D measurement

original pose driven by the orientation sensor's feedback. The novel design enables high accuracy (within 0.1mm) and high speed (at a minimum of 30 Hz) in measurement. The design of the smart target system is detailed in [12, 13]. This paper focuses mainly on the automatic calibration technology to achieve the high accuracy of the vision-based 6-D sensing system.

This paper is organized as follows: the next section describes the automatic distortion calibration development. Then refinement algorithms are discussed that use iteration calculations to handle tilt images and imperfect calibration boards. Next, the verification method for the calibration result is described. The last section presents the establishment of an automatic stereo calibration procedure.

CAMERA AUTOMATIC CALIBRATION DEVELOPMENT

Traditional camera calibration methods compute camera intrinsic parameters from a set of target features with known geometries. The most common target is a checkerboard. Calibration usually requires users to select corners and areas on the checkerboard images manually. The manual process has several problems:

1) Time consuming. Users must manually click four corners on each checkerboard image to define the region of interest. When there are multiple images, this process is very time-consuming.

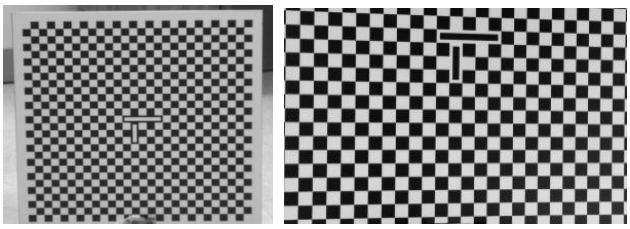
2) Prone to error. The corner detection relies on the accuracy of the user clicking that is prone to error.

3) Difficulty for stereo camera calibration. Using the dual camera as an example, stereo camera calibration needs to find correspondence between the left and right camera. Users either have to make sure both cameras contain the full checkerboard image, or count and align the same corners for both left and right images manually. This process is tedious and time-consuming.

A method for automatic camera calibration was developed. This method doesn't involve user clickings to define the region of interest. Images are processed in batch. The key to the automatic method is to enable auto-counting and auto-alignment. As shown in Fig. 2a), the calibration board used at NIST has two long rectangle makers in the center. The special markers are utilized as the indicator of the checkboard's orientation and the center position. If the marker features are correctly detected, any corner in the image can find the correspondence to the center. Users may customize their markers on their checkerboard, for example, adding extra dots or special shapes as indicators. The automatic procedure for calibration board corner detection is as follows.

1. Obtaining reverse images

The original image of the calibration board has white rectangles as the markers shown in Fig 2a). If the white rectangle



a) Original calibration board

b) Reverse image of calibration board

Figure 2. Calibration board

is used as the indicator, false detections have a large chance to happen because the edge of the calibration board is white. To solve this problem, a reverse calculation is implemented to reverse the white rectangle to black color, as shown in Fig. 2b) Then the contours of the two black rectangles are detected.

2. Obtaining binarized image

Images need to be filtered before further processing based on a threshold to obtain a binarized image. Two methods are developed to identify a proper threshold for the image. The first method is the adaptive thresholding based upon the Otsu Binarization method [14]. This is a well-established threshold method and we used an algorithm from the OpenCV library. This method is less robust and prone to false thresholding but runs faster. An improved method is developed by adding the calculation of image histogram, histogram smoothing, calculation of histogram gradient, and smart gradient divide thresholding. This new method is more robust compared with the first method. Because the algorithm is more complex, the calculation takes more time.

