Manufacturing Engineering Society International Conference 2017, MESIC 2017, 28-30 June 2017, Vigo (Pontevedra), Spain

Development of an error compensation case study for 3D printers

A.C. Majarena, J.J. Aguilar, J. Santolaria

Department of Design and Manufacturing Engineering, EINA. University of Zaragoza, Zaragoza 50018 , Spain. majarena@unizar.es

Abstract

The paper developed presents a case study that allows students to learn an easy way to improve the accuracy of low cost 3D printers. The document detailed a methodology to achieve this goal. First, it is necessary to print an initial CAD design. A commercial scanner is calibrated and the pieces are scanned to obtain the different errors. Then, a program is generated to compensate the code numerical control of the printer. This fact allows students to print a new piece having less errors than before, which it involves improve the printer accuracy.

© 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the Manufacturing Engineering Society International Conference 2017.

Keywords: Additive manufacturing, error compensation, 3D printer, flatbed scanner;

1. Introduction

There has been a rapid development in recent years of additive manufacturing (AM) techniques.

This technology builds 3D objects by adding layer-upon-layer of material (mainly plastic or metal) directly or from a digital file with the geometry of the 3D CAD [1].

Some of the advantages of AM are efficiency, considering speed, effort, process delays and staff time, mainly when complexity of the product is high, production volumes are low or changes are frequent and they do not need any tools like they do in conventional manufacturing processes. Moreover, additive manufacturing usually generates less waste than common subtractive techniques such as CNC/Milling machines. One of the disadvantages of AM is accuracy [2, 3].

There are different ways to measure the errors of a printer piece such as a coordinate measuring machine, and articulated arm coordinate measuring machine, a flatbed scanner, etc…

A flatbed scanner is a tool that everyone can obtain easily. There is specialized literature to improve their accuracy. In [4], a methodology is described which allows to obtained traceable measurements. The authors convert the reference

system of the flatbed scanner into a Cartesian reference system, thereafter calibrating it using standards with metrological traceability.

The objective of this work is to develop a case study to improve the accuracy of a low cost 3D printer. The case is design for students of the Master of Industrial Engineering for the Precision engineering and additive manufacturing subject. Fig. 1 shows some of the low cost 3D printers.
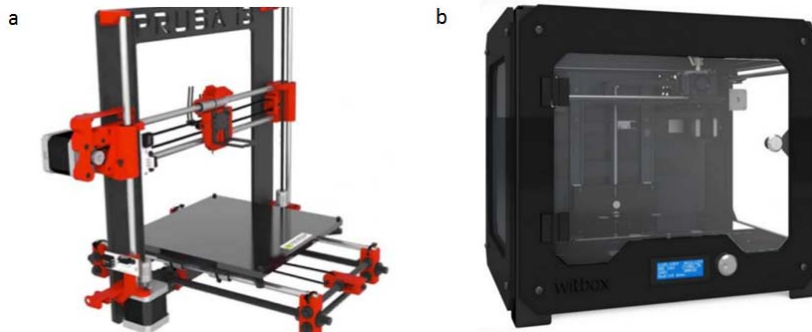

Fig. 1. (a) Prusa i3 Hephestos 3D printer; (b) BQ Witbox 3D printer.

2. Methodology

The first step is to design the piece of study in a CAD software such as Solid Edge [5], Solid Works [6], NX [7], etc.

A stl (Standard Triangle Language) file describes a raw unstructured triangulated surface by the unit normal and vertices (ordered by the right-hand rule) of the triangles using a three-dimensional Cartesian coordinate system.

The piece is saved in a stl format to be loaded in a printer software such as CURA. To design the pieces we thought in a piece that cover the printer workspace in X and Y directions and that the printer time wasn't too high. The piece of work is shown in Fig. 2.

The CURA program allows us to load the stl file and to set up the parameters for the printing process.

The different parameters such as layer height, shell thickness, speed printing temperature and support are defined here.

Fig. 3 shows the basic and advanced values given in this case. As it was mentioned before, the printer time was adjusted to be able to print the pieces in a class session, so we choose a normal quality to avoid that this time was very high with the best quality possible.
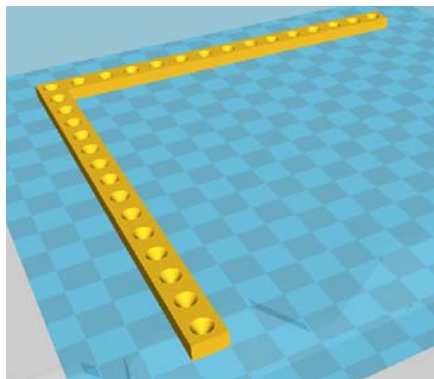

Fig. 2. Design of the piece.

| Basic | Advanced | Plugins | Start/End-GCode | | Basic | Advanced | Plugins | Start/End-GCode |

Quality

Layer height (mm)	0.2
Shell thickness (mm)	1.2
Enable retraction	✓	...

