



Generalizable process monitoring for FFF 3D printing with machine vision

Kim Torben Werkle¹ · Caroline Trage¹ · Jan Wolf¹ · Hans-Christian Möhring¹

Received: 6 September 2023 / Accepted: 16 October 2023 / Published online: 18 November 2023
© The Author(s) 2023

Abstract

Additive manufacturing has experienced a surge in popularity in both commercial and private sectors over the past decade due to the growing demand for affordable and highly customized products, which are often in direct opposition to the requirements of traditional subtractive manufacturing. Fused Filament Fabrication (FFF) has emerged as the most widely-used additive manufacturing technology, despite challenges associated with achieving contour accuracy. To address this issue, the authors have developed a novel camera-based process monitoring method that enables the detection of errors in the printing process through a layer-by-layer comparison of the actual contour and the target contour obtained via G-Code processing. This method is generalizable and can be applied to different printer models with minimal hardware adjustments using off-the-shelf components. The authors have demonstrated the effectiveness of this method in automatically detecting both coarse and small contour deviations in 3D-printed parts.

Keywords 3D-printing · Fused filament fabrication (FFF) · Contour error detection

1 Introduction

Additive manufacturing methods differ from traditional subtractive manufacturing methods such as milling, lathe, and drilling, in which a desired part is cut out of a solid block of material using expensive machinery, human resources, and significant knowledge. Additive manufacturing refers to a group of manufacturing methods in which the desired part is created through a layer-wise deposition of material [1]. In additive manufacturing, three-dimensional structures are directly built from a computer-aided design (CAD) model [2], which significantly reduces the complexity of manufacturing. Additive manufacturing is also known as rapid prototyping [3] or rapid manufacturing [4] and is classified into seven main categories [5], including binder jetting, direct energy deposition, material jetting, powder bed fusion, sheet lamination, vat photopolymerization, and material extrusion. The latter includes Fused Filament Fabrication (FFF), in which solid material is liquefied inside a printing head and extruded through a nozzle while still in a viscous state and

ultimately bonds with adjacent material [1]. A layer-wise deposition of material can be achieved by relative movement between the printing bed and the printing head and sufficiently exact positioning in the x–y sphere. After one layer is completed, either the part is lowered or the print head is lifted to apply the next layer. Despite several issues that affect the achievable contour accuracy, FFF is the most widespread additive manufacturing technology for a diverse group of users due to its comparatively small investment costs [6]. The most common quality issue in FFF is warping [7]. As pointed out by [8], printing errors can be classified as detachment, missing material flow, deformed object, surface errors, and deviation from the model. Causes of detachment, such as cracking, warp deformation, and bed adhesion issues, have not been entirely solved yet [9]. Missing material flow can result from running out of filament, a snapped or stripped filament mid-print, a clogged nozzle, an overheated extruder motor driver, and/or a file or CAD file or G-code error [10]. Deformed areas can occur in areas with an overhang due to gravity [11]. Surface errors are caused by restrictions in layer thickness and the staircase effect studied in [12]. Detecting the last error class, deviation from the model, is challenging because a direct comparison between the model and the actual printed part is needed. Possible causes include shrinkage [13] or warping [14]. Additionally,

✉ Jan Wolf
jan.wolf@ifw.uni-stuttgart.de

¹ Institute for Machine Tools, University of Stuttgart,
Holzgartenstr. 17, 70174 Stuttgart, Germany

step losses in stepper motors and vibration can influence the accuracy of the printing process.

The errors encountered in Fused Filament Fabrication (FFF) result in workpieces that fail to meet customer demands, hindering its widespread adoption across all domains. To overcome this challenge, there is a need for systems that can detect faulty processes and actively control printing parameters to achieve a more robust process. Previous studies have proposed various methods for error detection, such as monitoring the FFF process via acoustic emission [15], real-time condition monitoring using thermal sensors and accelerometers [16], designing a filament advance detection sensor [17], and closed-loop control in combination with a linear encoder to improve accuracy [18].

However, it should be noted that each error category potentially has multiple causes, with slight hardware-related variations in each printer model [8]. Therefore, controlling all variables influencing the printing process using sensors is a challenging task since multiple parameters need to be controlled simultaneously. Furthermore, custom error detection and control mechanisms cannot be easily transferred to a different 3D printer, hindering progress towards becoming state-of-the-art solutions for process monitoring.

To overcome these limitations, researchers have focused on camera-based process monitoring in the context of 3D printing [9]. Previous studies have used attention networks with a control loop for printing parameter correction, a real-time monitoring platform capable of detecting various errors using single and double camera systems [19], a visual quality control system comparing the actual geometry of the workpiece to the theoretical point cloud reconstructed from G-Code [20], and machine learning approaches such as Convolutional Neural Networks (CNNs) [21] and Support Vector Machines (SVMs) [22] to classify images of the printing process and detect spaghetti-shape errors [23].