3. Squares and rectangles detection

Given the binary images, an algorithm is needed to detect the contours of black blobs. The method to find the contours can be simplified to finding four lines by using iteration and area approximation. The result is shown in Fig. 3. The detected long

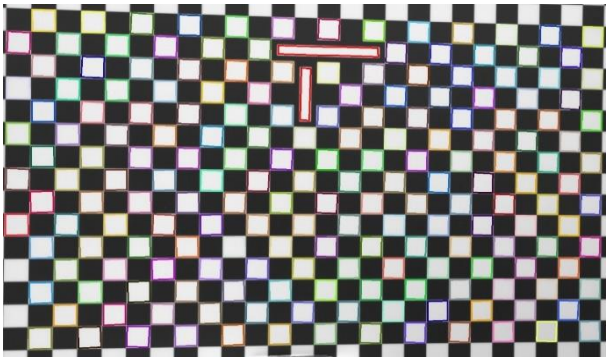


Figure 3. Illustration of the valid contours

rectangles are drawn in red color. Other detected squares are drawn in random colors. To remove noise and false detection, all the contours are arranged in a multi-level hierarchy-tree structure (parents and children) for further cleanup. In the end, the squares on the calibration board and the two long rectangles (drawn in red color in Fig. 3) are identified. The two long rectangles are further analyzed to determine the orientation. Corners are used as the 'anchor' to map the rest of the squares to a grid.

4. Recursive algorithm development for corner mapping

The left top corner of the long rectangle is treated as the anchor. The other square corners' positions are defined relative to the anchor. Then recursive searches are performed to find the nearest squares one-by-one. Grid coordinates are assigned to these square corners. By doing that, the so-called 'position awareness' is achieved. As a result, the corner correspondence from different images can be found automatically. Fig. 4 shows the assigned grid coordinates.

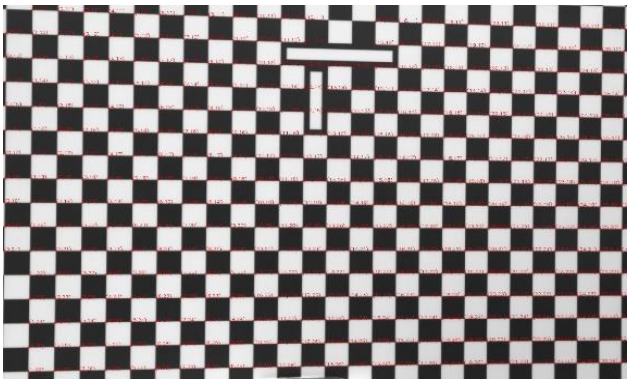


Figure 4. Labeled corners

5. Subpixel edge detection

The detected corners are passed through a subpixel edge detection algorithm. The algorithm runs in iterations to find the location of corners or radial saddle points in sub-pixel accuracy as shown in Fig. 5.

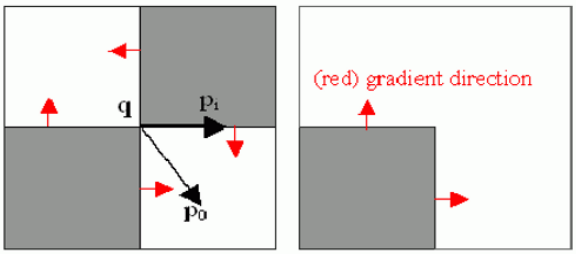
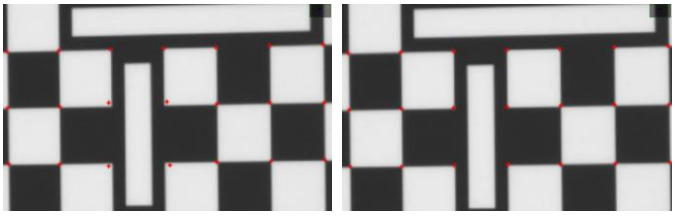


Figure 5. Subpixel edge detection algorithm from OpenCV library

During the development, multiple algorithms are implemented. It is found that the subpixel corner detection algorithms sometimes give the wrong location, as shown in Fig. 6a). The detected corners are marked with red crosses. When the corner is not a 'saddle' point, where near the anchor, the checkerboard conner becomes a white corner surrounded by the black background. The detected corner has deviated from the real corner locations. This problem is addressed by combining the current saddle point detection algorithm with a Harris corner localization approach [15]. Fig. 6b) shows that the results are more accurate using the new algorithm.



a) Corner detection with errors b) Improved corner detection

Figure 6. Corner detection improvement

6. Alignment of the detected corners to calibration board coordinate frame

After the corners are detected precisely, an iterative recursive neighbor search is performed to align the corners to a grid. We define the left top corner of the calibration board to the coordinates (0,0). The right bottom corner coordinate is (26, 26) because the current calibration board has 26 x 26 grid size.

