2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)

PredNet and CompNet: Prediction and High-Precision Compensation of In-Plane Shape Deformation for Additive Manufacturing

Zhen Shen[1], *Member, IEEE,* Xiuqin Shang[2], *Member, IEEE,* Yuqing Li[3], Yin Bao[4], Xipeng Zhang[5], Xisong Dong[6], *Member, IEEE,* Li Wan[7], Gang Xiong[8], *Senior Member, IEEE,* and Fei-Yue Wang[9], *Fellow, IEEE, ASME*

***Abstract*— The error compensation for printed objects in additive manufacturing (AM) has always been one of the most critical problems. The precision control of the AM is usually more difficult than the subtractive manufacturing system, whose precision can reach the micron level easily by using a servo system. For the AM, there usually exist shrinkage and curling effects which lead to deformation. In this paper, we focus on the in-plane shape deformation problem, and we build the PredNet and CompNet, using deep neural networks for the error prediction and compensation. We test our methods on dental crown models. We generate deformed models by simulation of the translation, scaling down and rotation deformation. The minimum $F1$ scores of error prediction and compensation can be up to 0.982.**

I. Introduction

Additive manufacturing (AM), also known as 3D printing [1], refers to processes of creating a three-dimensional object in which layers of material are formed under computer control. After three decades of development, the AM has played an important role in many fields such as medicine, aerospace, and robots. However, error control in 3D printing is always a key and complicated problem in AM. There are seven main types of 3D printing processes [2]: material injection, binder injection, material extrusion, powder bed melting, directed energy deposition, sheet lamination and vat photopolymerization. The accuracy of subtractive manufacturing systems can achieve the micron scale easily by a servo system, whereas the positioning accuracy of commercial 3D printers is at the scale of tens or hundreds of microns, and the error of the printed object would be larger. Except sheet lamination, other processes usually make the material tighter in the printing process. There usually exist shrinkage and curling effects, and the deformation can be affected by the shape of the object, which make the error control of the AM difficult.

The general method of error compensation for 3D printing is to predict the error in the printing process, and then take corresponding measures to compensate it. Huang [3] proposed to use the finite element method (FEM) to predict the deformation in the printing process. However, the complex process of the AM causes difficulties for an accurate prediction. Not only the property of the material but also the building process should be considered for a successful prediction, while there is no accurate model for describing the process. Luan and Huang proposed to predict and control arbitrary in-plane freeform shape deformation by a learning method [4]. In our previous work, we proposed to use a deep neural network to predict and compensate the error, but we only focused on direct prediction and compensation of 3D models with low accuracy [5]. Usually it is not easy to measure 3D models. In this paper, we mainly focus on high-precision prediction and compensation of in-plane shape deformation. We can work on high precision 2D data. And it is much easier to obtain 2D data, which makes the method in this paper more applicable.

We still use dental crown models in our experiment. The models are similar but different. In this paper, we use simulation to generate deformed models, and build prediction and high-precision compensation deep neural networks, which are called "PredNet" and "CompNet" respectively. The **key contributions** of this paper are as follows,

- We have designed a deep neural network for prediction and high-precision error compensation of in-plane shape deformation in 3D printing.
- Compared with our previous work, we predict and compensate the in-plane shape deformation of the model based on much more accurate data. The method is more accurate and easier to use in practice.

*This work was supported in part by the National Key Research and Development Program (No. 2018YFB1700403), the National Natural Science Foundation of China under Grants 61773382, 61773381, and 61533019, Chinese Guangdong's S&T project 2017B090912001, 2016B090910001, 2015B010103001, the Beijing Natural Science Foundation under Grant 4182065, Dongguan's Innovation Talents Project (Gang Xiong), and the 2018-2019 joint project between Beijing Ten Dimensions Technology Co., Ltd. and Institute of Automation, Chinese Academy of Sciences. (*Corresponding author: Gang Xiong.*)

[1]Z. Shen, [2]X. Shang, [8]G. Xiong and [9]F.-Y. Wang are with the State Key Laboratory for Management and Control of Complex Systems, and Beijing Engineering Research Center of Intelligent Systems and Technology, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China. [8]G. Xiong is also with the Cloud Computing Center, Chinese Academy of Sciences, Dongguan 523808, China (e-mail: {zhen.shen, xiuqin.shang, gang.xiong, feiyue.wang}@ia.ac.cn).

