International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 8, 20285
https://theaspd.com/index.php

Evaluating Process Parameters Impact On Deformation In 3D Printing: A Machine Learning And Simulation Study

Jyoti shah[1], Prashant Johri [2], Mayur P. Singh[1]
[1]Department of Mechanical Engineering, Galgotias University
[2]Department of Computer Science Engineering, Galgotias University

Abstract: *This paper presents a deformation prediction analysis for 3D printed parts, employing machine learning models alongside finite element simulations to develop a comprehensive dataset. We utilized five machine learning algorithms—Linear Regression, Decision Tree Regressor, Random Forest Regressor, Support Vector Regressor, and K-Nearest Neighbors Regressor—to assess the influence of critical process parameters, including layer height, print temperature, print speed, and bed temperature, on the deformation of printed components. Finite element simulations were conducted to generate accurate deformation data, which served as the foundation for training the machine learning models. Each algorithm's performance was rigorously evaluated, revealing insights into their predictive capabilities and the significant effects of the analyzed parameters on deformation outcomes. Importantly, the study also contributes to environmental protection and sustainable manufacturing by reducing material waste, minimizing failed prints, and lowering energy consumption, thereby supporting environmental management and green technologies. The results provide valuable guidance for optimizing 3D printing processes, ultimately enhancing printed parts' mechanical performance, reliability, and sustainability*

Keywords: *Deformation prediction, machine learning, finite element simulation, 3D printing, process parameters, layer height, print temperature, print speed, bed temperature, regression algorithms.*

1.0 INTRODUCTION

The advent of additive manufacturing, commonly known as 3D printing, has revolutionized various industries, including healthcare, aerospace, and manufacturing (Hsieh et al.). One of the key challenges in 3D printing is the accurate prediction of the deformation of printed objects, which can have a significant impact on the final product quality and performance. Traditional methods for predicting deformation, such as finite element analysis, can be computationally expensive and time-consuming, particularly for complex geometries. In recent years, the development of machine learning models has emerged as a promising approach to address this challenge. Machine learning algorithms can analyze large datasets of 3D printing parameters, process conditions, and the resulting deformation patterns to create predictive models that can be used to optimize the 3D printing process and minimize deformation.

Several reports are available in the literature dealing with the experimental and computation study of 3D printing. In this context, Ahn et al. [1] utilized factorial design to assess the impacts of build orientation, air gap, temperature, road width, and filament color on the tensile and compressive strength of ABS parts produced via fused deposition modeling (FDM). Their results indicated optimal processing parameters yielding tensile strengths of 65-72% with a 0.003 mm air gap and compressive strengths of 80-90%, compared to injection-molded components. Wang et al. [2] analyzed input factors affecting FDM tensile properties using the Taguchi method and Gray relational analysis, identifying build orientation as critical. The peak tensile strength reached 24.36 MPa at a layer height of 0.254 mm and a zero-degree orientation. Equbal et al. [3] examined five process parameters impacting the dimensional accuracy of FDM parts, employing Taguchi, artificial neural networks (ANN), and fuzzy logic methods (FLM) for mathematical modeling.

Yadav et al. [4] developed a genetic algorithm-integrated artificial neural network (GA-ANN) to optimize process parameters in fused deposition modeling (FDM) for enhancing tensile strength. Their findings showed a strong correlation between predicted and experimental results, validating the optimization method. In a separate study, Yadav et al. [5] employed the Adaptive Neuro-Fuzzy

Inference System (ANFIS) to model FDM process parameters, achieving improved estimation accuracy over conventional numerical methods. Various researchers have effectively utilized soft computing techniques, including ANN, ANFIS, and GA, to analyze process parameters and optimize output responses. Saad et al. [6] applied response surface methodology (RSM) to examine the effects of layer width, print velocity, and nozzle temperature on the surface quality of printed parts, concluding that reduced layer thickness and print velocity significantly lower surface roughness. The optimization accuracy of their techniques exceeded that of traditional RSM methods.

Deswal et al. [7] investigated the effects of various process parameters on the tensile strength of FDM-fabricated parts, optimizing these parameters using machine learning algorithms, including GA-ANN, GA-RSM, and GA-ANFIS. The hybrid models demonstrated significant potential for accurate prediction and optimization across various industrial applications. Ziadia et al. [8] analyzed the impact of process parameters and material selection on the mechanical properties of 3D-printed components, employing genetic algorithms for multi-objective optimization. Their findings confirmed that both parameters and material choices significantly influence mechanical properties, with the genetic algorithm effectively identifying optimal values. Zhang et al. [9] applied machine learning techniques to create a data-driven model for predicting melt pool temperature in the directed energy deposition (DED) process, utilizing Extreme Gradient Boosting and Long Short-Term Memory networks. Both algorithms exhibited high scalability and effectiveness in analyzing time-series data, accurately forecasting melt pool temperatures.

Singh et al. [10] employed a Back-Propagation Neural Network (BPNN) to assess relative density and predict porosity in bronze 3D printing. The model was trained using multiple algorithms and optimized with a Genetic Algorithm (GA) for enhanced accuracy. Caiazzo and Caggiano [11] applied machine learning in the directed energy deposition (DED) process to correlate input parameters of laser metal deposition with the output geometrical characteristics of printed components. They trained an artificial neural network (ANN) using experimental data in a two-phase process, demonstrating that ML can accurately predict processing parameters for specific metallic shapes. Tapia et al. [12] utilized a Gaussian process model to predict melt pool depth in laser powder-bed fusion, based on experimental data from 316L stainless steel, incorporating scan speed and laser power as input factors. The validated model showed exceptional performance with a low mean prediction error.