There are some requirements for taking calibration pictures. 1) Both of the rectangle markers must be seen in the image; 2) the calibration board cannot be rotated more than 45 degrees otherwise the algorithm may be confused by the location of the left top corner; 3) the calibration board (at least the two rectangle boards) occupies both cameras' field of view (FOV). Once the above procedures are followed, the calibration can be performed automatically and robustly. There is no need for a user to mouse-click hundreds of times which significantly reduces the problems caused by human errors.

CALIBRATION REFINEMENT APPROACH

Camera calibration accuracy is affected by the corner detection accuracy. Many calibration images are not fronto parallel (it means the calibration board is parallel to the image sensor). The titled calibration board image may cause non-linear distortion when localizing the subpixel corners [16]. A calibration refinement approach is developed to rectify the calibration images and refine the calibration with an iterative approach.

The refinement approach is described as follows.

- Corner detection: Detect the calibration board corners in the input images.
- First calibration: Use the detected corners to estimate camera parameters.
- Do iterations until convergence.
- Perform undistortion: Use the camera parameters to undistort and correct the input images to a frontal canonical pattern.
- Redo corner detection: Relocalize calibration pattern control points in the fronto parallel pattern.
- Re-project: Reproject the control points using the estimated camera parameters.
- Re-calibrate: Use the projected control points to re-fine the camera parameters.

Fig. 7 shows the original calibration images (left) and the fronto corrected images (right). Then the corner detection algorithms are applied to the fronto corrected images. The calibration procedure is performed for two iterations in our test. The calibration results show an improvement of about 5%. When the tilt angles of the calibration images are large, the improvement will be more significant with the benefit of the detection on fronto images.

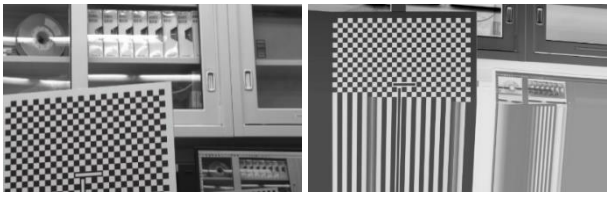


Figure 7. Original images and the fronto corrected images

Another algorithm is also implemented to handle the imperfect calibration board issue using an iterative approach [17]. Since the calibration board used at NIST is a high precision certified target, the existing test did not show significant improvement in calibration results. However, if users are using an imperfect calibration board, the developed algorithms are robust to handle the imperfect target condition.

VERIFICATION OF CALIBRATION RESULTS

After the calibration method is established, the calibration results need to be verified. A test method is developed using the straightness of line features to verify the distortion correction effects. An aluminate plate was machined to achieve surface flatness under 30 um. Three straight lines were precisely printed and applied on the top, middle, and bottom of the plane, as shown in the left picture of Fig. 8. The purpose of this test method is to evaluate the straightness of the lines that are detected by the camera after distortion correction. An unsuccessful calibration will result in curved lines with patterns. On the contrary, a successful calibration will result in a straight line. Thus, this is a good measure to verify the distortion correction of a calibration.