Fill

| Bottom/Top thickness (mm) | 0.6 |
| Fill Density (%) | 10 | ... |

Speed and Temperature

| Print speed (mm/s) | 50 |
| Printing temperature (C) | 220 |

Support

| Support type | None ▼ | ... |
| Platform adhesion type | Brim ▼ | ... |

Filament

| Diameter (mm) | 1.75 |
| Flow (%) | 100.0 |

Machine

| Nozzle size (mm) | 0.4 |

Retraction

| Speed (mm/s) | 50 |
| Distance (mm) | 5 |

Quality

Initial layer thickness (mm)	0.3
Initial layer line width (%)	100
Cut off object bottom (mm)	0
Dual extrusion overlap (mm)	0.15

Speed

Travel speed (mm/s)	100
Bottom layer speed (mm/s)	30
Infill speed (mm/s)	0.0
Top/bottom speed (mm/s)	0.0
Outer shell speed (mm/s)	0.0
Inner shell speed (mm/s)	0.0

Cool

| Minimal layer time (sec) | 5 |
| Enable cooling fan | ✓ | ... |

Fig. 3. Basic and advanced parameters.

The time estimated by the Cura program for the piece was 1 hour and 28 minutes and the material needed was 13 grams (4.21 meters).

It is important to set up the platform adhesion type in the support. In this case we have selected the option "Brim". This option adds a single layer flat area around the base of the model to ensure the print doesn't detach from the build plate.

The code to control the printer is generated and loaded in the 3D printer. For the class sessions we have used a Prusa i3 Hephestos. Fig. 4 shows the piece printed.
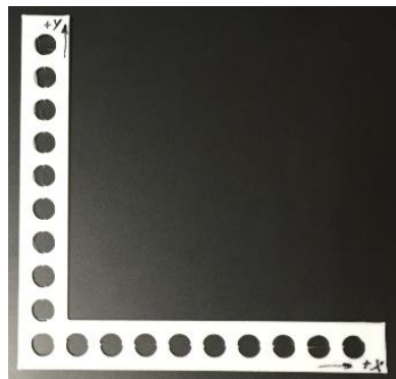
Fig. 4. Piece printed in a Prusa i3 Hephestos.

Once the piece is printed, students have measured and obtained the centers of the different circles using a scanner calibrated with a pattern.

The coordinate system (p, q) of the scanner is not a Cartesian system. Fig. 5 shows the (p, q) coordinates before and after adjusting.
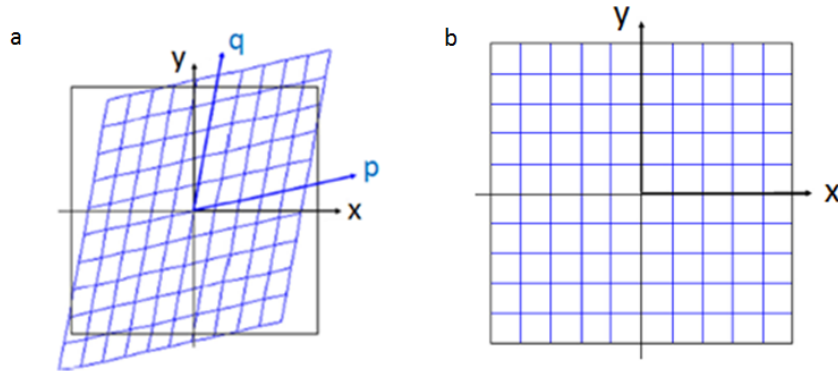

Fig. 5. (a) (p, q) Coordinates before adjusting; (b) (p, q) coordinates after adjusting.

For example, if the set of points of the scanner is represented such that p = cte (or q = cte), a curve is generally obtained and not a straight line (even if the curve can be much like a straight line). In addition, it could happen that:
- The curves $p = cte$ (or $q = cte$) were not exactly parallel.
- The curves $p = cte$ and $q = cte$ will not be cut completely perpendicular.
- The pixel dimensions according to the directions p and q were slightly different.