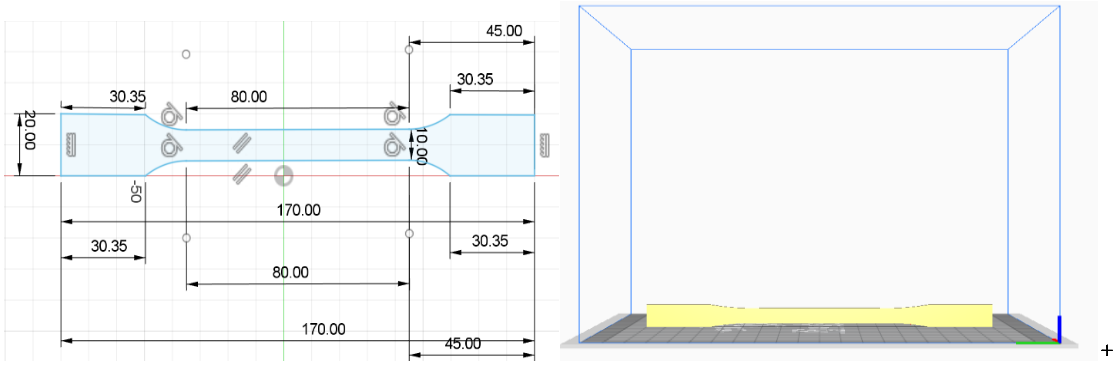
Sridhar et al. [13] aimed to optimize FDM process parameters using a multidisciplinary evolutionary algorithm and machine learning models. A response surface approach generated a dataset, with a screening design applied to 14 parameters. After training and testing, the random forest model achieved over 90% prediction accuracy. The study focused on optimizing infill percentage, layer height, print speed, and extrusion temperature to maximize tensile strength. Particle Swarm Optimization (PSO) and Differential Evolution (DE) were used for parameter optimization, with PSO proving effective in enhancing the tensile strength and mechanical properties of FDM-fabricated specimens [14]. Ulkir and Akgun [15] examined the influence of key printing factors—layer thickness, raster angle, and infill density—on the flexural strength of PETG components produced via FDM. They employed fuzzy logic modeling and analysis of variance, identifying infill density as the most significant factor affecting flexural strength.

Despite the advances made in optimizing FDM process parameters, existing studies primarily focus on tensile and flexural strengths, leaving a gap in understanding the deformation behavior of 3D printed parts under varying conditions. Therefore, the present work aims to develop a robust framework for predicting deformation using machine learning models, supported by finite element simulations, to comprehensively analyze the effects of layer height, print temperature, print speed, and bed temperature on the deformation of 3D printed components.

2. METHODOLOGY

2.1 Development of Dataset for Predicting Deformation (Warpage)

The dataset was specifically designed to capture key process parameters that influence deformation (warpage) in the Fused Filament Fabrication (FFF) additive manufacturing process. The input parameters considered include layer height, print speed, print temperature, and bed temperature, all of which significantly affect warpage. For each simulation, the corresponding deformation data was recorded. The dataset was organized with each row representing a unique combination of input parameters and a column representing the output deformation value (given in Table 1). The dataset was sufficiently large to cover a broad range of conditions, enabling the machine learning models to learn the complex relationships between the process parameters and the warpage response. This diverse set of data allows for a comprehensive exploration of how each parameter affects the deformation behavior.



(a) (b)

Figure 1: (a) Dimensions ISO 527-2 Type 1A (b) Orientation of Specimen on Print Bed visualised from Side View

2.2 Thermo-Mechanical Model for Warpage

The numerical simulations were conducted using Digimat, which provided a detailed understanding of the thermal and mechanical behavior of the printed parts. #D models has been developed as per the ASTM standard and finite element analysis has been carried out to investigate the deformation of the printed part (Figure 1). The finite element (FE) model of the printed structure was divided into 36,800 voxels, representing the physical part in the simulation. Each voxel captured thermal and mechanical interactions, enabling accurate predictions of warpage. The various steps involved in the simulation are as follows:

2.2.1 Heat Transfer Equation

The temperature distribution $T(x, y, z, t)$ across the part during printing is calculated using the heat transfer equation:

$$\rho c_p \frac{\partial T}{\partial t} = k\nabla^2 T + Q \qquad (1)$$

Where, ρ= material density, c_p= specific heat capacity, k= thermal conductivity, Q= heat due to printing nozzle, $\nabla^2 T$= spatial temperature gradient

2.2.2 Thermal Strain Calculation

As the material cools, it shrinks, generating thermal strain:

$$\epsilon_{th} = \alpha \Delta T \qquad (2)$$

where: α = coefficient of thermal expansion, ΔT = difference between the initial and current temperature.

2.2.3 Elastic and Inelastic Strain Calculation

The total strain (ϵ) in the material consists of elastic strain (ϵ_{el}), thermal strain (ϵ_{th}), and plastic strain (ϵ_{pl}) if the material undergoes any permanent deformation:

$$\epsilon = \epsilon_{el} + \epsilon_{th} + \epsilon_{pl} \qquad (3)$$

For isotropic linear materials, the elastic strain can be related to stress using Hooke's Law:

$$\sigma = E \cdot \epsilon_{el} \qquad (4)$$

where E is the Young's modulus of the material.

2.2.4 Residual Stress Calculation

Residual stress (σ_{res}) arises from differential cooling and thermal contraction:

$$\sigma_{res} = E \cdot (\epsilon_{th} + \epsilon_{pl}) - \sigma_{eq} \qquad (5)$$

where σ_{eq} is the equilibrium stress, ensuring compatibility across layers.

2.2.5. Finite Element Method (FEM) for Deformation

The residual stresses calculated are applied to a finite element model to predict the deformation or warpage ($\boldsymbol{u}$) of the part:

$$\boldsymbol{K} \cdot \boldsymbol{u} = \boldsymbol{F} \qquad (6)$$

where: $\boldsymbol{K}$**, and** $\boldsymbol{F}$ stiffness matrix and force vector respectively.