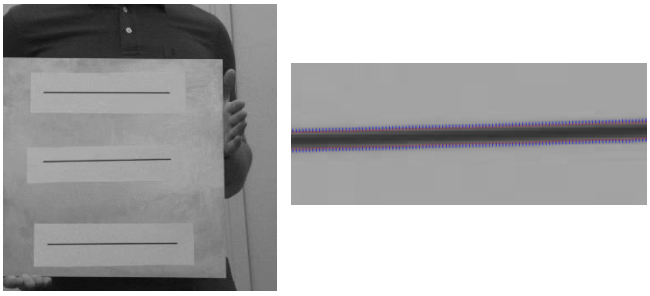


Figure 8. Image with lines used for distortion verification

Multiple line detection algorithms were tested to construct the line from the image. The traditional Hough line detection method has a problem in the test [18]. The way of finding the rising peak from the line edges does not perform very well in this case due to 1) saturation in the center of line; and 2) broken segment due to distortion. A robust line detection algorithm with sub-pixel level precision is desired. After tests, an algorithm based upon the approach by Trujillo-Pino et. al. [19] is implemented. The edge is located accurately based on orientation, intensity difference at both sides, subpixel position, and curvature. The detection results are shown in the right picture of Fig. 8. Once the edges are detected, we use the average location for both edges as the center and find the line.

Once lines are detected, a linear fit is performed to evaluate the straightness. Fig. 9 shows the linear fit of one of the lines

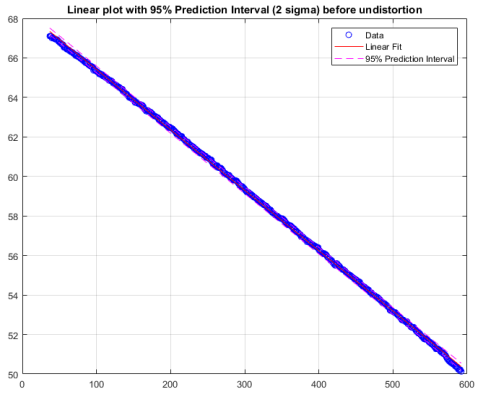
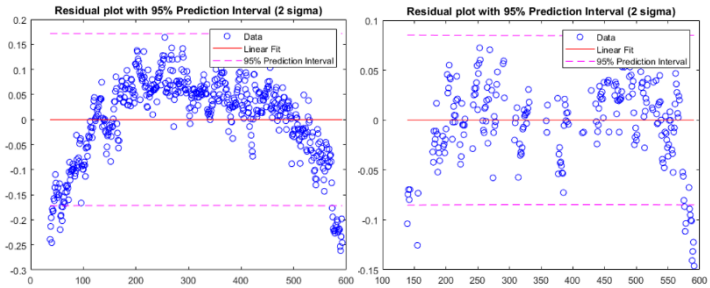


Figure 9. Linear fit of the line to evaluate

before applying lens distortion correction to the image. Fig. 10



a) Before b) After

Figure 10. Residual error of the line fitting before and after distortion correction

shows the residual errors of the line fitting before and after applying distortion correction. The horizontal axis draws the point numbers on the line. The tested line has about 600 points. The vertical axis is the residual error of each point in pixel units. Before the correction, as shown in Fig. 10a), there is a strong bending pattern and the residual error range is from -0.25 to 0.15 (in pixels). Fig. 10b) shows the improved results after correction. The residual error is significantly reduced. The residual error range is from -0.05 to 0.05 (in pixels) after distortion correction. The bending pattern is removed as well. The test demonstrates the effectiveness of the calibration results. This can also be used as a quick check of the vision-based system's accuracy before performing measurements.

ESTABLISH STEREO CALIBRATION PROCEDURE

In previous sections, an automatic camera calibration procedure for automatic corner detection and single camera calibration is presented. In this section, the stereo camera calibration procedure is presented.