Because of this, if the distance between two points A and B of a part is measured, the result $\sqrt{\Delta p^2 + \Delta q^2}$ where $\Delta p = p_B-p_A$ and $\Delta q = q_B-q_A$) may vary by changing the orientation in which measure. However, such distance is constant if the piece is dimensionally stable. That is to say, the application that assigns the value to the distance between points A and B is not invariable in front of rotations or translations of the part with respect to the scanner. To overcome this problem, an application must be find (x, y) = f (p, q) that transforms the non-Cartesian coordinates (*p*, *q*) in coordinates (*x*, *y*) that are. That is, the distance is invariable in the face of rotations and/or translations of the part relative to the scanner. If you are linearizing the function f and in addition:
- We choose the origin of the system (*x*, *y*) coincident with the origin of the system (*p*, *q*)
- It is possible to choose the orientation of the system (*x*, *y*) so that:

$$\partial x/\partial q = \partial y/\partial p \tag{1}$$

- The scaling of the system (*x*, *y*) is chosen so that:

$$\partial x/\partial q + \partial y/\partial p = 1 \tag{2}$$

Then, the application (x, y) = f (p, q) can be written as [4, 8, 9, 10]:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} p \\ q \end{bmatrix} + \begin{bmatrix} a & \frac{1}{2}\theta \\ \frac{1}{2}\theta & -a \end{bmatrix} \cdot \begin{bmatrix} p \\ q \end{bmatrix} \tag{3}$$

where

$$\partial x/\partial q - \partial y/\partial p = 2a \tag{4}$$

and

$$\partial x / \partial q = \partial y / \partial p = \frac{\theta}{2} \tag{5}$$

The coefficient a represents the relative difference between the pixel dimensions according to the p and q axes of the scanner and the coefficient θ represents the perpendicularity defect between these axes.

According to the linear transformation $(x, y) = f(p, q)$ introduced above, the distance $l = \sqrt{x^2 + y^2}$ between two points could be written as follows:

$$l^2 = x^2 + y^2 = [x \ y] \cdot \begin{bmatrix} x \\ y \end{bmatrix} = [p \ q] \cdot (I + G)^T (I + G) \cdot \begin{bmatrix} p \\ q \end{bmatrix} \cong [p \ q] \cdot (I + 2G) \cdot \begin{bmatrix} p \\ q \end{bmatrix} \tag{6}$$

so,

$$l^2 = (1 + 2a)p^2 + 2\theta pq + (1 - 2a)q^2 \tag{7}$$

The parameters a and θ are supposed to be very close to zero, so the terms where the product of both appear has been considered negligible, as in the $\mathbf{G^T G}$ matrix product.

If a piece is measured by varying its orientation with respect to the scanner and this piece is dimensionally stable, the distance L_0 between two points of that part should always be the same. For this reason, equation (8) is a constant, $L_0{}^2$, for any orientation. If the orientation is such that $p = r \cos \alpha$ and $q = r \sin \alpha$, where $r = \sqrt{p^2 + q^2}$, $\cos \alpha = p/r$ and $\sin \alpha = q/r$, then:

$$l^2 = (1 + 2a)r^2 \cos^2 \alpha + 2\theta r^2 \sin \alpha \cos \alpha + (1 - 2a)r^2 \sin^2 \alpha = cte = L_0^2 \tag{8}$$

Simplifying the above expression and considering several orientations α_i would give the following equations:

$$r_i^2 + 2r_i^2 a(\cos^2 \alpha_i - \sin^2 \alpha_i) + 2\theta r_i^2 \sin 2\alpha_i \cong L_0^2 \quad for \ i=1, \ ...,n \tag{9}$$

In matrix form, the above equations could be rewritten as follows:

$$\begin{bmatrix} 1 - 2 r_1^2 \cos 2\alpha_1 & -r_1^2 \sin 2\alpha_1 \\ 1 - 2 r_2^2 \cos 2\alpha_2 & -r_2^2 \sin 2\alpha_2 \\ \vdots \\ 1 - 2 r_n^2 \cos 2\alpha_n & -r_n^2 \sin 2\alpha_n \end{bmatrix} \cdot \begin{bmatrix} L_0^2 \\ \alpha \\ \theta \end{bmatrix} \cong \begin{bmatrix} r_1^2 \\ r_2^2 \\ \vdots \\ r_n^2 \end{bmatrix} \tag{10}$$