2.3 Mesh Convergence Analysis

A mesh convergence analysis was performed to ensure simulation accuracy as shown in Fig. 2. The study tested different mesh sizes and found that a mesh size of 0.65 mm provided the best balance between precision and computational efficiency. This level of mesh refinement was critical in capturing small but significant deformations in the part due to the thermal and mechanical effects present during printing.

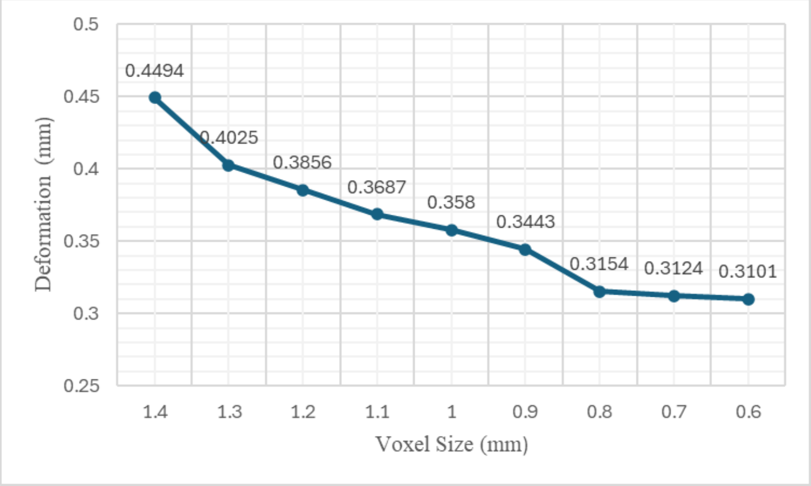


Figure 2: Convergence study

2.4 **Dataset Overview**

The dataset contained input features that are critical parameters influencing the deformation in 3D printing. These input features were:

- Layer height (mm)
- Print temperature (°C)
- Print speed (mm/s)
- Bed temperature (°C)

The output variable was:

- **Deformation**: The degree of deviation from the intended shape of the printed part. Glimpse of dataset us given in Table 1.

Table 1: Dataset

Layer Height (mm)	Bed Temperature (°C)	Print Speed (mm/s)	Print Temperature (°C)	Deformation (mm)
0.08	40	20	150	0.28
0.08	40	20	190	0.4113
0.08	40	20	250	0.5981
0.08	40	40	150	0.2789
0.08	40	40	190	0.4097
0.08	40	40	250	0.5959
0.08	40	70	150	0.2774
0.08	40	70	190	0.4075
0.08	40	70	250	0.5928
0.08	60	20	150	0.2468
0.16	40	70	150	0.17
0.16	40	70	190	0.3047
0.16	40	70	250	0.4428
0.16	60	20	150	0.1868
0.28	40	20	150	0.1129
0.28	40	20	190	0.1594
0.28	40	20	250	0.1885
0.28	40	40	150	0.1124

2.5 Data Splitting

To ensure robust model evaluation, the dataset was split into two subsets:

- **Training set**: 80% of the data was used to train the models.
- **Testing set**: The remaining 20% was reserved for testing the models and evaluating their performance.

The split aimed to prevent overfitting and ensure that the models generalized well to unseen data.

2.6 Development of Predictive Models for Deformation

A range of machine learning regression models was implemented to predict deformation (warpage) based on the selected process parameters. A detailed discussion about the various algorithm used has been given in Appendix 1.

a) Linear Regression

A simple model that assumes a linear relationship between the input variables and the output (deformation). The model fits a linear equation to the data:

$$\hat{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 \qquad (7)$$

Where X_1, X_2, X_3, X_4 are the input features and $\boldsymbol{\beta_0}, \boldsymbol{\beta_1}, \boldsymbol{\beta_2}, \boldsymbol{\beta_3}, \boldsymbol{\beta_4}$ are the coefficients determined during training.

b) Decision Tree Regressor

This model splits the data into subsets by making decisions at each node based on the input features. It recursively splits the data, reducing the error at each split by minimizing the mean squared error (MSE). The tree captures non-linear relationships between the features and deformation.

c) Random Forest Regressor

An ensemble method, Random Forest builds multiple decision trees and aggregates their predictions. This model helps reduce overfitting and increases the robustness of predictions. Each tree is trained on random subsets of data, and the final prediction is an average of all the trees' predictions.

d) Support Vector Regressor (SVR)

SVR is a model that finds a hyperplane in a high-dimensional space that best fits the data while maintaining a margin of tolerance (ϵ) for errors. It uses kernel functions to capture non-linear relationships between the input features and the output deformation:

$$\hat{y} = \sum_{i=1}^{N} \alpha_i K(X_{\text{new}}, X_i) + b \qquad (8)$$

e) K-Nearest Neighbors Regressor (KNN)

KNN is a non-parametric model that predicts deformation by averaging the output values of the kkk nearest training samples based on a distance metric like Euclidean distance. It is highly flexible but can be sensitive to noise and outliers.

2.7. Model Evaluation

The models were evaluated based on the following metrics:

- Mean Squared Error (MSE): Measures the average squared difference between actual and predicted deformation values.
- R^2 Score: Indicates the proportion of variance in the target variable that is predictable from the features (closer to 1 is better).

3.0 RESULTS AND DISCUSSION:

3.1 Analysis of various machine learning models:

The performance of various machine learning models—Linear Regression, Decision Tree Regressor, Random Forest Regressor, Support Vector Regressor (SVR), and K-Nearest Neighbors Regressor (KNN)—was evaluated for predicting deformation (warpage) in the Fused Filament Fabrication (FFF) process. The results are shown in Figure 3 and Table 2.

