euspen's 22nd International Conference & Exhibition, Geneva, CH, May/June 2022
www.euspen.eu

Improving position accuracy in large-scale laser structuring processes using surface feature detection algorithms

Martin Kohse[1], Tobias Hauffe-Waschbüsch[1], Robin Day[1], Thomas Bergs [1,2]

[1]*Fraunhofer Institute for Production Technology*
[2]*Laboratory for Machine Tools and Production Engineering (WZL)*

martin.kohse@ipt.fraunhofer.de

Abstract
Laser structuring is an effective way to functionalize or texture components. For the machining of large components > 500 x 500 mm, precision machine tools costing up to €1.5 mio are needed. This circumstance leads to a limited market applicability. Fraunhofer IPT is developing a cost-efficient laser structuring solution making it possible to scale up maximum processing area by mounting the laser structuring head on an industrial robot. However, high positioning accuracy of < 5 µm is needed for laser structuring while an industrial robot system delivers a maximum position repeatability of about 70 µm.
Fraunhofer IPT has developed a compensation strategy using an inline camera setup. The system detects reference features of markers and texture by using FAST algorithm for corner detection with a laplacian image pyramid. The current system is capable to set exposure times automatically and to compensate position inaccuracies of up to 1 mm to an overall accuracy of < 5 µm despite a camera chip resolution of 5.3 µm/px.
In this contribution we present an overview of the compensation algorithm with an evaluation method for process stabilization and identify challenges, e.g. illumination and material properties will be addressed.

Laser structuring, Robot, large parts, computer vision, process control

1. Introduction

Laser structuring is used, among others, to repeatedly mark or engrave complex 3D geometries. In this application, a focused, pulsed laser beam scans a target surface using high precision galvo-scanner [1]. Since the scanning field is limited, large surfaces must be divided into sufficiently small patches. Here, precise repositioning of the structuring system is of major importance; deviations of >5 µm lead to optically recognizable seams. The demand on positioning accuracy, thus, necessitates reliable high-precision system technology. Processing large moulds – e.g. bumpers, instrument panels or centre consoles for the automotive industry – requires complex and cost intensive plant technology [2]. Investment costs for systems with a working volume of 1 m^3 quickly exceed €1.5 mio, whereas only a small fraction (approx. €50k - €70k) are accounted for by the laser system. Industrial robots present a cost-efficient alternative for these kinds of laser processes. Today, such systems are mainly used in laser material processing for laser cutting or laser welding [3].

Such systems have not been used industrially for laser ablation, as the positioning accuracy of robot systems is not sufficient for this purpose. Currently, pose repeatability of such industrial robot systems cannot exceed approx. 0.3 mm [4]. There are several ways of dealing with robot positioning inaccuracies. For example, path and positioning deviations can be compensated for by using external laser trackers or stereo cameras [5]. Such systems would increase the total costs by €100k to €150k with the resulting increase in accuracy still insufficient for the purpose of laser structuring.

We have developed an approach that accepts the robot-related positioning inaccuracies of positioning the laser module and then compensates for the resulting error using an inline-camera based optical reference system.

2. Technical approach

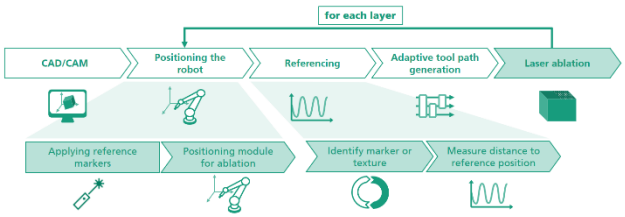

Figure 1: Developed process chain for robot based laser structuring

The compensation method is based on system components required for laser structuring. Figure 1 shows the developed process chain. The first process step "CAD/CAM" pre-calculates tool paths for the laser module as well as the robot that positions it. In the next step, "Positioning the robot," markers are applied by the laser system at defined positions on the surface. These markers serve both as anchor points for the texture and as a means for determinating the robot system's position in the following steps when the laser texture is applied. The marker positions are approached individually by the robot system and positional inaccuracies of the system are deliberately tolerated. At each marker position the integrated camera system takes a reference picture, which are then compared with images taken using the high-precision galvo-scanner in the later texturing process. Now, the robot successively positions the laser module above each segment such that the optical axis is located at its center. Before the laser structuring process starts, the actual position of the laser module in the current segment with respect to the reference markers needs to be determined. The process step "Referencing" deals with the camerabased detection of

