Joint Special Interest Group meeting between euspen and ASPE
Dimensional Accuracy and Surface Finish in Additive Manufacturing,
KU Leuven, BE, October 2017
www.euspen.eu

Photogrammetry for Repositioning in Additive Manufacturing

J.N. Jensen[1], R. Lyngby[1], H. Aanæs[1], E.R. Eiríksson[1], J.B. Nielsen[1], J. Wilm[1], D.B. Pedersen[2]

[1] *Department of Applied Mathematics and Computer Science, Technical University of Denmark*
[2] *Department of Mechanical Engineering, Technical University of Denmark*

jnje@dtu.dk

Abstract

In this preliminary work, we present our current status on how to use single camera photogrammetry to determine the orientation of an additively manufactured partly finished object that has been repositioned in the printing chamber, from a single image taken with a calibrated camera, and comparing this to the CAD model of the object. We describe how this knowledge can be used to update the machine code of the printer such that printing of the object can be resumed in the new location. This opens possibilities for embedding and assembling foreign parts into the additive manufacturing pipeline, adding another layer of flexibility to the process. However, due to various error sources in estimating the orientation of the object, more work is needed before this update can be applied.

Repositioning, photogrammetry, computer vision, additive manufacturing.

1. Introduction

There are several scenarios within additive manufacturing, where it would be desirable to reposition the printed part mid-way through the printing process. This could e.g. be that one would like to assemble ‘foreign’ parts into the print, like electronics, bearings or fasteners, or that a repositioning of the additively manufactured part mid-way through the print process would dramatically decrease the need for support structures, e.g. if one is printing an object with large protrusion. In this work, we propose using single camera photogrammetry to achieve such a repositioning. This is achieved by taking an image of the partly printed object with a calibrated camera, and registering to a CAD model of the object in question. Hereby the new position and orientation of the partly printed object can be determined via standard computer vision techniques [1]. Following this, the printer’s machine code can be updated, such that it prints at the correct location with respect to the updated orientation of the printed object. In this work, we do not do this update, but merely show the feasibility of using single camera photogrammetry to find the required transformation.

A future perspective of this work is error correction; where an additive manufactured part with faults, can be cleaned up, re-placed in the print chamber, and printing resumed. This will, however, require a method for determining the geometry of the failed part. This could e.g. be done using a structured light scanner [2] or by passive stereo [3]. This is, however, beyond the scope of this work.

To the knowledge of the authors, repositioning objects in additive manufacturing has not previously been done. The use of photogrammetry in additive manufacturing is not new, however, it has primarily been used to digitalize objects which then can be replicated using additive manufacturing [4].

3. Method

In this section, we describe how the pose of a repositioned object is determined from a single image hereof. Our setup consists of a single camera placed approximately at a 45° angle looking down, with a full view of the build-plate of the printer. The printer used is based on fused deposition modelling.

2.1. Camera calibration

To estimate the pose of an object from an image, relative to the coordinate system of the printer, knowledge of the camera and its pose in relation to the printer’s coordinate system is needed. We model the camera with the pinhole camera model, where a world point $\boldsymbol{Q}$ in homogenous coordinates is projected to an image point $\boldsymbol{q}$ in homogenous coordinates by

$$\boldsymbol{q} = \boldsymbol{P}\boldsymbol{Q}, \tag{1}$$

where $\boldsymbol{P}$ is a 3x4-matrix given by

$$\boldsymbol{P} = \boldsymbol{A}\,[\boldsymbol{R} \quad \boldsymbol{t}]. \tag{2}$$

The 3x3-matrix $\boldsymbol{A}$ contains the intrinsic parameters of the camera and is given by

$$\boldsymbol{A} = \begin{bmatrix} f & \beta & \Delta x \\ 0 & \alpha f & \Delta y \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where f is the focal length, $(\Delta x, \Delta y)$ is the principal point, α is a scaling factor, and β is a shearing factor. These parameters are determined according to the method described in [5].