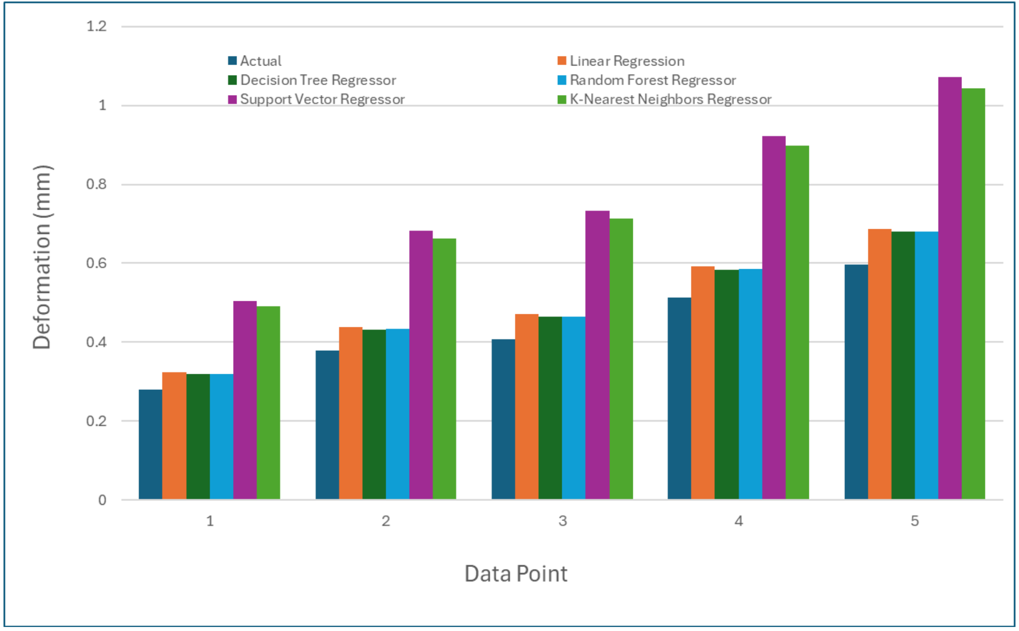


Figure 3: Comparison of various Machine learning models

Table 2: The performance of each model is summarized below:

Model	MSE	R^2 Score
Linear Regression	0.001758	0.851583

Decision Tree Regressor	0.001398	0.881942
Random Forest Regressor	0.001425	0.879667
Support Vector Regressor	0.009548	0.193884
K-Nearest Neighbors Regressor	0.009510	0.197095

It is evident from the results that, the Linear Regression demonstrated a reasonable performance, with an R^2 score of 0.8516 and MSE of 0.001758, explaining about 85% of the variance in deformation. Although it effectively captured the linear relationships between input parameters like layer height, print speed, and temperatures, it was limited in handling non-linearities and interactions between variables, which likely exist in this complex manufacturing process. In Decision Tree Regressor model, R^2 score is 0.8819 and an MSE is 0.001398 which reflect better results compared to Linear Regression. Its ability to model non-linear relationships and capture the complex interactions between parameters contributed to its improved performance. This is due to the fact that tree-based model can better identify thresholds and discontinuities in the data, making it highly effective for predicting deformation.

Random Forest Regressor closely followed, with an R^2 score of 0.8797 and an MSE of 0.001425. As an ensemble of decision trees, Random Forest enhances generalization by reducing overfitting, although it performed marginally worse than the single Decision Tree in this case. Nevertheless, it remains a reliable model due to its balance between bias and variance, making it suitable for deformation prediction in similar scenarios.

In contrast, Support Vector Regressor (SVR) showed significantly poorer results, with an R^2 score of 0.1939 and an MSE of 0.009548. This model failed to capture the complexity of the data, potentially due to the inadequacy of its linear kernel or insufficient hyperparameter tuning. Its inability to model the non-linear interactions between input variables rendered it ineffective for predicting deformation. Similarly, K-Nearest Neighbors (KNN) Regressor performed poorly, with an R^2 score of 0.1971 and an MSE of 0.009510. Like SVR, KNN struggled to handle the non-linearities in the data and failed to model the intricate relationships between process parameters. Its local approximation method likely missed the broader trends and interactions needed to predict warpage accurately.

3.2 Correlation matrix:

The correlation matrix shown in Figure 4 illustrates the relationships between various key parameters—layer height, bed temperature, print speed, and print temperature—and their influence on the deformation of a 3D-printed part. Among these, layer height and print temperature stand out as the most significant factors affecting deformation. There is a strong negative correlation (-0.70) between layer height and deformation, indicating that as the layer height increases, the deformation of the part decreases. This suggests that using larger layer heights can result in a more dimensionally stable final product.