positioning errors. Each segment contains three associated markers that enable the triangulation of the current position. Acquisition of the actual images is carried out by the high-precision galvo-scanner using the integrated camera system. Each recorded image of the actual position is then compared with the corresponding reference image. For this purpose, we developed a library based on OpenCV algorithms to identify identical features in reference and actual images; using that information, we calculated a displacement vector for each marker. In the next process step "Adaptive toolpath generation," the displacement vectors for the corresponding segment markers are used to adapt the pre-calculated tool path for the respective segment on the fly. The corrected tool path is now executed in the last process step "Laser ablation," resulting in no recognizable seams between the segments. These process steps are repeated for each segment in each layer, respectively.

2.1. Setup

Figure 2 shows the developed laser structuring module with its beam path illustrated. The robot used for module handling in this study is an ABB IRB 6660-205 industrial robot especially designed for high-performance applications. Due to its stiff design with parallel arm structure, it delivers a pose repeatability of about 70 µm. The maximum reach of the robot arm is 1.93 m extended with a linear axis of 6 m travel range. The maximum payload is 205 kg, where the laser structuring module weighs 24 kg. The laser structuring module is held by the industrial robot using a Schunk pneumatic quick tool changing adapter (SWS).

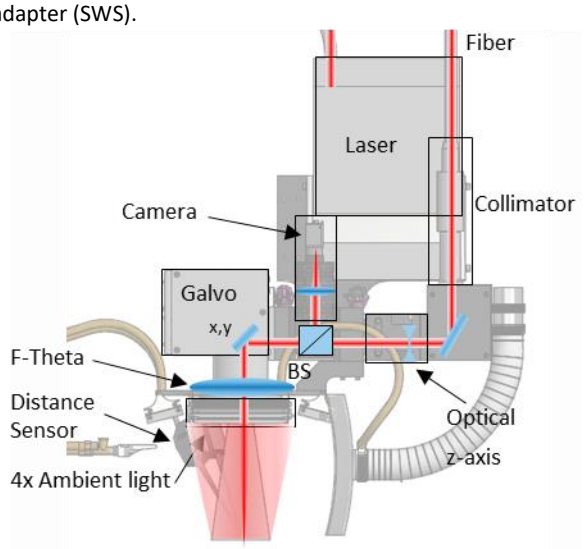

Figure 2: Developed laser module with beam path illustrated.

A nanosecond pulsed SPI G4 (model SP-070P-A-EP-Z) laser source provides pulsed laser radiation (10 ns - 520 ns) with a maximum power of 70 W and pulse energies up to 1 mJ at a central emission wavelength of 1064 nm with a Gaussian intensity distribution. A collimator (ILLK f30 d35) is used to feed the laser beam into the designed beam path collimating the beam to a diameter of 3 mm. The beam passes through an optical z-axis (ScanLab VarioScan20i_de) with a beam expansion factor of 2.8. With the optical z-axis the laser beam can be focused in 3D from the working plane by ±12 mm in the z-direction. The beam then passes through a dichroic beamsplitter (Scanlab camera adapter), which is transmittive for 1064 nm laser wavelength and reflective for the obervastion wavelength of 860 nm provided by four external high power output LED modules (Stemmer CCS HLDL2). The camera (IDS UI-5240CP-NIR) is especially designed for high quantum efficiency in the NIR range with a resolution of 1280 x 1024 pixels at a pixel size of 5.3 µm/px. A galvanometer scanner (ScanLab IntelliScan 14se) moves the laser beam in x-and y-direction with high position repeatability (< 1 µm). The beam is focused by a F-Theta lens (163 mm). A Keyence laser triangulation system is used to measure the module's distance to the working plane for laser focus adjustment while processing.