A framework for in-situ monitoring of objects printed with FFF by utilizing a thermal camera was presented [24]. Then the pixels of the thermal image are projected onto visible points. Although the authors describe the potential of their system for error detection, no such capabilities were implemented and thus showing the need for further research in defect detection. A dataset for classification of each printed layer in under printing, over printing, normal and empty regions by processing images taken with a high-speed 2D Laser profiler was published [25]. However, the authors

can only detect errors greater than half the layer height. When considering the financial expenses this method is not cost-efficient and thus cannot be implemented as a standard in every 3D printer. The introduction of 3D cameras raised interest amongst 3D printing researchers. The quality assessment of 3D printed parts with a 3D camera was studied [26]. The results show that dimensional inaccuracy can be detected. However, in addition to a costly 3D camera a turning build plate is needed to assess the printed parts from all sides which currently is not state of the art of commercially available 3D printers and thus it can be concluded that the method is not generalizable.

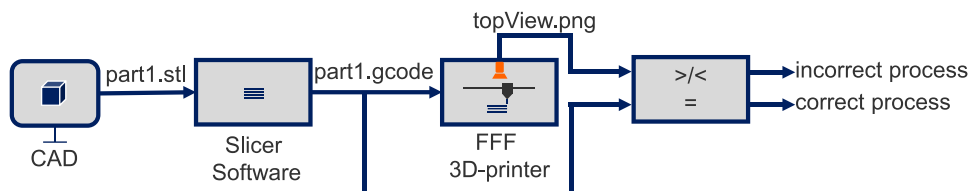
Computer vision combined with a ResNet50 convolutional neural network was applied to autonomously control the printing settings to correct defects [27]. However, this method does not support the analysis of the printing process. Although promising results were published downsides still exist when it comes to implementing machine learning on 3D-printers. Besides the need for powerful hardware and the acquisition of a database which represents the operating conditions, knowledge in the field of machine learning is not easy to obtain for hobby users and thus the correct implementation of algorithms is questionable and therefore limits the impact of these methods which results in the demand for easily implementable algorithms and low-cost solutions.

Despite significant progress, there is currently no generalizable and low-cost method to detect contour errors in each layer of the printing process. To address this, the authors present a new camera-based method that is generalizable and transferable to different printer models with minimal hardware adjustments, offering the potential to detect errors in printing with no regard to the actual 3D printer in use.

2 Materials and methods

The proposed method is based on a Canny edge detector which is applied to top view pictures of the workpiece to detect the actual contour of the topmost layer. Simultaneously the target contour of the workpiece gets extracted from the G-Code. This way a comparison of actual and target contour can be drawn enabling a novel method for process control. The proposed workflow to detect contour deviation is summarized in Fig. 1 and is described in detail in the following manner. First the hardware setup is presented.

Fig. 1 Proposed workflow for process monitoring



Secondly the algorithm for collecting the target and the actual contour is described. Finally, the applied metric for contour error detection is explained.

2.1 Hardware setup and camera calibration

The 3D printer utilized in this study is a self-built printer operating with RepRap firmware, as illustrated in Fig. 2. The machine vision system implemented in this work, designed for cost-effectiveness and generalizability in Fused Filament Fabrication (FFF), consists of a single Raspberry Pi 4 Model B with 4 GB RAM, equipped with a Raspberry Pi High Quality Camera featuring a 6 mm wide angle lens. A direct control is established between the Raspberry Pi 4 and the printer control board Duet2. The G-Code is generated through employment of the open-source Cura slicer software.

Mounting the Raspberry Pi HQ Camera on top of the printer enables a bird’s-eye view of the printing bed. However, without proper calibration, positional information derived from the images can be inaccurate due to lens-induced image distortion. Therefore, this approach is not suitable for precise contour detection (<0.1 mm).

In this study the authors applied the calibration method presented by [28] where a planar pattern is used to define the intrinsic camera parameters which make up the camera intrinsic matrix K where c_x and c_y describe the coordinates of the principal point and f_x and f_y describe the focal length:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

For the calibration a 9×6 checkerboard is used. In addition, the extrinsic camera parameters need to be determined. Extrinsic parameters are rotation matrix $R \in R^{3 \times 3}$ and translation vector $t \in R^{3 \times 1}$ which can be derived from Chasles’s theorem which states that every possible movement of a body can be composed of translation and rotation. The general

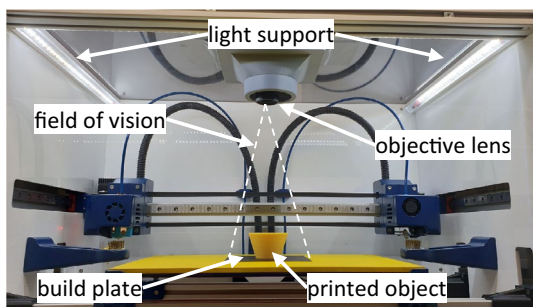


Fig. 2 3D-printer with camera mounted on top

rotation matrix R with roll, pitch and yaw angles α, β and γ is written as:

$$R = R_z(\gamma)R_y(\beta)R_x(\alpha)$$

$$= \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$= \begin{bmatrix} \cos\beta\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \cos\beta\sin\gamma & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma \\ -\sin\beta & \sin\alpha\cos\beta & \cos\alpha\cos\beta \end{bmatrix}$$