The pose of the camera relative to the coordinate system of the printer is described by the 3x3 rotation matrix $\boldsymbol{R}$ and the 3x1 translation vector $\boldsymbol{t}$. These are also known as the extrinsic parameters. This is necessary since we are interested in knowing the pose of the printed object in the printer's coordinate system, and not with the position of the camera as the origin. This is known as the Perspective-n-Point (PnP) problem. To solve it, a number of point-pairs for which we know their coordinates in the coordinate system of the printer and their corresponding position in an image are needed. These are achieved by printing four thin circle markers, such that they are visible to the camera. Successively, their positions in the image are found by first thresholding to their colour, followed by ellipse fitting [6]. The markers can be seen in Figure 1. Their coordinates in the printer's coordinate system are known from the printer machine code. The PnP problem is solved by the method in [7].

To determine camera parameters, intrinsic and extrinsic, the implementation of Bradski 2000 [8] is applied.

2.1. Pose estimation

Finding the pose of the printed object is very similar to that of finding the pose of the camera. We use correspondences of points on the object with known 3D coordinates and 2D image coordinates. The 3D coordinates used can be in an arbitrary coordinate system which we may define ourselves, and the aim is then to find the rotation $\boldsymbol{R}_p$ and translation $\boldsymbol{t}_p$ of the object transforms that yields its coordinates in the coordinate system of the printer. To get the point correspondences we choose a number of corners, as these are easily distinguishable, on the surface of the 3D model being printed, and manually annotate the position of these in the image. The transformation is then found by the method described in [9]. To increase the accuracy, we perform a bundle adjustment [1], i.e. we find a solution to

$$\underset{\boldsymbol{R}_p,\boldsymbol{t}_p}{\arg min} \textstyle\sum_i \|\boldsymbol{q}_i - \boldsymbol{P}\,[\boldsymbol{R}_p \quad \boldsymbol{t}_p]\boldsymbol{Q}_i\|^2, \tag{4}$$

where the first transformation is used as the initial guess.

With this transform, it is possible to update the machine code of unprinted parts of the object, before printing is resumed. This can be done by estimating the original pose, given by $\boldsymbol{R_0}$ and $\boldsymbol{t}_0$, and the new pose, given by $\boldsymbol{R_1}$ and $\boldsymbol{t}_1$. The relative transformation between the two poses can then be computed by

$$[\boldsymbol{R}_r \quad \boldsymbol{t}_r] = [\boldsymbol{R}_1 \quad \boldsymbol{t}_1]\begin{bmatrix} \boldsymbol{R}_o^T & -\boldsymbol{R}_o^T\boldsymbol{t}_o \\ 000 & 1 \end{bmatrix}, \tag{5}$$

i.e. first the inverse transformation of the original pose back to the origin of the coordinate system followed by the transformation to the second pose.

4. Results

We have printed a 10 x 10 x 30 mm cuboid, repositioned and estimated its pose using the method described. The results of two estimated poses are shown in Figure 1. Visually the red lines, which show the estimated poses, coincide very well with the green cuboid. Future derivatives of this work will include a more detailed analysis on the precision of this estimation.

In Figure 2, a virtual rod has been transformed, using the estimated pose of the cuboid, and superimposed on the corresponding image. This transformation could be applied to the printer's machine code, such that printing of the object could be resumed. However, due to inaccuracy in the order of millimetres, this update is not applied in this work.

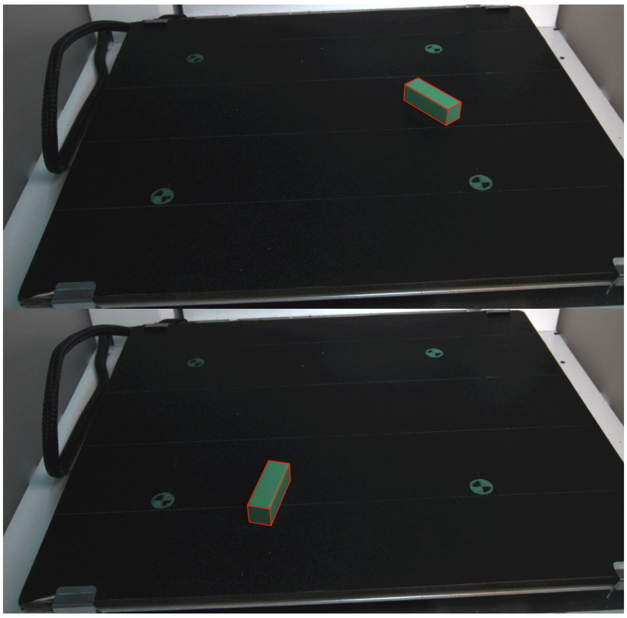

Figure 1. Examples of estimated poses. The red lines are overlaid and shows the estimated pose.