2.2. Design of reference markers

Reference markers are used as an achor point for the texture and as a means to determine the position of the module with regard to the respective segment. The feature detection algorithm (ref. Chapter 3.1) is based on corner identification. For the referencing, it is of utmost importance to detect as many features as possible in the reference- and actual images. Accordingly, the markers (ref. Figure 3) consist of a large number of 90 degree angles. A cross divides the marker into four quadrants with rectangular structures in each quadrant. Additionally, the cross provides orientation for subsequent measuring of possible texture displacement.

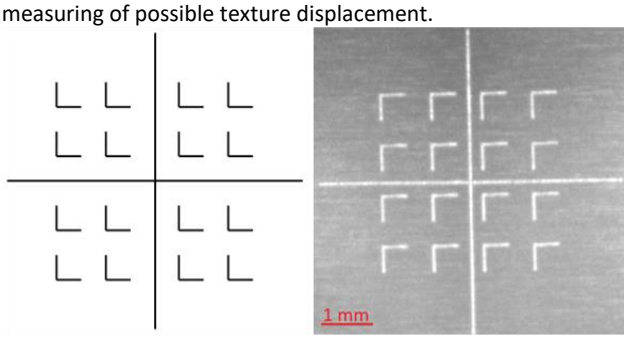

Figure 3: Developed marker as CAD- and as a reference image

3. Methodology

3.1. Corner feature detection

The compensation method is based on feature detection algorithms originating from the field of computer vision. An image is reduced to significant features, characerized by uniqueness and recognizability. For this purpose, properties such as position, direction or brightness are assigned to each feature [6]. Open source algorithms provided by the openCV library were used for feature identification in reference and actual image [7]. The openCV library offers different algorithms for feature detection. One is the FAST algorithm. It was especially developed to detect features in real time for video applications. It detects corners by iterating through each pixel and comparing the brightness levels with the surrounding 16 pixels. If there are at least 12 pixels with a significant change in brightness, the position is saved as a corner feature [8]. Another algorithm used is the ORB algorithm. It combines the FAST algorithm functionality for feature determination with the BRIEF algorithm. The FAST detection algorithm within ORB is executed several times in the same image. For this purpose, the image is scaled for each execution using a Laplacian image pyramid. Thus, the feature detection is more stable against disturbances such as image noise [9]. Figure 4 shows the extracted features using the ORB algorithm from the CAD marker. Each corner and each end of the angular structures are detected reliably.

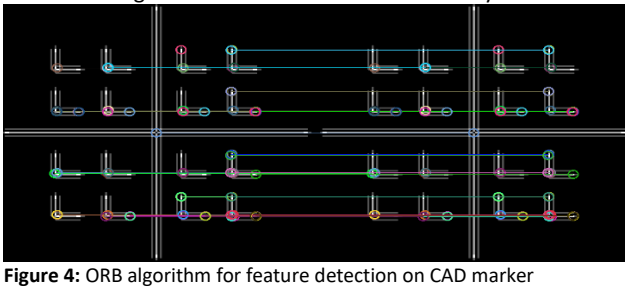

Figure 4: ORB algorithm for feature detection on CAD marker

3.2. Calculation of displacement vector

In the process step "Referencing," the position error of the handling system is calculated. This calculation is based on comparing features in the reference- and actual image for each marker in each segment. Due to reflections to the camera originating from metallic surfaces, the exposure time of the camera system is automatically set so that each histogram is linearly distributed between 0 and 255. The ORB algorithm is used to detect features and to match corresponding feature pairs in the reference- and actual image. In the first step the most significant 200 corner features are identified in the reference image. Due to high image resolution, a Laplacian image pyramid technique is used to scale down the image to test the identified corner for validity. In contrast to the reference image, the maximum number of features in the actual image is not limited, which increases the chance of finding corresponding features. For each feature in the reference image, the actual image is scanned at the same position within a 50 pixel radius for features with similar feature properties. This limits the maximum detectable position deviation of the system to 265 µm. Every potential pair within the radius is assigned a score. The pair with the highest score is defined as a feature pair. If the score does not reach a certain threshold, a pair will not be used for further calculations. For each feature pair found, the displacement vector $\mathbf{d_n}$ is calculated. When the mean displacement vector $\overline{\boldsymbol{d}}$ is used, feature pairs are discarded if they deviate by more than half of the standard deviation. To improve the measured final position error even more, an optimization algorithm is implemented. The remaining distance $\mathbf{d_R}$ between the mean displacement vector $\overline{\boldsymbol{d}}$ and the corresponding pair displacement vector $\mathbf{d_n}$ is minimized using the iterative gradient descent algorithm [10]:

$$\overline{\boldsymbol{d}}_{n+1} = \overline{\boldsymbol{d}}_n - \gamma \nabla \boldsymbol{d}_R$$

with

$$\gamma = \frac{\boldsymbol{d}_R}{\|\nabla \boldsymbol{d}_R\|^2}$$

Figure 5 shows an illustration of the result after feature detection and pair matching. When a feature *A* in the reference image is detected in the actual image (A') as well, the corresponding displacement vector $\mathbf{d_1}$ is calculated. A resulting mean displacement vector $\overline{\boldsymbol{d}}$ is calculated using $\mathbf{d_n}$. The remaining distance $\mathbf{d_R}$ is minimized with regard to each displacement vector $\mathbf{d_n}$ and mean displacement vector $\overline{\boldsymbol{d}}$. The termination criterion of the optimization algorithm is either the 100th iteration step or an increase of the remaining vector $\mathbf{d_R}$. In this case the step size is halved to a minimum of 1/128 if the remaining distance increases instead of decreasing.

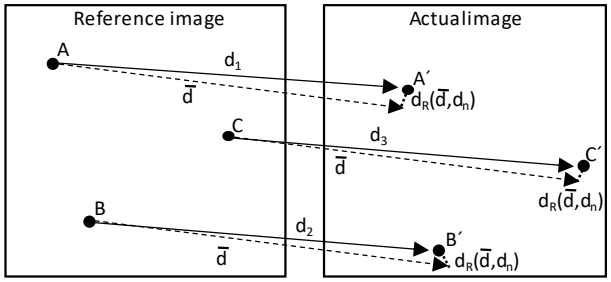

Figure 5: Principle of displacement vector calculation using optimization algorithm

3.3. Evaluation of displacement vector by score

In the robot-based laser-structuring process, several disturbance variables impair the image acquisition, e.g. reflections or oscillations of the laser module itself. If the actual image differs substantially from the reference image, either no feature pairs or incorrect feature pairs are detected. An evaluation system is implemented to handle these scenarios by evaluating the resulting displacement vector as a function of the number of detected feature pairs. The function

$$S(\overline{d}_n, n) = e^{-\overline{d}_n}\,(1 - e^{3-n})$$

can take values between 0 and 1, where 0 represents no pair feature detection and 1 ideal feature detection.

4. Results

4.1. Feature detection algorithms in real szenario

Figure 6 shows the results of pair identifaction in the same real process image using FAST algorithm and ORB algorithm as described in Chapter 3.1. When the FAST algorithm is used, the steel texture or noise in the image is more relevant to the algorithm than the applied marking for feature detection. Due to different image acquisition scenarios in the process, noise cannot be used as reliable feature source. Using the ORB algorithm with the Laplacian image pyramid, the noise is detected less as a relevant feature pair in comparison to the FAST algorithm. Most feature pairs originate from the applied marking despite difficult background reflections.

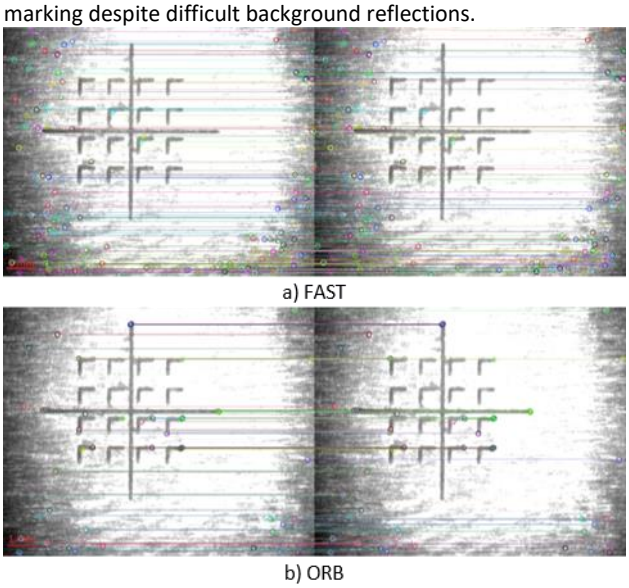

Figure 6: Feature detection algorithms applied to identical images. a) FAST Algorithm b) ORB Algorithm as implemented in Chapter 3.1.