[3]Y. Li is with the University of Southern California, Los Angeles 90007, California, USA (e-mail: yuqingli@usc.edu).

[4]Y. Bao is with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: yin-bao@qq.com).

[5]X. Zhang is with the School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: zhangxipeng2015@ia.ac.cn).

[6]X. Dong is with the Qingdao Academy of Intelligent Industries, Qingdao 266109, Shandong, China (e-mail: xisong.dong@ia.ac.cn).

[7]L. Wan is with Beijing Ten Dimensions Technology Co., Ltd. (e-mail: li.wan@10dim.com).

978-1-7281-0356-3/19/$31.00 ©2019 IEEE

II. LITERATURE REVIEW

Most anti-deformation methods in AM are heuristic. For example, for the Fused Deposition Modeling (FDM), a scheme is proposed by using parameters of the printing process and the input data to predict the 2D deviation of the printed shape [6]. For the StereoLithography (SL), a method based on contour point displacement is utilized to predict the repeatable and unexpected deformation of product shapes [7]. Moreover, an optimal compensation method is validated to reduce shrinkage of cylindrical products in the SL process [8].

For the Digital Light Processing (DLP) based vat polymerization method, one image is projected onto the surface of the liquid resin in UV light, and a multiple exposure strategy is employed to reduce the internal force [9]. By combining deformation data and previous in-plane shape data, the Bayesian method is presented to predict the deformation of new shapes in [10]. One general method is to use the FEM for the error prediction. Bugeda *et al.* gave a numerical analysis of stereolithography processes using the FEM [11]. Huang and Jiang used dynamic FEM for numerical analysis of the mask type stereolithography process [12]. However, as remarked by Luan and Huang in [4], "improving part accuracy based purely on such simulation approaches is far from being effective", due to the complex processes of AM. Further, Luan *et al.* put forth a prescriptive statistically process control scheme to monitor shape deformation from shape to shape [13].

Neural networks usually extract features based on unsupervised, supervised or semi-supervised feature learning and hierarchical feature extraction algorithms, which usually include an input layer, an output layer and one or more hidden layers. There is great progress of neural networks in recent years, with successful applications in areas such as computer vision, natural language processing, robotics, speech, and audio recognition. The neural networks have been applied in the error compensation for subtractive manufacturing. Ramesh *et al.* summarized the thermal error compensation of machine tools and considered that the neural network is an important method for thermal error compensation of machine tools [14]. Raksiri and Parnichkun proposed to use a back-propagation neural network to estimate the geometric and cutting force induced error in a 3-axis milling machine [15]. Cho *et al.* presented an integrated machining error compensation approach based on polynomial neural network (PNN) method and inspection database based of an on-machine-measurement system, and implemented the method under actual machining conditions [16]. A back-propagation (BP) neural network combined with the bayesian-regularization is proposed to correctly predict the relationship between technical parameters and curve radius in ship production [17]. Yang *et al.* proposed a measurement optimal model based on genetic theory and neural network to compensate for the error of camera calibration [18]. Fines and Agah applied the artificial neural network to the calculation of the error compensation value of machine tools, and integrated the system into the open structure control system of actual machine tools [19]. In our previous work [5], we proposed to use 3D deep neural network to predict and and compensate the error for 3D models directly. We trained an "inverse function network" which used the deformed model as input and the nominal model as output. When testing, we input a model into the "inverse function network" and obtain the compensated model, which can be sent to the 3D printer. Then we obtain a more accurate actual model. In this paper, we take the in-plane 2D but more accurate case for the 3D printing, an autoencoder [20] deep neural network is employed for the prediction and compensation.

III. PROBLEM FORMULATION AND METHOD

In this section, we introduce the principle of our method for the prediction and error compensation in the AM and then show the detailed structures of PredNet and CompNet.

