doi: 10.4302/plp.v14i4.1181 PHOTONICS LETTERS OF POLAND, VOL. **14** (4), 83-85 (2022)

Matrix data analysis methods for application in laser beam position measurement modules

Grzegorz Budzyn,*[1] Jedrzej Baranski,[1,2] and Janusz Rzepka[2]

[1]*Faculty of Electronics, Photonics and Microsystems, Wroclaw University of Science and Technology, Janiszewskiego 9, 50-370 Wrocław,*
[2]*Lasertex Co. Ltd, Swojczycka 26, 51-501 Wroclaw*

Received November 10, 2022; accepted December 20, 2022; published December 31, 2022

Abstract—This paper addresses the problem of selecting an optimal algorithm suitable for real-time processing of matrix sensor data in laser beam positioning applications. We compare four different algorithms and prove that the chosen one provides the same results in time, by several orders of magnitude lower and with negligible memory consumption, making it suitable for use in microcontroller-based sensing modules.

Laser technology is currently used in many areas of life, science and industry. This is possible due to the wide availability of various laser sources with parameters tailored to users' requirements, and due to the great convenience of using laser equipment. One of the areas in which laser sources are commonly used is the measurement of the geometry of numerically controlled machines and the quality of movement in the individual axes of the machine [1–3]. A laser source with a good quality and stable output beam position establishes a reference line in space to which the actual movement of the machine is compared. While the source requirements are now easily met by the use of semiconductor lasers coupled to a single-mode fiber optic cable, a bigger problem is the quality and accuracy of the laser beam position detector.

Two main types of detectors are used in the industry: a solid-state position sensor detector PSD (usually a tetralateral version) and an array detector in the form of a CCD or CMOS camera [4–6]. The first detector is characterized by simplicity of design, but also high nonlinearity and sensitivity to the quality of the incident beam. Much higher accuracy can be achieved with array detectors. Their disadvantage, however, is the wide data stream that must be analyzed, and the consequent need for a sophisticated signal processing system with high power consumption and complexity.

In this paper, we present a novel algorithm for processing data from matrix (camera) sensors, which allows to significantly reduce the requirements for signal processing circuitry while maintaining the required accuracy and linearity of laser beam position detection. The algorithm can be implemented in the standard structures of available 32-bit single-chip microcontrollers.

The problem of measuring the geometry of CNC machines has been described many times in the literature, such as [7–8]. One non-trivial problem is measuring the straightness of a machine or machine axis. There are several methods with and without a laser. Methods that do not require a laser source (such as an autocollimator or steel string) tend to be simpler and less expensive, but provide lower-quality results over longer periods of time. Most standard laser methods require a complex laser source and dedicated optics, as shown schematically in Fig. 1. With such an arrangement, it is possible to measure straightness of motion over a range of a few millimeters with an accuracy of one micrometer and a resolution of less than 100 nanometers. Such good performance requires not only a complex optical system, but also an expensive frequency-stabilized laser head.

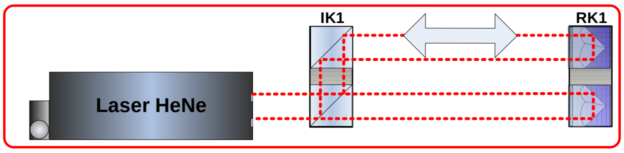

Fig. 1. Machine straightness measurement method with the use of He-Ne laser source and dedicated optics.

Simplification of the measurement configuration is possible when the amplitude principle is used instead of the frequency principle. In this case, the beam position of a single-mode laser is measured in the arrangement shown in Fig. 2. The simpler and cheaper laser source and much simpler optics make this configuration very tempting from a practical point of view. The main limitation of this configuration is the effect of air turbulence and the quality of the beam position detector used.

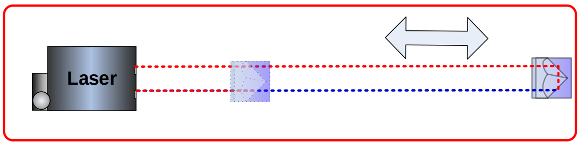

Fig. 2. Machine straightness measurement method with the use of single mode laser source and dedicated optics.

* E-mail: grzegorz.budzyn@pwr.edu.pl

Currently, there are three main types of sensors used in beam position detectors: quadrupole diodes, PSD sensors and camera sensors. The first type is not suitable for detecting positions off-center of the detector. The PSD sensor is much more linear (especially the cushion type), but careful calibration over the entire measurement range is still required. There is also the problem of the sensor's low sensitivity and high susceptibility to air fluctuations in the beam path. These problems are overcome by a third type of sensor - the camera sensor. Most available camera sensors are linear and highly sensitive to the laser beam. It is also possible to detect different beam shapes, for example, of the Bessel type. The main disadvantage of this type of sensor is the high demand for computing power. This problem leads to noticeable power consumption and large size of the detection module. In this paper, we present data processing methods that solve this problem. We focus on processing of camera sensor data algorithms used especially in small microcontroller modules.