3.1. Statistical evaluation

The feature detection algorithm implemented was statistically analyzed for achievable accuracy. For this purpose, three different scenarios were considered. Figure 7 shows the detected displacement in a stationary setup.

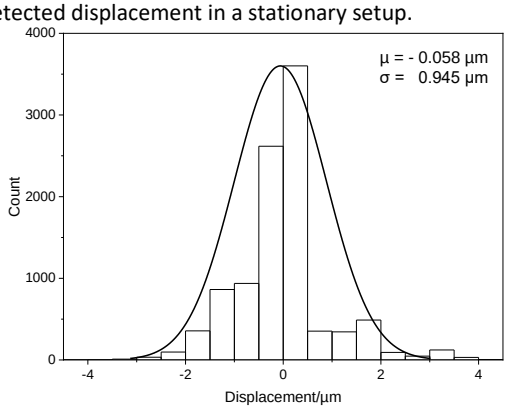

Figure 7: Detected displacement in a stationary setup

The image acquisition occurred 10 000 times at the 0,0 position in the F-Theta image field. Since neither the laser module nor the scanner moved, a resulting displacement vector of 0 µm is

expected. The results show a mean value μ = -0.058 µm, close to zero, with a standard deviation of σ = 0.945 µm. Due to the camera pixel size of 5.3 µm, a shift of brightness by one pixel can have a strong impact on the displacement vector. The implemented algorithm reduces this influence significantly.

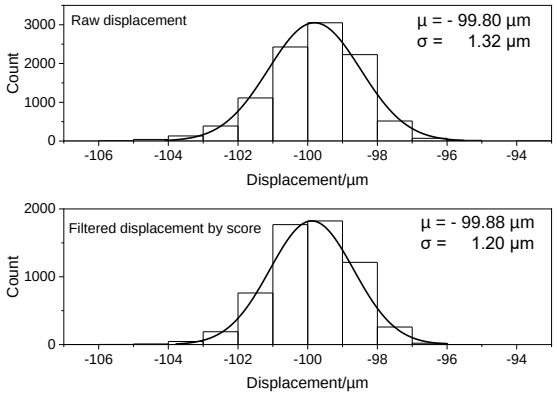

Figure 8: Detected displacement in a known displacement scenario of 100 µm

For validation of the achievable accuracy, the high precision galvanometer scanner is used to take images in several positions of the F-Theta image plane. Figure 8 shows the detected displacement when the galvonameter scanner is moved by 100 µm in the x-direction; here an actual image is acquired for displacement detection with a sample size of n = 10 000. The achieved mean value is μ = - 99.80 µm with a standard deviation of σ = 1.32 µm. In comparison to the stationary setup, the standard deviation increased by 0.375 µm. This increase can be explained by the position accuracy of the galvanometer scanner (approx. 0.5 - 1 µm). Change of brightness in the actual image can lead to an increase of the standard deviation as well. The acquired data were filtered using the evaluation system $S(\overline{d}_n, n)$ (ref. Chapter 3.3). Every displacement vector with a score of $S(\overline{d}_n, n) < 0.3$ was discarded. This shifts the mean displacement to μ = - 99.88 µm and decreases the standard deviation to σ = 1.20 µm. Figure 9 shows the result when the developed algorithm is applied to a real robot-based structuring scenario. Due to the rather high position accuracy of the robot used (70 µm), a random positioning program was implemented to simulate handling systems with a position inaccuracy of about 200 µm radius around the desired position.