A. Problem Formulation

We give a brief description of the process of 3D printing we are concerned with in this paper. The DLP based 3D printing equipment and principle are shown in Fig. 1. The UV light from a light-emitting diode (LED) is used to cure the polymer resin layer by layer. A digital micro-mirror device (DMD) is used to project the image of the slice a model onto the surface of the liquid resin, which is cured and a layer is formed. Then the build platform moves and leaves gap for building next layer. The process repeats until the whole object is built.

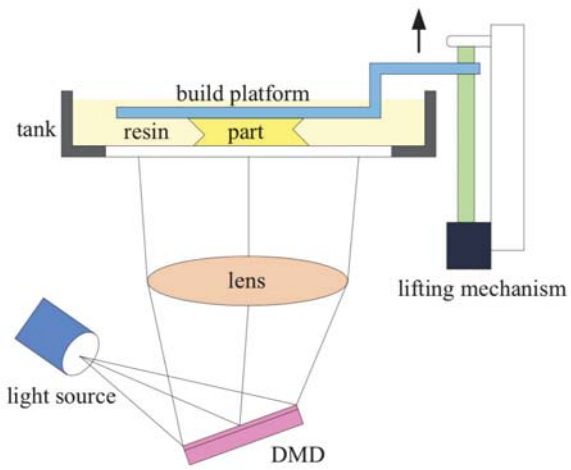


Fig. 1. Vat photopolymerization based on the DLP technology

The process of heating and cooling, adhesion and polymerization in the 3D printing process will cause the shrinkage and curling effects. Therefore, the deformation in the AM process is complex and is affected by a variety of factors, such as the type of printing material and shape of an object. Both the shape of the layers and the forces between the layers should be considered. We test our method on a category of objects. In this paper, use dental crowns. As shown in Fig. 2, we show a dental crown and its cross-section in a 3D view, as well as the image that should be projected onto the resin.

The gray shafts are supports to help build the model in the manufacturing process and will be removed after printing.

(a) (b) (c)

Fig. 2. 2(a) represents what a dental crown looks, and the gray shafts are supports to help build the model in the manufacturing process and will be removed after printing. 2(b) shows a layer of the dental crown. 2(c) is the image of the cross-section data, which is the data we use in our experiments.

We use a function to represent the deformation process of the model slice in printing, as shown in (1).

$$p' = f(p), \tag{1}$$

where p represents the nominal image of a slice of the model, and p' represents the deformed one after printing.

We train a deep neural network model "PredNet" to approximate $f(.)$. In other words, $f(\cdot)$ can be regarded as a virtual 3D printer, and "PredNet" is used to simulate the printing process of a 3D printer. We use $p_0, p_1, \ldots, p_{N-1}$ as the input and $p'_0, p'_1, \ldots, p'_{N-1}$ as the output, where N represents the number of training samples. For a test Model, we can obtain the predicted deformed model slice by the PredNet. And then we can calculate the predicted error easily.

Further, following the idea of "inverse function network" for error compensation proposed by us in [5], we assume that $f(\cdot)$ is invertible, we have (2),

$$p = f^{-1}\left(p'\right) \tag{2}$$

We train a deep neural network "CompNet" to approximate $f^{-1}(\cdot)$. In contrast to function 1, we use $p'_0, p'_1, \ldots, p'_{N-1}$ as the input, and $p_0, p_1, \ldots, p_{N-1}$ as the output. For the compensation, we input p to $f^{-1}(\cdot)$ to obtain a compensated model slice p''. We can input p'' into the virtual 3D printer (deformation function) to obtain a more accurate printed model. We show the ideas in Fig. 3.

B. Method

The architectures of PredNet and CompNet are the same, which are designed based on the U-Net [21], consisting of an Encoder and a Decoder, as shown in Fig. 4.