The task of developing an automatic stereo camera calibration method is more challenging than the method for automatic single camera calibration. During stereo calibration, operators have to fit the calibration board into both cameras' field of view (FOV). Sometimes the calibration board looks smaller in the image, more tilted, and having more background, as shown in Fig. 11. These conditions present more problems. Stereo

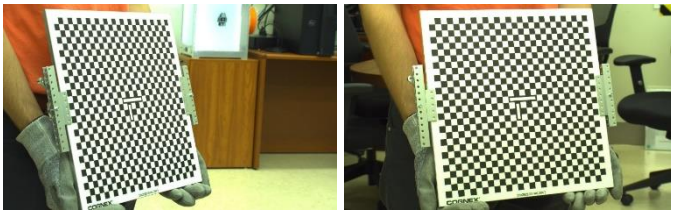


Figure 11. One pair of images from left and right cameras used for stereo calibration

calibration requires more robust corner detection algorithms to handle the complicated background. Fig. 12 shows all the

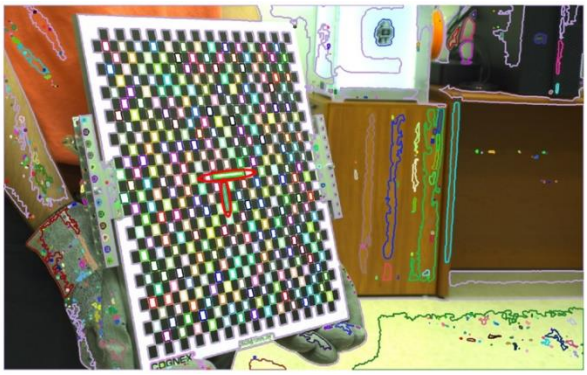


Figure 12. Detected contours from the stereo calibration image

contours detected from the image. The two rectangles inside the red circles are what we used for the 'anchor'. Fig. 13 shows the contour detected for an image in the single camera calibration

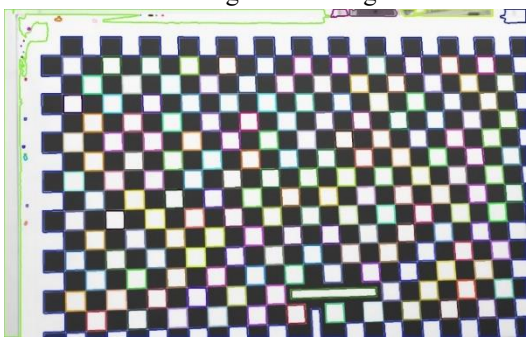


Figure 13. Detected contours from the single camera calibration image

case, which shows the levels of challenges stereo auto-calibration is facing compared with Fig. 12. It is a challenging task to pick out all the calibration board corners from all the detected contours.

Significant improvements to the algorithms are developed for corner auto-detection in stereo calibrations. One of them is the square and rectangle detection algorithm. Because all the squares and rectangles do not appear perfect when the board looks smaller, the new algorithm relaxes the detection criteria and implements a new concept of "two-point contour" to better detect the rectangle. The iteration procedure for corner searching is also modified in the final step to remove the remaining false detections. Fig. 14 shows the detected corners using the improved algorithms. Results show significant improvements compared with the results shown in Fig. 12.

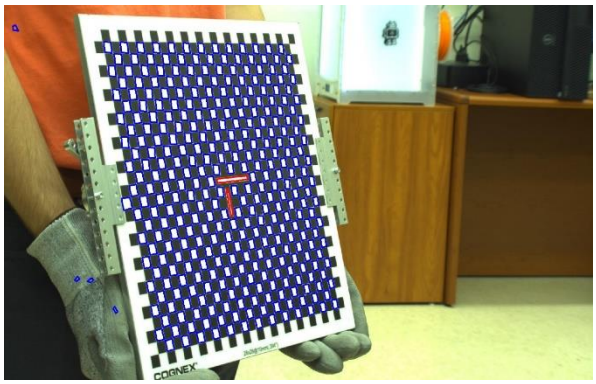


Figure 14. Improved detected corners from the stereo calibration image

Since no need to calibrate distortion, the checkerboard is placed parallel to the two camera's centerline. About 3-6 positions are needed to be placed from the near to far range.