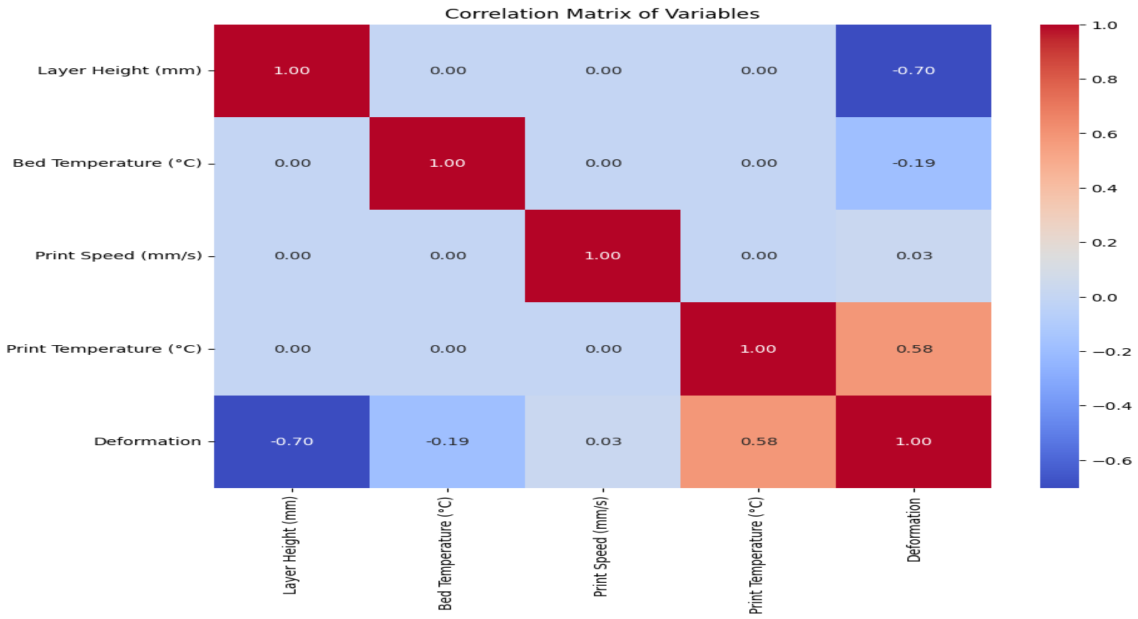


Figure 4: Correlation matrix among the input parameters and output parameter

Conversely, the print temperature shows a moderately strong positive correlation (0.58) with deformation, meaning that higher print temperatures tend to increase the deformation. This highlights the importance of carefully controlling the print temperature to minimize warping or unwanted deformations in the 3D-printed part. Bed temperature, while negatively correlated with deformation (-0.19), has a weaker influence, suggesting that while raising the bed temperature may slightly reduce deformation, it is not a major factor. Print speed, on the other hand, shows almost no correlation (0.03) with deformation, indicating that variations in print speed have little to no effect on the structural integrity of the printed part.

In terms of the relationships between the input parameters themselves, the matrix shows that layer height, bed temperature, print speed, and print temperature are largely independent of each other, with most inter-parameter correlations hovering around zero. This means that adjustments to one parameter are unlikely to affect the others, allowing for more flexibility in the optimization process. Overall, to minimize deformation, controlling layer height and print temperature should be the primary focus, while bed temperature and print speed can be considered secondary factors.

3.3 Parametric study:

It is evident from the above discussion that the Decision Tree Regressor performed the best with the lowest Mean Squared Error (MSE = 0.001398) and the highest R^2 score (0.881942). This model explains approximately 88% of the variance in the deformation data, making it the most suitable for predicting deformation in 3D-printed materials. Therefore in the subsequent study, Decision Tree Regressor will be used to perform the analysis.

3.3.1 Influence of layer height on the deformation

Figure 5 demonstrate the relationship between predicted deformation and layer height in 3D printing process. It can be observed that deformation remains relatively constant within certain height ranges, suggesting threshold effects. Generally, as layer height increases, predicted deformation decreases, likely due to the greater mass and stability of thicker layers. The results also show three distinct steps, indicating different material behaviors or deformation regimes, where the first step may represent a range more susceptible to deformation, while later steps indicate increased resistance. This behavior could be attributed to material properties that influence deformation at various heights, manufacturing process factors such as cooling rates and printing orientation, and layer adhesion quality, which can affect risks of delamination or warping. Overall, the findings underscore the importance of optimizing layer height to enhance print quality and stability.

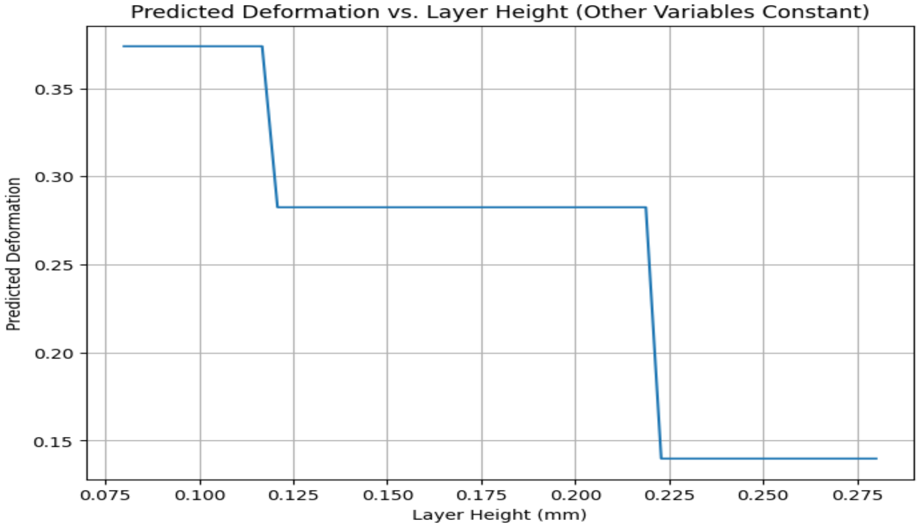


Figure 5: Layer Height vs Deformation

3.3.2 Influence of Bed Temperature on the deformation

Figure 6 highlights the influence of bed temperature on the deformation of 3D printed part. It is evident from the results that lower bed temperatures (below 50°C) are associated with higher deformation due to poor layer adhesion, leading to warping. As the bed temperature increases, particularly around the 70°C–80°C range, the deformation decreases significantly. This indicates that in this range, the print bed provides better support and adhesion for the printed material, reducing the chances of deformation. However, after a certain point (around 80°C), the deformation stabilizes, indicating diminishing returns from further increasing the bed temperature. Beyond this threshold, other factors such as print temperature or layer height might have a more pronounced effect on further improving part quality.