Encoder Architecture As shown in Fig. 4, we use the U-Net as a basic model to construct an Encoder consisting of a Conv_Block followed by a 2×2 max pooling operation. The structure of the Conv_Block is shown on the right side of Fig. 4, which consists of double 3×3 convolutional layers, Batch Normalization, and ReLU activation functions. We used mini-batch in our network, as it requires less memory and the network typically can be trained faster. The size of the input is the $batch\ size \times 1 \times H \times W$, where H and W represent the height and width of images. The number of

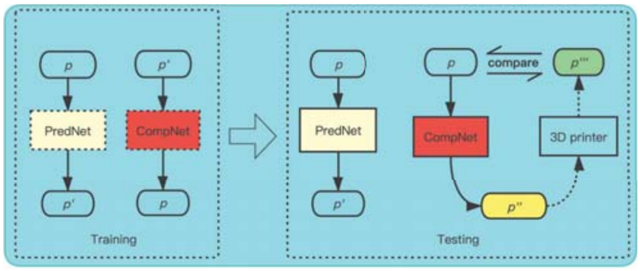


Fig. 3. The left diagram shows the training process: PredNet is trained using p_i as the input and p'_i as the output, while CompNet is trained by opposite input and output. The right diagram shows the application of the networks: the prediction is performed directly, while for the compensation, we input p into "CompNet" to obtain a compensated model slice p'', which is sent to the virtual 3D printer (deformation function) to get p''' . We can verify the compensation effect by comparing p''' with the nominal model slice p.

the feature channels in every layer can be seen at the top of the Fig. 4, After each Conv_Block, the number of feature channels is doubled.

Decoder Architecture We build a decoder network based on U-Net to recover the images of dental crowns from the Conv_Block_3.

If we stack a series of fully connected layers behind the Encoder network, and set the Batch size to 32 or larger, the computational burden will be too heavy for our GPU to afford. Thus, we design a network as shown in Fig. 4 instead. We use Upsample layer to halve the feature channels and double the size of the feature map and then connect the corresponding size of feature map (Conv_Block_2) on the encoder to it, and then a Conv_Block is used to halve the number of the feature channel. Finally, a 1×1 convolutional layer, Batch Normalization, and Sigmoid function are used to map each feature vector to the required number of classes. The advantage of this architecture is that it significantly reduces the number of parameters that need to be trained compared to the fully connected layer.

A suitable loss function can be critical The loss function should calculate the differences between predicted images and nominal images, and shoulbd be differentiable and robust against outliers [22]. We use the Binary Cross Entropy (BCE) as our loss function.

IV. Experiments

In this section, we show the experiments of the proposed method through dental crowns datasets and the performances of our deep neural networks.

A. Datasets

Dataset by Simulation We have a large amount of data of the crowns from dental hospitals and factories. However, it is not easy to obtain the data for the printed crowns. There might be several ways. One is to measure the printed crowns by a 3D scanner. Another is to cut the printed crown into layers and obtain the data by a camera. The third is to pause and take a photo after a layer is built in the printing process. The three methods may face difficulties of errors and lack