One of the optical source of choice for measuring the straightness of CNC axes today is the DFB semiconductor laser with SMF single-mode fiber optic pigtail. The output beam profile is Gaussian type and is described by the equation:

$$f(x,y) = Aexp\left(-\left(\frac{(x-x_0)^2}{2\sigma_X^2}+\frac{(y-y_0)^2}{2\sigma_Y^2}\right)\right), \tag{1}$$

where x_0 and y_0 are the coordinates of the beam center, A is a constant, σ_x and σ_y are the beam parameters. The simplest detection method would be to fit a camera image to a two-dimensional curve. A useful algorithm can be found in [9] or [10]. We would call this method Gauss2D. The method requires a significant amount of memory and considerable computing power. The complexity of the algorithm is also of the order of O(n^2), which is very unfavorable for high-resolution sensors.

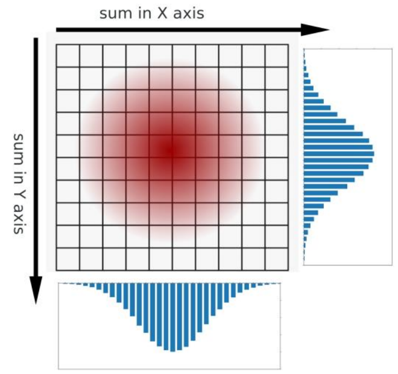

Fig. 3. Rule of histogram preparation used in Gauss1D, COG and COGv2 algorithm.

The two-dimensional Gaussian fitting algorithm can be simplified when the pixels are summed vertically and horizontally, as shown in Fig. 3. The histograms prepared in this way, vertical and horizontal, can then be fitted to two one-dimensional Gaussian functions [9]. This method is referred to as Gauss1D. As will be shown further on, the beam position results obtained by this method are very similar to the Gauss2D method, but with much less CPU load and much less memory usage.

Further simplification is possible when the one-dimensional Gauss fit is replaced by the much simpler center-of-gravity COG algorithm. The value of the beam position is then calculated from the formula:

$$COG = \frac{m_1 \cdot d_1 + m_2 \cdot d_2 + \cdots + m_n \cdot d_n}{d_1 + d_2 + \cdots + d_n}, \tag{2}$$

where m_i is the index of i-th histogram bin and d_i is the value of the bin. This algorithm requires only very basic operations of addition and multiplication. In the microcontrollers supporting single cycle Multiply-and-Accumulate MAC instructions this algorithms can be calculated very efficiently.

A common problem with the aforementioned algorithms is the need to buffer the entire camera frame before calculations. In small microcontrollers, this is a problem because the amount of available high-speed SRAM is usually limited. Our idea to solve this problem is to calculate the histogram of the image in each line, rather than in each frame. In this case, it is required that the MCU has enough memory only for a single line and histogram tables. In our tests, for optimal performance, we decided to calculate the position based on the COG algorithm, hence the designation of the described method is COGv2.

In order to compare the described algorithms, we prepared an optical system schematically shown in Fig. 4. The complete system was placed on a calibrated CNC machine with a pre-loaded control program for automatic testing. A monochrome global shutter camera with a resolution of 1920×1200 pixels was used as the array matrix. The camera data was then preprocessed on a 32-bit digital signal controller platform based on an STM32H743 processor clocked at 500 MHz. All algorithms except Gauss2D were implemented directly on the processing platform. Due to implementation issues, Gauss2D was only tested in a Matlab environment on a standard PC.

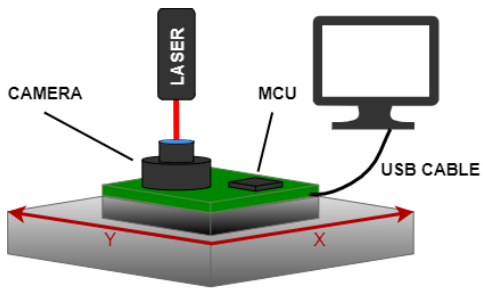

Fig. 4. Block diagram of measurement setup.

During repeated tests, the main expected parameters of the detector were checked using all the above-mentioned algorithms. An example of one of the linearity measurements is shown in Fig. 5. In this measurement, the reference machine was moved along one of the axes by the useful measurement distance of the detector, i.e. 4000 µm. Since in this particular measurement the detector axes were not completely parallel to the axis of the reference machine, hence the value of Position X in the graph shown in Fig. 6 varies by about 24 µm. This does not affect the generality of the analysis. The measured uncertainty of the reference machine was 2.1 µm. The uncertainty also includes air fluctuations between the laser source and the array detector.