Even with the improvements, some extreme cases may fail auto-detection when images are very blurry. To address this issue, an algorithm is developed to detect the condition and remove the blurry images automatically. These low-quality images are highlighted to users. They can be used to train users in understanding the image quality and improving skills when taking measurements to provide better quality calibration images.

The calibration final results are saved in both xml and yaml format for easy export/import between Python, C++, or Matlab code. The saved information includes: left camera matrix, right camera matrix, left camera distortion coefficients, right camera distortion coefficients, 3x3 rotation matrix between two cameras, 1x3 translation matrix between two cameras, fundamental matrix, essential matrix, and the projection error.

In summary, automatic camera calibration methods are developed using a calibration board with markers. The auto-calibration procedure includes both single camera calibration and stereo camera calibration. Advanced algorithms are developed to improve calibration accuracy via improving corner detection in sub-pixel accuracy. Improvements are also made for algorithm robustness to enhance the handling of the complex

background. This auto-calibration procedure eliminates the need for users to click hundreds of times and significantly reduces the problems caused by human errors.

CONCLUSION

Manufacturers are facing challenges in the robot's accuracy assessment and accuracy improvement. NIST's development of the vision-based 6-D sensing system enables the capture of the robot's dynamic movements in high accuracy. This sensing technology provides a solution to allow the robot system's health monitoring and assessment. As the foundational supporting technology, a high-efficiency and high-accuracy vision-based system calibration procedure is critical for real applications. The developed sub-pixel corner extraction and iteration improvements enable the high accuracy of sensing system calibration. The automatic procedure and the separation of the distortion calibration with stereo calibration significantly reduce the time for on-site adjustment. The verification method enables the quick check of the system accuracy. The 6-D sensing system was used on a Universal Robot (UR5) to monitor and assess the robot accuracy degradation with different payloads, speeds, and temperatures. The test data set was published in [20]. NIST is actively seeking to develop additional industrial use cases using the 6-D sensing system for further applications.

NIST DISCLAIMER

Certain commercial entities, equipment, or materials may be identified in this document in order to illustrate a point or concept. Such identification is not intended to imply recommendation or endorsement by NIST, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

REFERENCES

[1] E. Khalastchi and M. Kalech, "Fault Detection and Diagnosis in Multi-Robot Systems: A Survey," *Sensors (Basel),* vol. 19, no. 18, p. 4019, 2019, doi: 10.3390/s19184019.

[2] M. A. Costa, B. Wullt, M. Norrlöf, and S. Gunnarsson, "Failure detection in robotic arms using statistical modeling, machine learning and hybrid gradient boosting," *Measurement :Journal of the International Measurement Confederation,* vol. 146, pp. 425-436, 2019, doi: 10.1016/j.measurement.2019.06.039.

[3] S. Fan, L. M. Zhang, J. Wang, Y. F. Wang, Q. S. Zhang, and H. Zhao, "Vision-Based Fault Classification for Monitoring Industrial Robot," *2018, Conference Proceedings: Technical Committee on Control Theory, Chinese Association of Automation*, pp. 5889-5894, doi: 10.23919/ChiCC.2018.8483793.

[4] E. Sita, T. Thomessen, A. G. Pipe, M. Studley, and F. Dailami, "Usability Study of a Robot Companion for Monitoring Industrial Processes," 2020, *Conference Proceedings: IEEE, p*p. 37-42, doi: 10.1109/ACIRS49895.2020.9162607.

[5] R. N. A. Algburi and H. Gao, "Health Assessment and Fault Detection System for an Industrial Robot Using the Rotary Encoder Signal," *Energies (Basel),* vol. 12, no. 14, p. 2816, 2019, doi: 10.3390/en12142816.