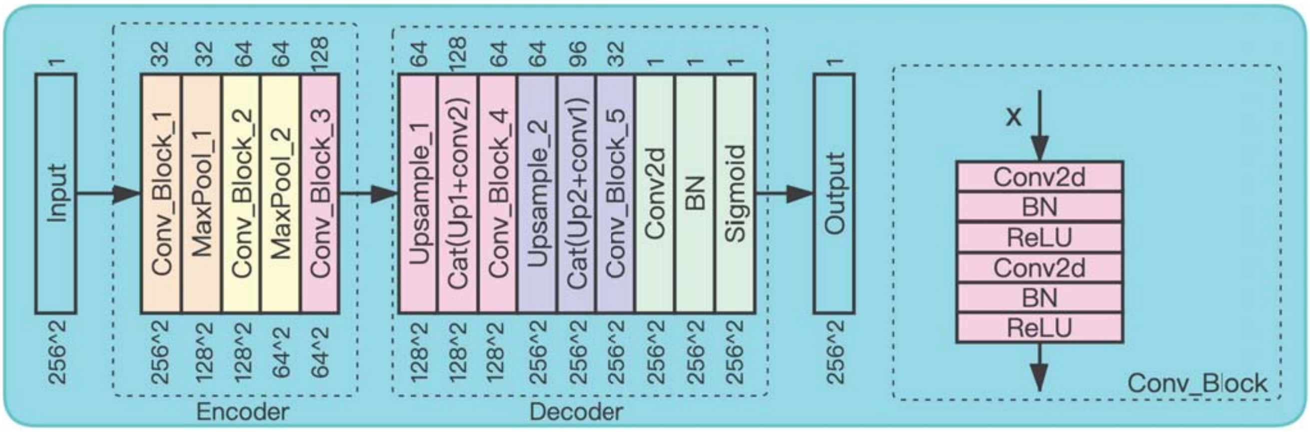


Fig. 4. The structure of PredNet and CompNet is shown on the left. The input of the network is the preprocessed images of size 256×256. the figure at the top of the picture is the number of the feature channels in each layer, while the figure at the bottom of the picture is the size of the feature map. The structure of the Conv_Block is shown on the right. The Conv_Block consists of two convolutional layers as well as the Batch Normalization, and is activated by the ReLU function.

of automatic tools. In this paper, as a first step to tackle this problem, we use a simulation method to generate data for deformed images to train a deep neural network together with the nominal image. We test the following three basic kinds of deformations: translation, scaling and rotation.

- Translation: Move the image by 20% to the right (13 pixels).
- Scaling down: Shrink image by a scale factor of 0.9.
- Rotation: Rotate the image by 9 degrees (10 percent of 90 degrees) around the center of the image.

Dataset Construction We use 71 single crown models. All these 3D models are designed by computer-aided design (CAD) software and then converted to STL models. The STL model is a tessellated representation of a solid object and has become the most popular file format for additive manufacturing. In Fig. 5 we show some samples of the crown models. We slice the models with uniform thickness and obtain a binary image data set of 28,433 images. All the images are clipped into 256×256.

(a) (b) (c)

Fig. 5. Samples of dental crown models

B. Metric

We call the image given by the deformation function as "real output". Its pixel color is either white or black. The white pixels represent the foreground, which is the information we care about, i.e., the dental crown layer. We represent the white pixel with a 1. The black pixels represent the background, and are shown by 0. The output of the network requires a threshold value. If the output is not less than the threshold value, it will be set to 1, otherwise it will be set to 0. We call the output image of the network "expected output".

We use three metrics, *Recall*, *Precision*, and $F1$ *score*, to evaluate our method. *Recall* shows the ability of the network to recognize the pixel value of 1, namely the dental crown. The higher the *Recall*, the better the network can identify dental crowns. The *Precision* represents the ability of network to separate the dental crown from the background. The higher the *Precision*, the stronger the ability of the network to segment the dental crown correctly. While $F1$ *score* is the harmonic mean of *Recall* and *Precision*.

C. Implementation

Our network is implemented with PyTorch 1.0 on an NVIDIA TITAN V. We optimize the networks using Adam [23] with an initial learning rate of 0.0001, the batch size of 32, and the threshold of 0.5. The input vectors and target vectors are randomly divided into two sets, 80% as training data and 20% as testing data.

D. Results

Here we show two experiments. For the first experiment, PredNet is trained to approximate the deformation function $f(\cdot)$. For the second experiment, we train a CompNet as the compensation network to approximate the inverse deformation function $f^{-1}(\cdot)$.

For the PredNet, we use the nominal dental crowns as the input and the three deformations of the dental crowns as the output, respectively. In Fig. 6, we show the loss function graphs of training and testing of the network. It can be seen from the graphs that the neural network we build has good fitting ability.

The predicted results of three kinds of deformation are shown in TABLE I. It can be seen from the experimental results that our network has a good ability of predicting the

(a) (b) (c)

Fig. 6. Loss functions for the PredNet, left to right: translation, scaling down and rotation

errors for the three kinds of deformation. The $F1$ scores of those three deformations are all larger than 98%.