If this system is solved by an ordinary least squares fit, the estimation of the parameter vector

$$y = \begin{bmatrix} L_0^2 \\ \alpha \\ \theta \end{bmatrix} \tag{11}$$

would be as shown by Equation (12). This equation has the form $A \cdot y \cong b$

$$y = (A^T A)^{-1} A^T \cdot b \tag{12}$$

In order to improve the estimation of the coefficients a and θ, instead of working with a single distance between two points, we work with the mean distance between circles of the 2D pattern by rows, columns or diagonals of Fig. 6. The pattern is formed by 76 rows and columns of circles separated from each other by a nominal distance of 2 mm. This pattern has been placed parallel to the scanner's p and q axes (in the positions needed to cover the scanner's

measurement range). It is proposed to work with all available rows and columns and with several diagonals (at least 2 of them per position).
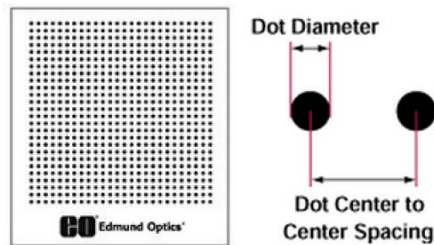

Fig. 6. 2D pattern grid.

Position and straightness errors are determined as can be seen in Fig. 7. As it can be noticed, errors increase as the print head move away from the origin. Position errors are positive in X axis while are negative in Y axis. Maximum errors are greater in Y axis than in X axis with a value of -1.5 mm.

The maximum straightness error in Y when the longitudinal motion is in X is -0.07 mm.

The maximum straightness error at X when the longitudinal motion is in Y is 0.22 mm. In this case, we can observe greater variability of the data and a greater error range than the previous case.
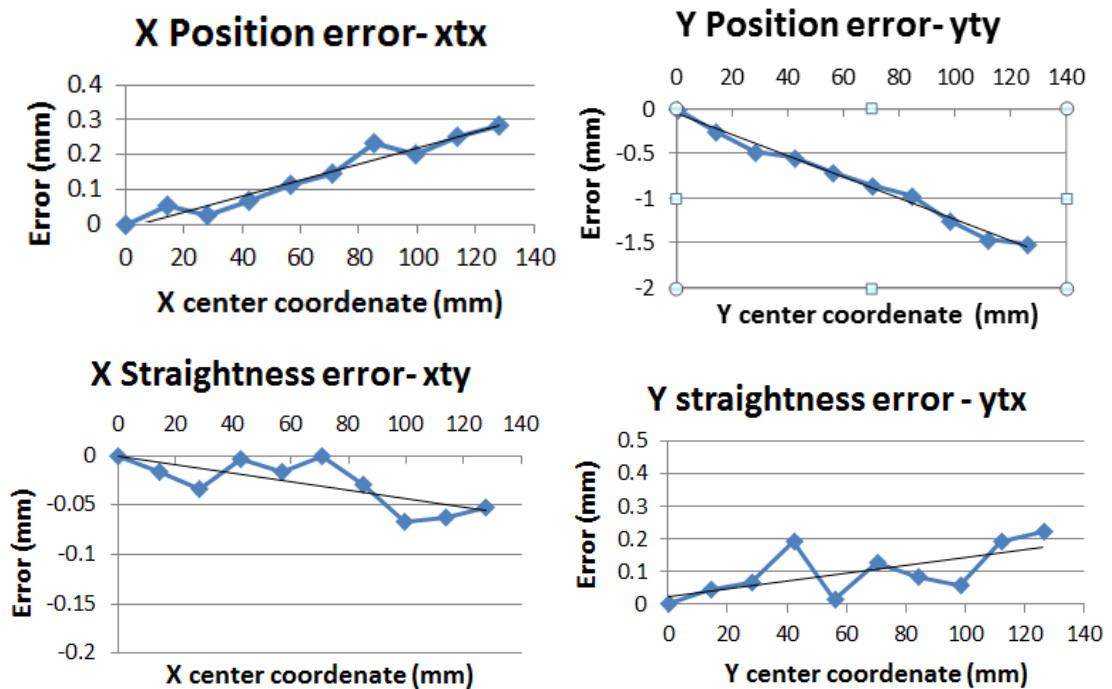

Fig. 7. Position and straightness errors.

The following step was to compensate the 3D printer trajectory. To do this, the numerical control code is compensated in Matlab. First, the Matlab program read the numerical control code and the error matrices with position and straightness. Then, it corrects the numerical program by reading each numerical control line and applying the compensation needed in function of the different coordinates X and Y. Finally, the piece is printing with the compensated numerical control code. The new piece is then measured and new errors are analyzed.