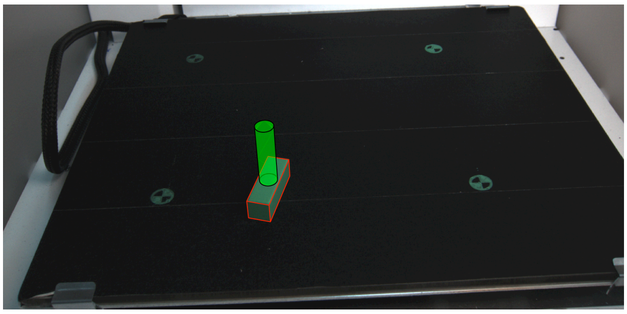

Figure 2. Using the estimated pose of an object to transform another object or part to the correct position.

4. Discussion and conclusion

We have shown how single camera photogrammetry can be used to determine the pose of a repositioned partly printed object in a 3D printer, by comparing an image of the object in the new position with a CAD model of it.

More work is needed, in order to use this pose estimate to resume printing of a paused object. This is due to the lack of accuracy in which the orientation can be determined, which is very important if one wants to successfully resume printing. The accuracy is influenced by various parameters, the most important being the precision of the camera calibration. To obtain a better calibration more than four markers should be used, to average out any errors there might be in determining the marker positions in the image.

Another important error source is the fact that a printed object does not resemble the corresponding CAD model completely, but naturally include errors due to the manufacturing process. The printer used is a low-cost consumer-grade printer, and a better printer could help lessen the effect of this error source.

In a lot of other applications, the same level of accuracy in the pose estimation is not needed, e.g. in robotics knowing the pose with millimeter accuracy is usually sufficient for a robot to be able to grip the object. A very high accuracy is however needed in order to successfully resume a paused 3D print.

In this paper, we have presented our preliminary findings on part repositioning using photogrammetry. The demonstrated setup contains low-cost components and has the potential to be used with any 3D printer where the camera can be placed such that the printing chamber is visible.

References

[1] Hartley R and Zisserman A, 2003, "Multiple View Geometry in Computer Vision", 2nd edition, Cambridge University Press, New York, NY, USA.

[2] Eiríksson ER, Wilm J, Pedersen DB and Aanæs H, 2015, "Precision and Accuracy Parameters in Structured Light 3-D Scanning", *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* **40**

[3] Aanæs H, Jensen RR, Vogiatzis GV, Tola E and Dahl AB, 2016, "Large-scale data for multiple-view stereopsis", *International Journal of Computer Vision.*

[4] Kaufman J, Rennie AEW and Clement M, 2015, "Single Camera Photogrammetry for Reverse Engineering and Fabrication of Ancient and Modern Artifacts", *Proc. CIRP* **36** 223-229

[5] Zhang Z, 2000, "A Flexible New Technique for Camera Calibration", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22(11)** 1330-1334.

[6] Fitzgibbon AW and Fisher RB, 1995, "A Buyer's Guide to Conic Fitting", *Proc. 5.th British Machine Vision Conference* 513-522

[7] Gao X-S, Hou X-R, Tang J and Cheng H-F, 2003, "Complete Solution Classification for the Perspective-Three-Point Problem", *IEEE Transactions on Patt*ern Analysis and Machine Intelligence **25(8)** 930-943.

[8] Bradski G, 2000, "The OpenCV Library", Dr. Dobb's Journal: Software for the Professional Programmer **25(11).**

[9] Ansar A and Daniilidis K, 2003, "Linear pose estimation from points or lines", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25(5)** 578-589.