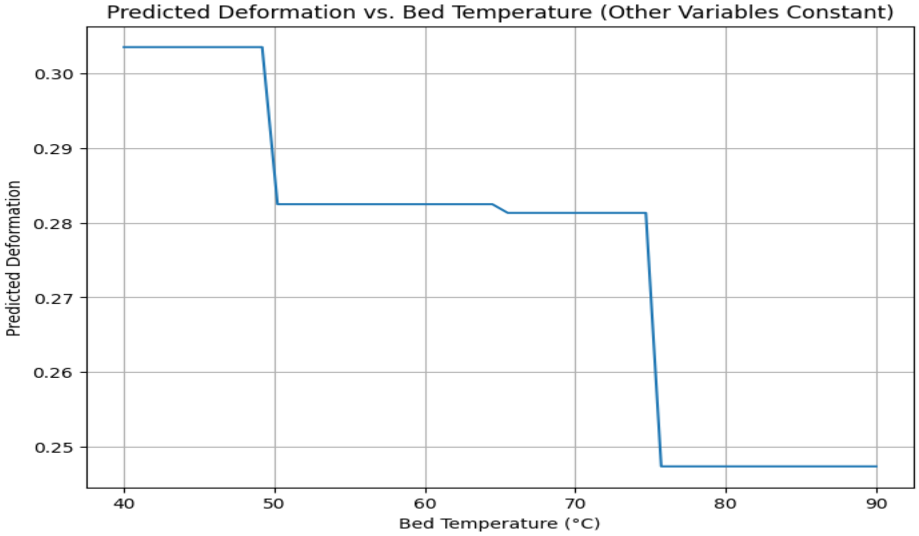


Figure 6: Bed Temperature vs Deformation

3.3.3 Influence of Print Speed on the deformation

Figure 7 presents the effect of print speed on the deformation of 3D printed part. It is evident from the results that as print speed increases, predicted deformation generally decreases. However, there are specific print speed ranges where the material exhibits different deformation behaviors. It is clear from the results that moderate print speeds (around 30–50 mm/s) showing the least deformation. This implies that there's an optimal range of print speeds where deformation is minimized, likely because the material has enough time to solidify properly without excessive warping or distortion. At very low speeds (around 20–30 mm/s), the deformation is higher, possibly due to excessive heat accumulation in the printed layers, leading to increased warping. Similarly, at very high speeds (above 50 mm/s), the deformation increases again, likely due to incomplete layer cooling and weaker layer bonding.

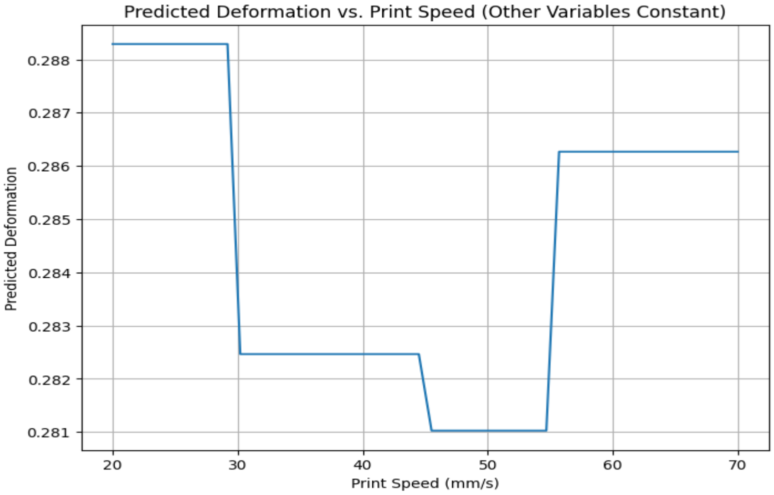


Figure 7: Print Speed vs Deformation

3.3.4 Influence of Print Temperature on the deformation

In figure 8, the effect of print temperature on the deformation of 3D printed part has been given. It is clear form results that at lower temperatures, specifically between 160°C and 180°C, the deformation remains constant at approximately 0.20, indicating minimal effect of temperature on deformation within this range. However, as the temperature increases beyond 180°C, a slight rise in deformation occurs, reaching around 0.25 by 200°C. The most significant change happens between 200°C and 220°C, where the predicted deformation sharply rises to about 0.40, suggesting a substantial structural change or weakening in response to the higher temperature. Beyond 220°C, up to 240°C, the deformation plateaus at 0.40, indicating that further increases in temperature no longer lead to additional deformation. This behavior reflects the material's or system's sensitivity to temperature, with distinct transitions at 180°C and 220°C, followed by stabilization.

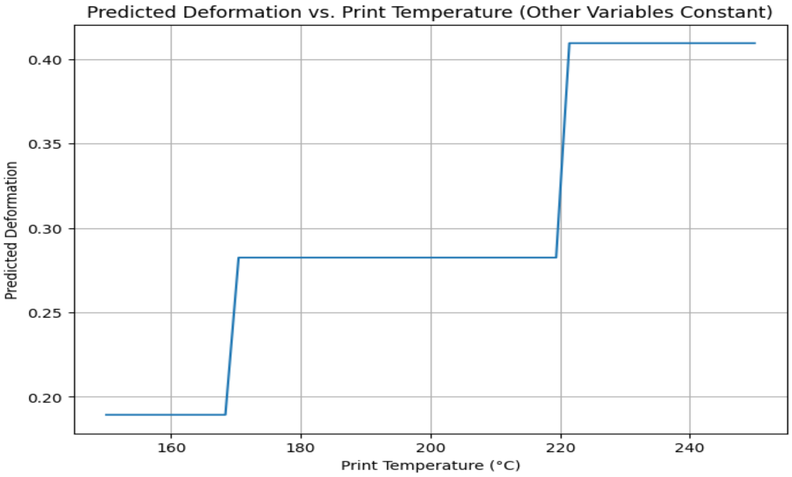


Figure 8: Print Temperature vs Deformation

4.0 CONCLUSION

In this study, we developed a robust framework for predicting deformation in 3D printed parts, utilizing machine learning models and finite element simulations to create a comprehensive dataset. Our analysis demonstrated that the Decision Tree Regressor emerged as the most accurate model for predicting deformation, effectively capturing the non-linear relationships inherent in the data. While other models, such as Linear Regression and Random Forest, also performed well, they did not match the predictive accuracy of the Decision Tree. This work underscores the significance of critical process parameters—layer height, print temperature, print speed, and bed temperature—in influencing deformation outcomes. The findings provide essential insights for optimizing 3D printing processes, thereby enhancing the mechanical performance and reliability of printed components. Future research could further explore the integration of additional parameters and advanced modeling techniques to improve predictive capabilities in additive manufacturing.