To find the translation vector t and rotation matrix R a reference object with known size of 100 mm × 100 mm is used. The origin of the world coordinate system is set to the lower left corner of the reference object. Each part of this study is printed on top of this reference object. To find the translation vector ${}^c t_w$ and rotation matrix ${}^c R_w$ of the reference object from camera to world coordinate system which are further abbreviated as c and w respectively, the following steps were taken:

1. Taking images with the mounted camera in the printer and undistorting them.
2. Detection of the coordinates of edges of a reference object with known size.
3. Solving the perspective-n-Points-problem and thus calculating the transformation matrix from world to camera coordinates which exploits homogenous coordinates and therefore results in a 4×4 matrix ${}^c T_w$:

$${}^c T_w = \begin{bmatrix} {}^c R_w & {}^c t_w \\ \mathbf{o}^T & 1 \end{bmatrix}$$

With known values of matrices K for intrinsic camera parameters and ${}^c T_w$ for extrinsic camera parameters the transformation from the target contour of the printed part according to the G-code to camera coordinates c_p can be calculated as:

$$c_p = K * {}^c T_w * \begin{pmatrix} GCode(x) \\ GCode(y) \\ 0 \\ 1 \end{pmatrix}$$

In this formula the authors made use of the fact that due to lowering the printing bed after each layer the distance from the top of the printed part to the camera stays constant. This way no influence of z coordinates occurs. To guarantee a correctly executed camera calibration a validation is needed which according to Fig. 3 delivered good results.

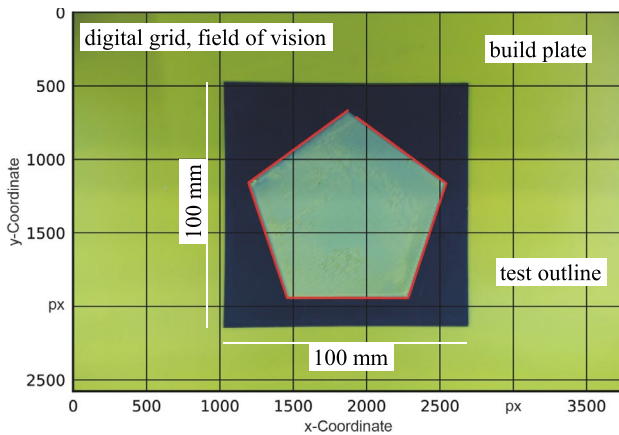


Fig. 3 Validation of the camera calibration and undistortion process

2.2 Proposed algorithm

The proposed algorithm allows a layer-wise comparison of the actual contour and the target contour received by G-Code processing. To ensure precise contour detection, a final post-processing step regarding the 3D printer control is required. This involves moving the printing head into a pre-defined position on the edge of the printing bed after each printed layer to avoid obstruction of the printed part during image capture. The Raspberry Pi Model 4, responsible for taking pictures, communicates with the Duet2 mainboard controlling the 3D printer through General-Purpose Input/Output

(GPIO) pins on the duex5 extension board in combination with the M42 P3 S1 command in the G-code. The undistorted image is converted from RGB to a greyscale image, on which a Gauß-filter in combination with a median filter of kernel size $H=5$ is applied to reduce noise. A thresholded binary image is obtained, which is used to detect the contour by a canny edge detector. To prevent unwanted edges from being included in the error detection process, a mask is applied. The `cv2.findContours()` function in OpenCV [29] is applied to the image, which saves all coordinates of the image that lie on an edge of the workpiece and thus serves as a generator for the actual contour. Information in the form of G-code needs to be processed to receive the target contour, which is split into wall-outer, wall-inner, skin, and fill for printing. Since contour errors are best indicated by a deviation in wall-outer, only printing head movements belonging to this class are examined, reducing the algorithm's complexity. Only G1 commands are considered, while G0 commands are ignored, as only the former commands extrude material for building the contour. Since the number of hardcoded points in G-code is limited, it is insufficient for accurate contour error detection. Therefore, additional target contour values are generated by a script according to Fig. 4.