TABLE I
THE PERFORMANCE OF THE PREDNET ON THE THREE KINDS OF DEFORMATION

(a) Performance of Training

Deformation	Recall	Precision	$F1$ Score
Translation	1.000	1.000	**1.000**
Scaling down	0.989	0.989	**0.989**
Rotation	0.982	0.982	**0.980**

(b) Performance of Testing

Deformation	Recall	Precision	$F1$ Score
Translation	1.000	1.000	**1.000**
Scaling down	0.984	0.993	**0.989**
Rotation	0.982	0.978	**0.980**

For the CompNet, we use the deformed images as input and the nominal images as the output. The loss function graphs of training and testing of CompNet are shown in Fig. 7, which also show a good fitting ability.

(a) (b) (c)

Fig. 7. Loss function for the CompNet, left to right: translation, scaling down and rotation

The performance of the CompNet is shown in TABLE II. We can see that the $F1$ scores are all larger than 98%.

TABLE II
THE PERFORMANCE OF THE COMPNET ON THREE KINDS OF DEFORMATION

(a) Performance of Training

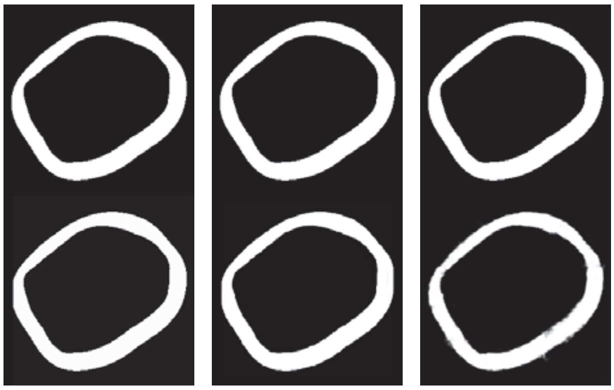
Deformation	Recall	Precision	F1 Score
Translation	1.000	1.000	**1.000**
Scaling down	0.992	0.992	**0.992**
Rotation	0.982	0.982	**0.982**

(b) Performance of Testing

Deformation	Recall	Precision	F1 Score
Translation	1.000	1.000	**1.000**
Scaling down	0.993	0.991	**0.992**
Rotation	0.983	0.980	**0.982**

TABLE III
THE $F1$ SCORES BEFORE AND AFTER THE COMPENSATION

Deformation	Before compensation	After compensation
Translation	0.6894	**0.9995**
Scaling down	0.7188	**0.9893**
Rotation	0.8906	**0.9671**



(a) (b) (c)

Fig. 8. The compensation effects, up: nominal image, down: actual image, left to right: translation, scaling down and rotation

Furthermore, after training the CompNet, we input the nominal image into it and can obtain the compensated image, which can be input into the deformation function to obtain the actual image. By comparing the actual image with the nominal image, we know the performance of the compensation, shown in TABLE III. In Fig. 8, we show samples for the compensation.

V. CONCLUSION AND FUTURE WORK

In this paper, we have built two networks: PredNet and CompNet, for the prediction and compensation of in-plane shape deformation in the additive manufacturing. We test on three kinds of deformation: translation, scaling down and rotation and show that the networks work well. The real case can be more complex and nonlinear deformation exist. In the future, we will try to use the Finite Element Method may be used as the simulation method to generate deformed models. Also, we will try to collect real data and test the method.

References

[1] P. Markillie, “A third industrial revolution,” *The Economist*, vol. 21, pp. 3–12, 2012.

[2] P. Jensen-Haxel, “3D printers, obsolete firearm supply controls, and the right to build self-defense weapons under Heller,” *Golden Gate UL Rev.*, vol. 42, p. 447, 2011.

[3] Y.-M. Huang and H.-Y. Lan, “CAD/CAE/CAM integration for increasing the accuracy of mask rapid prototyping system,” *Computers in Industry*, vol. 56, no. 5, pp. 442–456, 2005.

[4] H. Luan and Q. Huang, “Prescriptive modeling and compensation of in-plane shape deformation for 3-D printed freeform products,” *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 73–82, 2017.