REFERENCES:

1. Ahn, S.H., Montero, M., Odell, D., Roundy, S., Wright, P.K., 2002, Anisotropic material properties of fused deposition modeling ABS. Rapid Prototyp J, 8/4: 248–257.
2. Wang, C.C., Lin, T.W., Hu, S.S., 2007, Optimizing the rapid prototyping process by integrating the Taguchi method with the gray relational analysis. Rapid Prototyp J, 13/5: 304–315.
3. Equbal, A., Sood, A.K., Ohdar, R.K., Mahapatra, S.S., 2011, Prediction of magnitude preciseness in fused deposition modelling: a fuzzy logic approach. Int J Prod Qual Manage, 7:22–43.
4. Yadav, D., Chhabra, D., Garg, R.K., Ahlawat, A., Phogat, A., 2019, Optimization of FDM3Dprinting process parameters for multi-material using artificial neural network. Mater Today Proc, . http://dx.doi.org/10.1016/j.matpr.2019.11.225.
5. Yadav, D., Chhabra, D., Garg, R.K., Ahlawat, A., Phogat, A., 2019, Modeling and analysis of significant process parameters of FDM 3D printer using ANFIS. Mater Today Proc, . http://dx.doi.org/10.1016/j.matpr.2019.11.227.
6. M.S. Saad, A.M. Nor, M.E. Baharudin, M.Z. Zakaria, and A.F. Aiman, Optimization of Surface Roughness in FDM 3D Printer Using Response Surface Methodology, Particle Swarm Optimization, and Symbiotic Organism Search Algorithms, Int. J. Adv. Manuf. Technol, 2019, 105, p 5121–5137
7. Deshwal, S., Kumar, A., & Chhabra, D. (2020). Exercising hybrid statistical tools GA-RSM, GA-ANN and GA-ANFIS to optimize FDM process parameters for tensile strength improvement. *CIRP Journal of Manufacturing Science and Technology*, *31*, 189-199.
8. Ziadia, A., Habibi, M., & Kelouwani, S. (2023). Machine learning study of the effect of process parameters on tensile strength of FFF PLA and PLA-CF. *Eng*, *4*(4), 2741-2763.
9. Zhang Z, Liu Z, Wu D. Prediction of melt pool temperature in directed energy deposition using machine learning. Addit. Manuf. 2021;37:101692.
10. Singh A, Cooper D, Blundell N, Gibbons G, Pratihar D. Modelling of direct metal laser sintering of EOS DM20 bronze using neural networks and genetic algorithms. In: Proceedings of the 37th international MATADOR conference; 2012. p. 395e8. Manchester, UK.
11. Caiazzo, F., & Caggiano, A. (2018). Laser direct metal deposition of 2024 Al alloy: trace geometry prediction via machine learning. *Materials*, *11*(3), 444.
12. Tapia G, Khairallah S, Matthews M, King WE, Elwany A. Gaussian process-based surrogate modeling frame work for process planning in laser powder-bed fusion additive manufacturing of 316L stainless steel. Int J Adv Manuf Technol 2017;94:3591e603.
13. Sridhar, S., Venkatesh, K., Revathy, G., Venkatesan, M., & Venkatraman, R. (2024). Adaptive fabrication of material extrusion-AM process using machine learning algorithms for print process optimization. *Journal of Intelligent Manufacturing*, 1-25.
14. Mishra, A., Jatti, V. S., & Paliwal, S. (2023). Evolutionary AI-Based Algorithms for the Optimization of the Tensile Strength of Additively Manufactured Specimens. In *Machine Intelligence for Smart Applications: Opportunities and Risks* (pp. 195-211). Cham: Springer Nature Switzerland.
15. Ulkir, O., & Akgun, G. (2024). Prediction of Flexural Strength with Fuzzy Logic Approach for Fused Deposition Modeling of Polyethylene Terephthalate Glycol Components. *Journal of Materials Engineering and Performance*, *33*(9), 4367-4376.

APPENDIX#1

A: Linear Regression Algorithm

- **Input:** Dataset with 4 input variables (X_1, X_2, X_3, X_4) and one output variable (y).

- **Initialize**: Parameters (weights w_1, w_2, w_3, w_4, and bias b) to zero or small random values.
- **Hypothesis**: Predicted value ($\hat{y}$) is calculated as:
$$\hat{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4$$
- **Loss Function**: Use Mean Squared Error (MSE) to measure the error:
$$MSE = (1/n) * \Sigma (\hat{y}_i - y_i)^2$$
where n is the number of samples.
- **Gradient Descent**:
- Compute the gradients for each weight (w_1, w_2, w_3, w_4) and the bias (b):
$$\partial MSE/\partial w\square = -(2/n) * \Sigma X\square_i(\hat{y}_i - y_i), forj = 1,2,3,4$$
$$\partial MSE/\partial b = -(2/n) * \Sigma(\hat{y}_i - y_i)$$
- Update the parameters w_1, w_2, w_3, w_4, and b:
$$w\square = w\square - \alpha * (\partial MSE/\partial w\square), forj = 1,2,3,4$$
$$b = b - \alpha * (\partial MSE/\partial b)$$
where α is the learning rate.
- **Repeat**: Steps 3–5 for a set number of iterations or until the model converges.
- **Output**: Final values of w_1, w_2, w_3, w_4, and b (the trained model).