The script subdivides the linear interpolation commands in sufficiently small intervals which allow further contour points to be artificially generated. An arbitrary example of this process is presented: Let there be a starting position of $P_0(x_0/y_0)$ and the successive line in the G-Code demand a linear movement to Position $P_1(x_1/y_1)$. The values of these

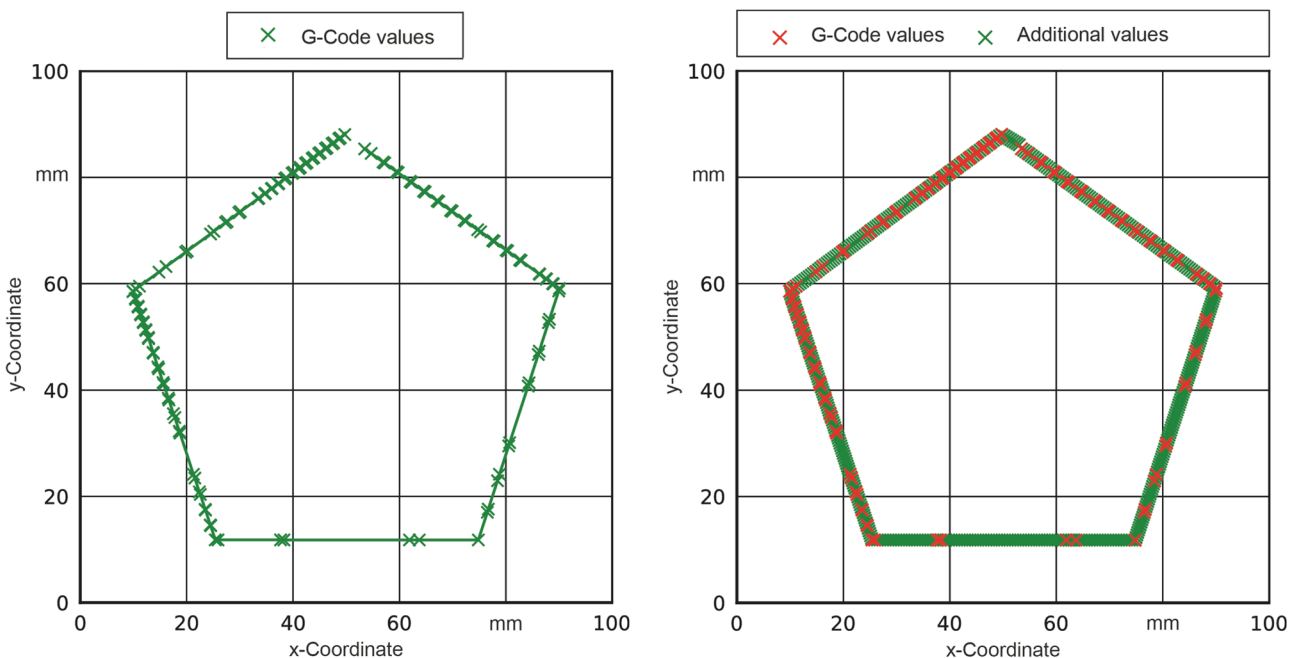


Fig. 4 Target contour values in G-code (left) and additional values (right)

coordinates is extracted via a text processing program which exploits the syntax of the G-Code When defining a set L which contains all additional points between P_0 and P_1 with the interval of s in mm the following expression is valid:

$$L = \left\{ \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \end{pmatrix} \frac{s}{\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}} t \right\}$$

$$\forall t \in \mathbb{N} | 0 < t < \frac{\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}}{s}$$

In case the distance of the two points divided by the interval s is no odd number, the value of s is decreased to satisfy this condition, which therefore guarantees equal spacing between P_0 and P_1 . During experiments this study found that intervals with a size of 0.5 mm promise good error detection results. Finally, after receiving the actual and the target contour of the workpiece a comparison can be drawn and decisions for error detection based on the following pre-defined metric can be made to classify the printing process as faulty or correct.

2.3 Error detection

The above-mentioned algorithm for comparing the actual and the target contour of a 3D-printed part enables a computer vision-based detection of two types of errors namely:

- Detection of over extrusion or detection of material in an unwanted position.
- Partial detachment of extruded material or missing material.

Both types of errors only focus on the contour of the workpiece in each printed layer. The first case is detected by a point-wise determination of the distance of each point of the actual contour C_i received by camera processing to the closest point of the target contour G_m received by G-code processing. If this distance d is exceeding a pre-defined maximum distance d_{max} an error is detected. The distance d is calculated as:

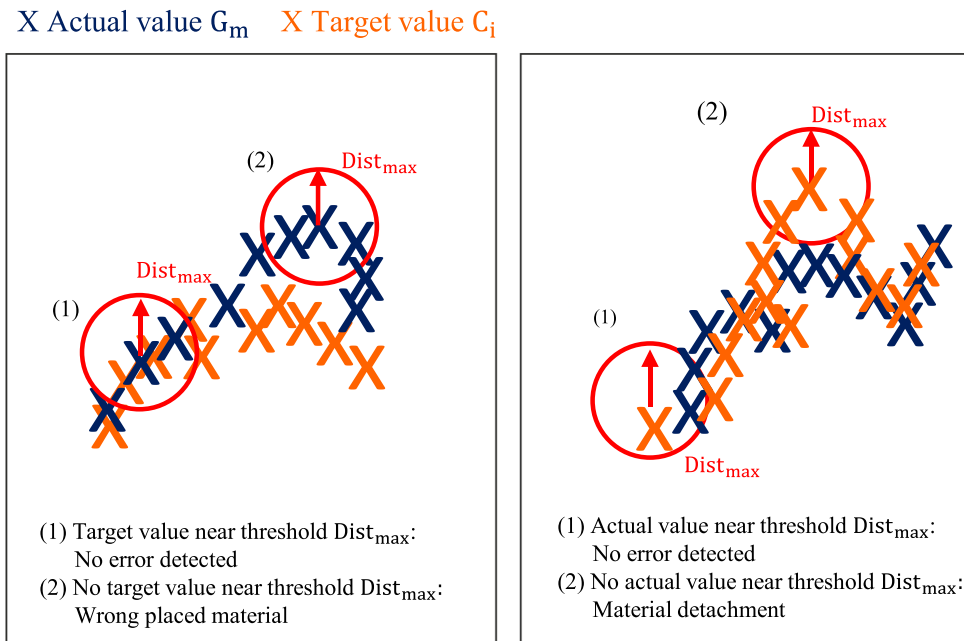
$$d = \sqrt{(C_i(x) - G_m(x))^2 + (C_i(y) - G_m(y))^2}$$

Wrongly placed material which might happen during over-extrusion then gets detected if no point in the target contour G_m can be found such that $d < d_{max}$. This approach is visualized in Fig. 5 (left). The second error type containing partial detachment is contrary to the first error type. In this approach an error is detected if no target contour point is within a pre-defined distance of any actual contour point. This approach is visualized in Fig. 5 (right).

3 Results

For validation of the proposed generalizable computer vision-based process monitoring method for contour error detection two workpieces with artificially generated contour errors are prepared. In a first experiment the authors

Fig. 5 Approach for detecting wrongly placed material (left) and material detachment (right)



deliberately placed the part to print in such a way that a missing support structure would lead to a collapsed part with coarse contour errors. The target geometry of this can be seen in Fig. 6.

As expected, the missing support structure leads to a large deviation of target contour and actual contour. However, the proposed algorithm is able to detect the faulty process as it is shown in Fig. 7. Contour errors which result from wrongfully placed material were marked red and contour errors which occurred near the target contour were marked in pink indicating material detachment.

After validating the proposed algorithm on substantial contour errors, additional experiments for validation on smaller errors are undertaken. For this a second workpiece with dimensions 15 mm × 15 mm is designed as depicted in Fig. 8.

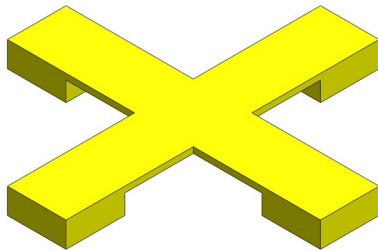


Fig. 6 Target geometry of the workpiece without support structures

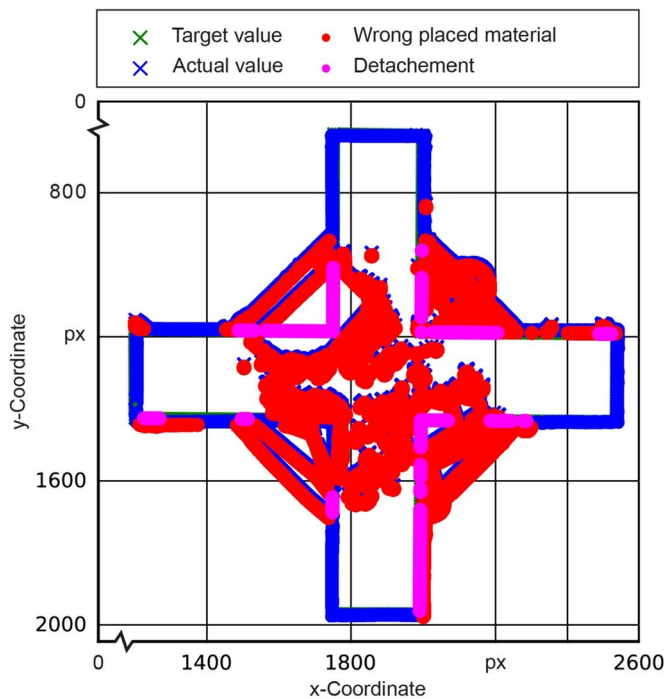


Fig. 7 Detected coarse errors with the proposed algorithm (left) and captured image (right)

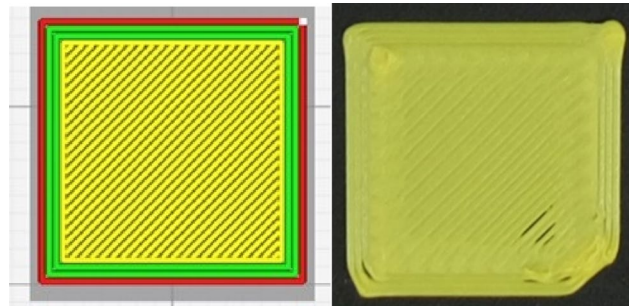


Fig. 8 Target contour (left) and actual contour of the workpiece with contour errors (right)

The criteria for selecting a second workpiece were set to allow the detection of contour errors resulting from normal 3D printing conditions, which means that the error is not introduced by the user like in the first workpiece. Both workpieces fully represent the errors detectable with just one single camera as a birds-eye view.