[6] S. K. Mustafa, P. Y. Tao, G. Yang, and I. Chen, "A geometrical approach for online error compensation of industrial manipulators," in *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 6-9 July 2010 2010, pp. 738-743, doi: 10.1109/AIM.2010.5695784.

[7] B. C. Jiang, R. Duraisamy, G. Wiens, and J. T. Black, "Robot metrology using two kinds of measurement equipment," *Journal of Intelligent Manufacturing,* vol. 8, no. 2, pp. 137-146, 1997/03/01 1997, doi: 10.1023/A:1018508805175.

[8] J. H. Jang, S. H. Kim, and Y. K. Kwak, "Calibration of geometric and non-geometric errors of an industrial robot," *ROBOTICA,* vol. 19, pp. 311-321, 2001.

[9] S. Wang, S. Liu, and Q. Mao, "A CMM-Based Method of Control Point Position Calibration for Light Pen Coordinate Measuring System," *Sensors,* vol. 20, no. 19, p. 5592, 2020, doi: 10.3390/s20195592.

[10] A. Wiles, D. Thompson, and D. Frantz, "Accuracy assessment and interpretation for optical tracking systems" *SPIE 2004*. Proceedings vol. 5367, 2004, doi:10.1117/12.536128.

[11] T. Sun, Y. Zhai, Y. Song, and J. Zhang, "Kinematic calibration of a 3-DoF rotational parallel manipulator using laser tracker," *Robotics and computer-integrated manufacturing,* vol. 41, pp. 78-91, 2016, doi: 10.1016/j.rcim.2016.02.008.

[12] G. Qiao and B. A. Weiss, "Industrial Robot Accuracy Degradation Monitoring and Quick Health Assessment," *Journal of Manufacturing Science and Engineering,* vol. 141, no. 7, 2019, doi: 10.1115/1.4043649.

[13] G. Qiao, C. Schlenoff, and B. A. Weiss, "Quick positional health assessment for industrial robot prognostics and health management (PHM)," 2017, *Conference Proceedings: IEEE*, pp. 1815-1820, doi: 10.1109/ICRA.2017.7989214.

[14] O. Nina, B. Morse, and W. Barrett, "A recursive Otsu thresholding method for scanned document binarization," in *2011 IEEE Workshop on Applications of Computer Vision (WACV)*, 5-7 Jan. 2011 2011, pp. 307-314, doi: 10.1109/WACV.2011.5711519.

[15] T. Panchal, H. Patel, and A. Panchal, "License Plate Detection Using Harris Corner and Character Segmentation by Integrated Approach from an Image," *Procedia Computer Science,* vol. 79, pp. 419-425, 2016/01/01/ 2016, doi: https://doi.org/10.1016/j.procs.2016.03.054.

[16] A. Datta, J.-S. Kim, and T. Kanade, "Accurate camera calibration using iterative refinement of control points,"

2009, *Conference Proceedings: IEEE*, pp. 1201-1208, doi: 10.1109/ICCVW.2009.5457474.

[17] K. H. Strobl and G. Hirzinger, "More accurate pinhole camera calibration with imperfect planar target," 2011, *Conference Proceedings: IEE*E, pp. 1068-1075, doi: 10.1109/ICCVW.2011.6130369.

[18] C. Galamhos, J. Matas, and J. Kittler, "Progressive probabilistic Hough transform for line detection," in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, 23-25 June 1999 1999, vol. 1, pp. 554-560 Vol. 1, doi: 10.1109/CVPR.1999.786993.

[19] A. Trujillo-Pino, K. Krissian, M. Alemán-Flores, and D. Santana-Cedrés, "Accurate subpixel edge location based on partial area effect," *Image and vision computing,* vol. 31, no. 1, pp. 72-90, 2013, doi: 10.1016/j.imavis.2012.10.005.

[20] G. Qiao, "Degradation measurement of robot arm position accuracy," 2019, doi: https://www.nist.gov/el/intelligent-systems-division-73500/degradation-measurement-robot-arm-position-accuracy.