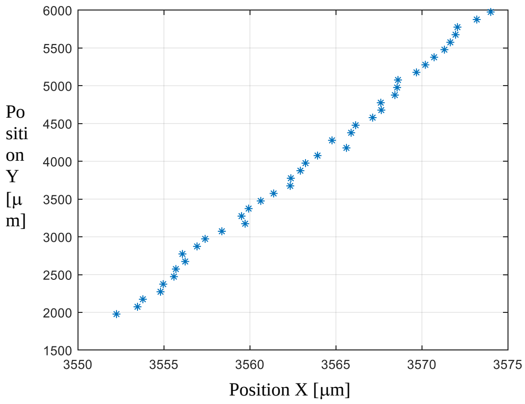

Fig. 5. Linearity measurement with Gauss2D algorithm. Reference machine position uncertainty σ=2.1 µm.

After eliminating the trend, the value of the position deviation in Y axis is shown in Figure 6. In the graph, the results obtained with the COGv2 algorithm have been added for comparison. Other algorithms (Gauss1D and COGv1) have not been added for graph clarity. The difference between the presented algorithms is within the uncertainty of the reference machine. The results obtained with the algorithms not shown were also within the uncertainty of the reference machine.

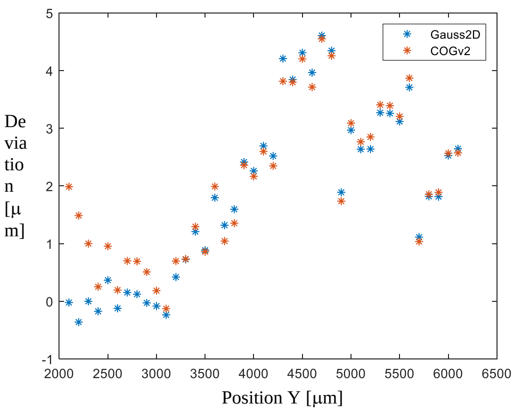

Fig. 6. Measured deviation of reference machine position for axis Y for two calculations algorithms: Gauss2D and COGv2.

The difference between the algorithms is clearly visible in Table 1 which shows the execution time of each algorithm for analyzing the 100 image data and an estimate of the required resources for analyzing a single image.

Table. 1. Comparison of execution time and required memory of analyzed algorithms.

	Execution time [s]	Required memory [MB]
Gauss2D	7.459	4493.1
Gauss1D	0.415	4.7
COGv1	0.015	4.6
COGv2	0.015	0.1

Due to problems with the implementation of the Gauss2D algorithm on the MCU platform, the comparison was performed in the Matlab environment on a PC. The results indicate the very significant drawbacks of the Gauss2D algorithm, i.e., significantly longer execution time and extremely high memory requirements. For this reason, we were unable to implement this algorithm on an embedded platform. The execution time of Gauss1D was much shorter than Gauss2D, but also much longer than the COG algorithms. Both COG algorithms offer similar execution times, but with much lower memory requirements for the COGv2 version.

In summary, laser beam positioning modules for industrial applications should guarantee accuracy and repeatability in the simplest form possible. As we have shown in the paper, it is possible to build a module based on a high-linearity camera consisting a power- and cost-efficient single-chip microcontroller. This is possible because the very simple COGv2 algorithm provides comparable accuracy and repeatability to the standard two-dimensional Gaussian fitting method, but in much less time and with negligible memory required.

This research was supported by NCBiR under project POIR.01.01.01-00-1609/20.

References

[1] S. Donati, *Electro-Optical Instrumentation—Sensing and Measuring with Lasers* (Prentice Hall, Upper Saddle River, NJ, 2004).
[2] K.N. Joo, J.D. Ellis, E.S. Buice, J.W. Spronck, R.H. Munning Schmidt, Optics Expr. **18**(2), 1159 (2010).
[3] G. Budzyń, J. Rzepka, J. Machine Engin. **20**(3), 76 (2020).
[4] Reference materials for PSD and CMOS sensors, www.hamamatsu.com, referenced 7-11-2022
[5] S. Das, A. Saha, IEEE Trans. Instr. Meas. **68**(9), 3308 (2019).
[6] P. Zhang, J. Liu, H. Yang, L. Yu, IEEE Access **7**, 140319 (2019).
[7] J. Rzepka, G. Budzyn, Proc. SPIE **5144**, 840 (2003).
[8] S. Xiang, H. Li, M. Deng *et al.*, Int. J. Adv. Manuf. Technol. **96**, 2915 (2018).
[9] N. Hagen, E. Dereniak, App. Opt. **47**(36), 6842 (2008).
[10] S.M. Ross, *Introduction to Probability and Statistics for Engineers and Scientists* (Academic Press 2014). doi.org/10.1016/B978-0-12-394811-3.50017-4