In Fig. 8 (left) the printed part is depicted and the contour differences compared to Fig. 8 (right) are clearly visible. The proposed algorithm is found to be able to reliably detect smaller errors in the workpiece contour. The results can be seen in Fig. 9 where the two defined error types were detected in the corners of the workpiece which is a fitting result when considering Fig. 8. Again both error types were

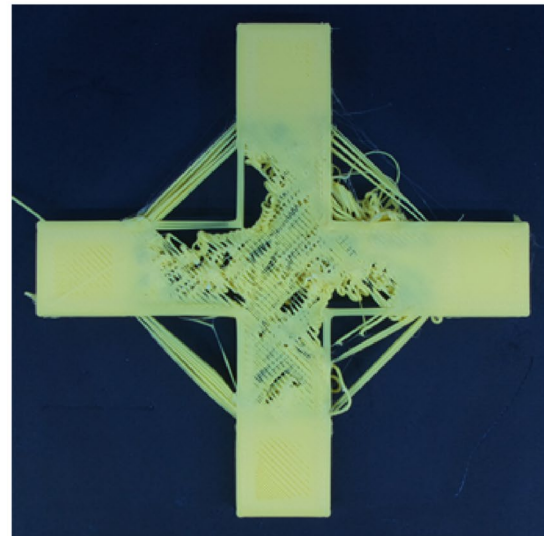
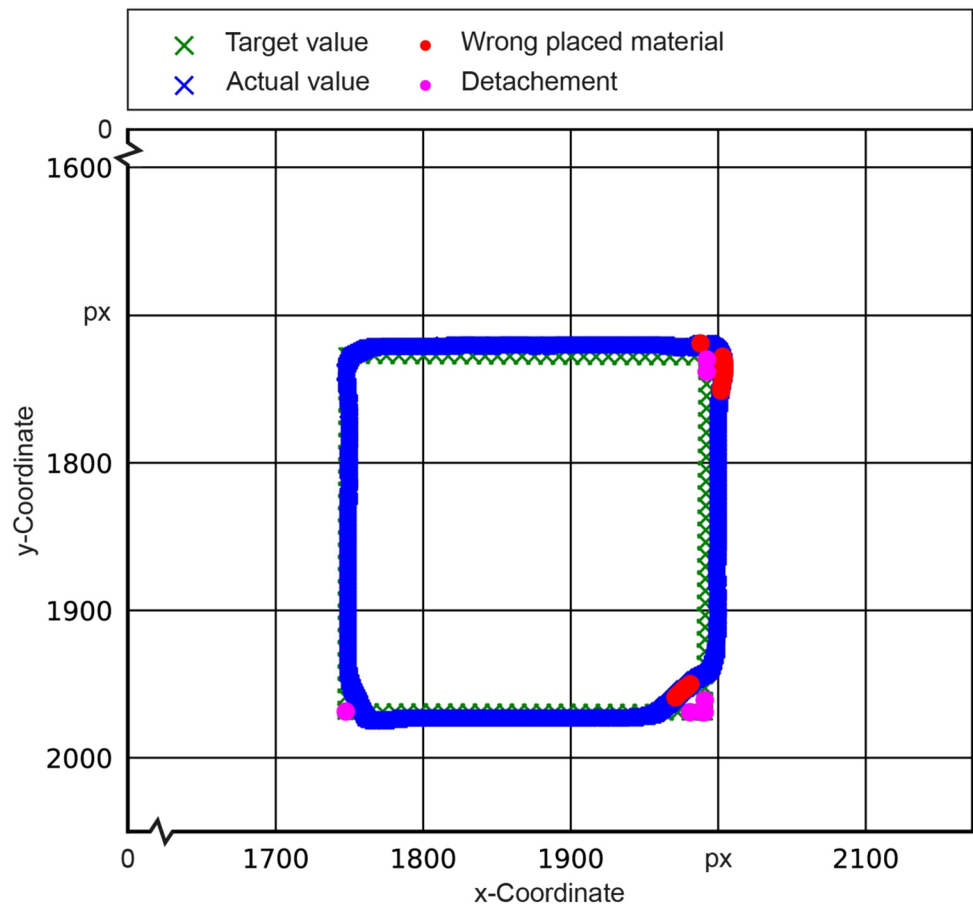


Fig. 9 Detected contour errors with the proposed algorithm



successfully detected which proves the functionality of the proposed method for coarse and small errors.