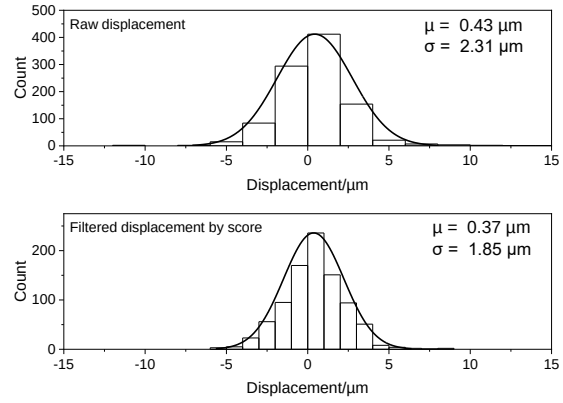

Figure 9: Detected displacement in an unknown robot induced displacement scenario

Due to the unknown position of the system with regard to the reference image, the displacement vector for each robot movement was applied to the galvanometer scanner. A third image is acquired and compared to the reference image. The mean displacement vector is expected to be around μ = 0 µm. The mean displacement vector after n = 1 000 robot movements is μ = 0.43 µm. The standard deviation σ = 2.31 µm is significantly higher compared to the tests without robot movement. One reason for this is that the compensation algorithm is executed two times to generate the result. When the evaluation system applied filters each displacement vector with a score < 0.4, the standard deviation can be decreased to σ = 1.85 µm. We assume that the standard deviation of each execution is independent of another. This leads to a standard deviation of σ = 1.85 µm.

Summary and Outlook

Within this research study, a low-cost industrial robot with novel technology is used for the first time to improve the accuracy of large-area laser structuring. The system-related inaccuracies of the robot are compensated for by using components native to standard laser structuring systems. We implemented an ORB algorithm for feature detection based on OpenCV library together with an optimization algorithm to further improve the result. The influence of disturbance variables on the algorithm could be reduced by an automated exposure time algorithm and an evaluation score system. The needed high positioning accuracy of < 5 µm can be met with the setup presented here. A cost-efficient large-area laser system has been developed avoiding additional cost intensive technology to increase the accuracy of the robot.

Acknowledgements

The research project was carried out in the framework of the Industrial Collective Research Programme (IGF no. 21046 N) of the Research Association for Programming Languages for Production Facilities (FVP). It was supported by the Federal Ministry for Economic Affairs and Climate Action (BMWK) through the AiF. Part of the work has as well been supported by the State NRW through the Ministry of Economic Affairs, Innovation, Digitalization and Energy within the “5G-COMET” Project (005-2008-0093).

References

[1] Bliedtner, J., Müller, H., Barz, A., 2013. *Lasermaterialbearbeitung: Grundlagen - Verfahren - Anwendungen - Beispiele*. Fachbuchverl. Leipzig im Carl-Hanser-Verl., München.

[2] Haight, R., Carr, A.V., 2018. *Industrial Applications of Ultrafast Lasers*. WORLD SCIENTIFIC.

[3] Graaf, M. de, Aarts, R., 2013. Applications of robotics in laser welding, in *Handbook of Laser Welding Technologies*, Elsevier, p. 401.

[4] ABB. Absolute Accuracy Industrial robot option: Robotics. https://library.e.abb.com/public/931fcb281dbe7fecc1257b1300579a6a/AbsAccPR10072EN_R5.pdf. Accessed 15 December 2021.

[5] Fernandez, S.R., Olabi, A., Gibaru, O. On-line Accurate 3D Positioning Solution for Robotic Large-Scale Assembly Using a Vision System and a 6Dof Tracking Unit, in *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, p. 682.

[6] Zitová, B., Flusser, J., 2003. Image registration methods: a survey *21*, p. 977.

[7] Bradski, G., 2000. The OpenCV Library.

[8] Leonardis, A., Bischof, H., Pinz, A., 2006. *Computer Vision – ECCV 2006*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[9] Burt, P., Adelson, E., 1983. The Laplacian Pyramid as a Compact Image Code *31*, p. 532.

[10] Poljak, B.T., 1987. *Introduction to optimization*. Optimization Software, New York.