3. Results obtained

Once the trajectory is corrected, and the new piece is printed and measured, errors are obtained. Fig. 8 shows the new errors once the code is compensated.
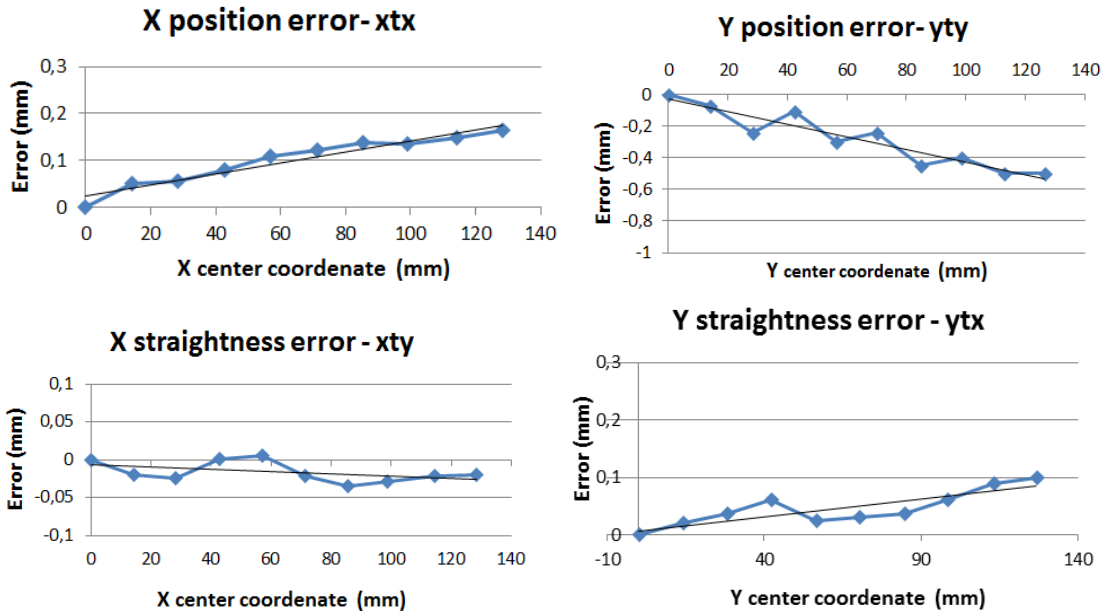

Fig. 8. Position and straightness errors for the piece printed after compensation.

If we analyzed the X and Y position error, we can observe that the error has been considerably reduced, obtaining maximum errors of 0.164 mm in X position error and -0.5 mm in Y position error.

The maximum straightness error in Y, when the longitudinal movement is in X, is -0.035 mm, thus decreasing considerably the errors obtained for the initial piece. In addition, the maximum straightness error in X, when the longitudinal movement is in Y, is 0.1 mm approximately.

4. Conclusions

The case study allows students to learn an easy way to improve the accuracy of low cost 3D printers. By printing an initial CAD design, measuring the piece, and obtaining the error, students can generate a program to compensate the computer numerical control code of the printer. Moreover, students calibrate a flatbed scanner converting a commercial scanner into a dimensional coordinate instrument of two coordinates.

This fact allows them to print a new piece having less errors than before, which it involves improve the printer accuracy.

References

[1] Additive Manufacturing. Available at: http://additivemanufacturing.com/basics/ (2016 November).
[2] C.K. Chua, S.M. Chou, T.S. Wong, Int. J. Adv. Manuf. Tech. 14 (1998) 146-152.
[3] P. Shah, R. Racasan, P. Bills, Case Studies in Nondestructive Testing and Evaluation, In Press.
[4] J. de Vicente, A.M. Sánchez-Pérez, P. Maresca, J. Caja, E. Gómez, Measur. 73 (2015) 304-312.
[5] Automation Siemens. Available at: https://www.plm.automation.siemens.com/es_es/products/solid-edge/st9/index.shtml (2017 May).
[6] Solidworks. Available at: http://www.solidworks.es/sw/products/solidworks-products-adwords.htm?mktid=7773&gclid=CL2G4t6X4NMCFUG4Gwod_bkGqw (2017 May).
[7] Automation Siemens. Available at: https://www.plm.automation.siemens.com/es_es/products/nx/index.shtml (2017 May).

[8] JCGM 100:2008, JCGM, Available at: http://www.bipm.org/en/publications/guides/gum.html (2017 January).
[9] JCGM 101:2008, JCGM (2008), Available at: http://ww.bipm.org/en/publications/guides/gum.html (2015 October 04).
[10] JCGM200:2008. JCGM, Available at: http://www.bipm.org/en/publications/guides/vim.html (2017 January).