As a result of the above mentioned experiments it is obvious that the canny edge detector in combination with budget friendly hardware is able to detect two contour error types with a resolution of up to 0.1 mm. The limiting factor for a further improved error detector is the cameras resolution.

4 Discussion

Camera based process monitoring in 3D printing has widely been studied with a focus on machine learning. However, these methods require advanced knowledge of artificial intelligence algorithms and a representative dataset. Despite promising results, the transferability of these findings to different 3D-printers requires domain knowledge and powerful hardware for training and thus is unpractical for the end user.

To close this gap between research and end users a cost-effective method is presented which only requires a Raspberry Pi and the corresponding Raspberry Pi High Quality Camera for a combined €75. The following method was found to be a successful process monitoring technique for FFF printed parts. A post-processing step is applied to the

G-Code generated by the slicer to extend the target contour coordinates using linear interpolation. Additionally, a communication pipeline is established between the Duet2 mainboard and a Raspberry Pi through GPIO pins controlled by G-Code, enabling optimal timing for layer-wise picture-taking and control without interference from the printing head. The layer-wise image of the workpiece is then subjected to image processing and contour detection via a canny edge detector. The resulting actual contour is compared to the target contour derived from G-Code processing. A novel metric is applied to the resulting point clouds for error detection, and the algorithm makes a decision regarding the printing process state. The proposed method is capable of detecting two types of printing errors: over-extrusion and material detachment. The experimental results show that the method is capable of detecting coarse as well as small contour errors with a resolution of 0.1 mm. Based on a financial point of view the proposed method has the potential to amortize the required investments since high quality filament is expensive. When considering productivity the contour error detector has a huge potential to eliminate printing time after an error occurred since an unsupervised 3D-printer will continue printing until the end of the G-Code is reached. The issue of wasting time on an already failed print is relevant

for commercial and hobby users. Thus, the proposed method will lead to a decrease in wasted filament which goes hand in hand with greener additive manufacturing as well as to an increased productivity since an unnecessary completion of a failed print will be eliminated.

5 Conclusion

Fused filament fabrication has a huge potential to further revolutionize manufacturing and rapid prototyping. However, the process is prone to faulty process conditions, which are hard to control. Therefore, error detection methods need to be developed. Next to sensor networks cameras offer significant benefits as they are able to detect known as well as unknown errors. Besides many studies which focus on machine vision paired with machine learning, an easy to implement and budget friendly solution to detect contour errors has not been presented. That is why in this study a layer-wise process monitoring technique based on a canny edge detector is presented to detect contour errors during the printing process. Experiments show the effectiveness of the method to detect coarse and small contour errors and two types of errors namely over-extrusion and material detachment can be detected with a resolution of 0.1 mm. The method is generalizable and thus can be implemented on every FFF 3D-printer with minor adjustments. Thus, this method minimizes required engineering knowledge and financial resources for implementation and therefore offers users the potential to monitor the printing process regardless of their background. To further increase the effectiveness of the method new metrics can be developed and will be studied in a next step to identify additional errors. Furthermore, more machine vision algorithms can be compared to the canny edge detector to determine which one works best in detecting contour errors.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data availability The data that support the findings of this study are available upon reasonable request from the authors.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are