[5] Z. Shen, X. Shang, M. Zhao, X. Dong, G. Xiong, and F. Wang, “A learning-based framework for error compensation in 3-D printing,” *IEEE Transactions on Cybernetics*, 2019, in press.

[6] L. Cheng, A. Wang, and F. Tsung, “A prediction and compensation scheme for in-plane shape deviation of additive manufacturing with information on process parameters,” *IISE Transactions*, vol. 50, no. 5, pp. 394–406, 2018.

[7] Z. Zhu, N. Anwer, and L. Mathieu, “Deviation modeling and shape transformation in design for additive manufacturing,” *procedia CIRP*, vol. 60, pp. 211–216, 2017.

[8] Q. Huang, J. Zhang, A. Sabbaghi, and T. Dasgupta, “Optimal offline compensation of shape shrinkage for three-dimensional printing processes,” *IIE Transactions*, vol. 47, no. 5, pp. 431–441, 2015.

[9] H. John, V. Schillen, and A. El-Siblani, “Method for producing a three-dimensional object by means of mask exposure,” Aug. 26 2014, US Patent 8,815,143.

[10] A. Sabbaghi, Q. Huang, and T. Dasgupta, “Bayesian model building from small samples of disparate data for capturing in-plane deviation in additive manufacturing,” *Technometrics*, vol. 60, no. 4, pp. 532–544, 2018.

[11] G. Bugeda, M. Cervera, G. Lombera, and E. Onate, “Numerical analysis of stereolithography processes using the finite element method,” *Rapid Prototyping Journal*, vol. 1, no. 2, pp. 13–23, 1995.

[12] Y.-M. Huang and C.-P. Jiang, “Numerical analysis of a mask type stereolithography process using a dynamic finite-element method,” *The International Journal of Advanced Manufacturing Technology*, vol. 21, no. 9, pp. 649–655, 2003.

[13] H. Luan, B. K. Post, and Q. Huang, “Statistical process control of in-plane shape deformation for additive manufacturing,” in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*. IEEE, 2017, pp. 1274–1279.

[14] R. Ramesh, M. Mannan, and A. Poo, “Error compensation in machine tools—a review: Part ii: thermal errors,” *International Journal of Machine Tools and Manufacture*, vol. 40, no. 9, pp. 1257–1284, 2000.

[15] C. Raksiri and M. Parnichkun, “Geometric and force errors compensation in a 3-axis CNC milling machine,” *International Journal of Machine Tools and Manufacture*, vol. 44, no. 12-13, pp. 1283–1291, 2004.

[16] M.-W. Cho, G.-H. Kim, T.-I. Seo, Y.-C. Hong, and H. H. Cheng, “Integrated machining error compensation method using OMM data and modified PNN algorithm,” *International Journal of Machine Tools and Manufacture*, vol. 46, no. 12-13, pp. 1417–1427, 2006.

[17] P. Li, F. Gao, C. Wang, Y. Mao, and J. Cui, “Research of the curve radius of shape formed in profile cold forming with BP neural networks approach based on experiment,” *Journal of Ship Production and Design*, vol. 32, no. 1, pp. 50–58, 2016.

[18] P. Yang, J. Zhu, and J. Li, “Plane measurement based on genetic optimization calibration and compensation of neural networks,” *Electronic Science and Technology*, no. 5, p. 14, 2017.

[19] J. M. Fines and A. Agah, “Machine tool positioning error compensation using artificial neural networks,” *Engineering Applications of Artificial Intelligence*, vol. 21, no. 7, pp. 1013–1026, 2008.

[20] G. E. Hinton, A. Krizhevsky, and S. D. Wang, “Transforming auto-encoders,” in *International Conference on Artificial Neural Networks*. Springer, 2011, pp. 44–51.

[21] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical Image Computing & Computer-assisted Intervention*, 2015.

[22] H. Fan, H. Su, and L. J. Guibas, “A point set generation network for 3D object reconstruction from a single image,” in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 605–613.

[23] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.