included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Gibson I, Rosen DW, Stucker B (2009) Additive manufacturing technologies: rapid prototyping to direct digital manufacturing. Springer, Berlin. <https://doi.org/10.1007/978-1-4419-1120-9>
- Tofail S, Koumoulos E, Bandyopadhyay A, Bose S, O'Donoghue L, Charitidis C (2017) Additive manufacturing: scientific and technological challenges, market uptake and opportunities. *Mater Today* 1:22–37. <https://doi.org/10.1016/j.mattod.2017.07.001>
- Kruth JP, Leu MC, Nakagawa T (1998) Progress in additive manufacturing and rapid prototyping. *CIRP Ann* 47(3):525–540. [https://doi.org/10.1016/S0007-8506\(07\)63240-5](https://doi.org/10.1016/S0007-8506(07)63240-5)
- Levy GN, Schindel R, Kruth JP (2003) Rapid manufacturing and rapid tooling with layer manufacturing (LM) technologies: state of the art and future perspectives. *CIRP Ann* 52(2):589–609. [https://doi.org/10.1016/S0007-8506\(07\)60206-6](https://doi.org/10.1016/S0007-8506(07)60206-6)
- ASTM F2792–12A (2012) Standard terminology for additive manufacturing technologies. ASTM International, West Conshohocken
- Shahrubudin N, Lee TC, Ralan R (2019) An overview on 3D printing technology: technological, materials, and applications. *Procedia Manuf* 35:1286–1296. <https://doi.org/10.1016/j.promfg.2019.06.089>
- Dave HK, Patel ST (2021) Introduction to fused deposition modeling based 3D printing process. In: Dave HK, Davim JP (Eds) fused deposition modeling based 3D printing. *Mater Form Mach Tribol*. Springer, Cham. https://doi.org/10.1007/978-3-030-68024-4_1
- Baumann F, Roller D (2016) Vision based error detection for 3D printing processes. *MATEC Web Conferences*. <https://doi.org/10.1051/mateconf/20165906003>
- Brion D, Pattinson S (2022) Generalisable 3D printing error detection and correction via multi-head neural networks. *Nat Commun*. <https://doi.org/10.1038/s41467-022-31985-y>
- Loh G, Pei E, Gonzalez-Gutierrez J, Monzón M (2020) An overview of material extrusion troubleshooting. *Appl Sci*. <https://doi.org/10.3390/app10144776>
- Cacace S, Cristiani E, Rocchi L (2017) A level set based method for fixing overhangs in 3D printing. *Appl Math Model* 44:446–455. <https://doi.org/10.1016/j.apm.2017.02.004>
- Chennakesava P, Narayan YS (2014) Fused deposition modeling - insights, international conference on advances in design and manufacturing
- Marvah O, Yahaya N, Darsani A, Mohamad E, Haq R, Johar M, Othman M (2019) Investigation for shrinkage deformation in the desktop 3D printer process by using DOE approach of the ABS materials. *J Phys*. <https://doi.org/10.1088/1742-6596/1150/1/012038>
- Ramian J, Ramian J, Dziob D (2021) Thermal deformations of thermoplast during 3D printing: warping in the case of ABS. *Materials*. <https://doi.org/10.3390/ma14227070>
- Wu H, Wang Y, Yu Z (2016) In situ monitoring of FDM machine condition via acoustic emission. *Int J Adv Manuf Technol* 84:1483–1495. <https://doi.org/10.1007/s00170-015-7809-4>
- Rao P, Liu J, Roberson D, Kong Z, Williams C (2015) Online real-time quality monitoring in additive manufacturing processes

- using heterogeneous sensors. *J Manuf Sci Eng* 137(6):1007–1019. <https://doi.org/10.1115/1.4029823>
17. Heras E, Haro F, de Augustin del Burgo J, Marcos M, D'Amato R (2018) filament advance detection sensor for fused deposition modelling 3D printers. *Sensors* 18:1495. <https://doi.org/10.3390/s18051495>
 18. Weiss B, Sorti D, Ganter M (2015) Low-cost closed-loop control of a 3D printer gantry. *Rapid Prototyp J*. <https://doi.org/10.1108/RPJ-09-2014-0108>
 19. Nuchitprasitchai S, Roggermann M, Pearce J (2017) Factors effecting real-time optical monitoring of fused filament 3D printing. *Prog Addit Manuf* 2:133–149. <https://doi.org/10.1007/s40964-017-0027-x>
 20. Paraskevoudis K, Karayannis P, Koumoulos E (2020) Real-time 3D printing remote defect detection (stringing) with computer vision and artificial intelligence. *Processes*. <https://doi.org/10.3390/pr8111464>
 21. Holzmond O, Li X (2017) In situ real time defect detection of 3D printed parts. *Addit Manuf* 17:135–142. <https://doi.org/10.1016/j.addma.2017.08.003>
 22. Delli U, Chang S (2018) Automated process monitoring in 3D printing using supervised machine learning. *Procedia Manuf* 26:865–870. <https://doi.org/10.1016/j.promfg.2018.07.111>
 23. Kim H, Lee H, Kim J, Ahn S (2020) Image-based failure detection for material extrusion process using s convolutional neuronal network. *Int J Adv Manuf Technol* 111:1291–1302. <https://doi.org/10.1007/s00170-020-06201-0>
 24. Binder L, Rackl S, Scholz M, Hartmann M (2023) Linking thermal images with 3D models for FFF printing. *Procedia Computer Science* 217:1168–1177. <https://doi.org/10.1016/j.procs.2022.12.315>
 25. Lyu J, Akhavan J, Manoochehri S (2022) Image-based dataset of artifact surfaces fabricated by additive manufacturing with applications in machine learning. *Data Brief* 41:107852. <https://doi.org/10.1016/j.dib.2022.107852>
 26. Li X, Zhang M, Zhou M, Wang J, Zhu W, Wu C, Zhang X (2023) Qualify assessment for extrusion-based additive manufacturing with 3D scan and machine learning. *J Manuf Processes* 90:274–285. <https://doi.org/10.1016/j.jmapro.2023.01.025>
 27. Jin Z, Zhang Z, Gu G (2019) Autonomous in-situ correction of fused deposition modeling printers using computer vision and deep learning. *Manuf Lett*. <https://doi.org/10.1016/j.mfglet.2019.09.005>
 28. Zhang Z (2000) A flexible new technique for camera calibration. *IEEE Trans Pattern Anal Mach Intell* 22:1330–1334. <https://doi.org/10.1109/34.888718>
 29. Bradski G (2000) The OpenCV Library. Dr. Dobb's J Software Tools

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.