B: Decision Tree Regressor Algorithm

- **Input**: Dataset with 4 input variables X (where X=[X1,X2,X3,X4]) and one output variable y.
- **Initialize**:
  - Set the maximum depth of the tree $d_{\max}$ (optional).
  - Set a minimum number of samples per leaf nminn_{\text{min}}nmin.
- **Tree Construction (Recursive Splitting)**:

For each node:

  - **Stopping Criteria**:
    - If all samples in the node belong to the same target value or the node contains fewer than nminn_{\text{min}}nmin samples, stop splitting and make this a leaf node.
    - Compute the predicted value for this leaf as:
$$\hat{y} = \frac{1}{m}\sum_{i=1}^{m} y_i$$
where m is the number of samples in the node.
  - **Otherwise**, for each feature jjj (where j=1,2,3,4j = 1, 2, 3, 4j=1,2,3,4):
    - For each possible split point s of feature Xj:
      - Split the data into two subsets:

Left subset: $L = \{i: X_{ij} \leq s\}$

Right subset:
$$R = \{i: X_{ij} > s\}$$

      - Calculate the **Mean Squared Error (MSE)** for the split:
$$\text{MSE} = \frac{1}{|L|}\sum_{i\in L}(y_i - \widehat{y_L})^2 + \frac{1}{|R|}\sum_{i\in R}(y_i - \widehat{y_R})^2$$

where:
$$\widehat{y_L} = \frac{1}{|L|}\sum_{i\in L} y_i$$
$$\widehat{y_R} = \frac{1}{|R|}\sum_{i\in R} y_i$$

$$\widehat{y_{\text{new}}} = \frac{1}{m}\sum_{i\in\text{leaf}} y_i$$

- Choose the feature jjj and the split point sss that results in the **lowest MSE**.
- **Split** the data into two child nodes based on the best split found:
  - Left child: data where $Xj \leq s$
  - Right child: data where $Xj > s$

Recursively repeat the splitting process for each child node (left and right) until the stopping criteria are met.

- **Prediction**:
  - To predict for a new input $Xnew = [X1, new, X2, new, X3, new, X4, new]$The predicted value is the mean of the target values yyy in the leaf node:

$$\widehat{y_{\text{new}}} = \frac{1}{m}\sum_{i\in\text{leaf}} y_i$$

- **Output**: A decision tree model that can be used to predict the target variable yyy for new input data.

C: Random Forest Regressor Algorithm

Input: Dataset with 4 input variables XXX (where X=[X1, X2, X3, X4] and one output variable y.

Initialize:

Set the number of trees n_{trees}.

Set the maximum depth of each tree $d_{\max}$ (optional).

Set a minimum number of samples per leaf $n_{\min}$.

Construct Forest: For each tree ttt from 1 to n_{trees}:

Bootstrap Sampling:

Create a bootstrap sample S_t by randomly selecting mmm samples (with replacement) from the original dataset.

Tree Construction:

Build a decision tree T_t using the bootstrap sample StS_tSt:

For each node in the tree:

Stopping Criteria:

If the node contains fewer than $n_{\min}$ samples or if the maximum depth $d_{\max}$ is reached, stop splitting and make this a leaf node.

Compute the predicted value for this leaf as:

$$\hat{y} = \frac{1}{m}\sum_{i=1}^{m} y_i$$

where mmm is the number of samples in the node.

Otherwise, for each feature j (where j=1,2,3,4):

Randomly select m_{features} from the total features.

For each feature j in the selected features:

For each possible split point s:

Split the data into left and right subsets.

$$\widehat{y_L} = \frac{1}{|L|}\sum_{i\in L} y_i$$

$$\widehat{y_R} = \frac{1}{|R|}\sum_{i\in R} y_i$$

Calculate the **Mean Squared Error (MSE)** for the split.

$$\mathrm{MSE} = \frac{1}{|L|}\sum_{i \in L}(y_i - \widehat{y_L})^2 + \frac{1}{|R|}\sum_{i \in R}(y_i - \widehat{y_R})^2$$

Choose the feature and the split point that result in the lowest MSE.
Split the data based on the best split found.
Prediction: To predict for a new input Xnew=[X1,$_{\text{new}}$,X2,$_{\text{new}}$,X3,$_{\text{new}}$,X4,$_{\text{new}}$]
Obtain predictions from each tree T_t.
The final prediction $\hat{y}new$ is the average of the predictions from all trees:

$$\widehat{y_{\text{new}}} = \frac{1}{n_{\text{trees}}}\sum_{t=1}^{n_{\text{trees}}} T_t(X_{\text{new}})$$

Output: A random forest model that can be used to predict the target variable yyy for new input data.

D: Support Vector Regressor (SVR) Algorithm

- **Input**: Dataset with 4 input variables X (where X=[X1,X2,X3,X4]) and one output variable y.
- **Initialize**:
  - Choose a kernel function K (e.g., linear, polynomial, radial basis function).
  - Set the regularization parameter C.
  - Set the epsilon ϵ that defines the epsilon-tube within which no penalty is associated in the training loss function.
- **Train the Model**:
  - Formulate the optimization problem as follows:
  - Minimize the objective function:

$$\min_{w,b,\xi_i} \quad \frac{1}{2}|w|^2 + C\sum_{i=1}^{N} \xi_i$$

subject to the constraints:

$$y_i - w^T \phi(X_i) - b \leq \epsilon + \xi_i$$

where:

  - w is the weight vector,
  - b is the bias,
  - ξi are the slack variables for the error,
  - ϕ(Xi) is the feature mapping.
- **Optimization**:
  - Use Sequential Minimal Optimization (SMO) and find w and b.
- **Prediction**:
  - To predict the output y^ for a new input Xnew=[X1,$_{\text{new}}$,X2,$_{\text{new}}$,X3,$_{\text{new}}$,X4,$_{\text{new}}$]
  - $\hat{y} = w^T \phi(X_{\text{new}}) + b$
  - If using the kernel trick:

$$\hat{y} = \sum_{i=1}^{N} \alpha_i K(X_{\text{new}}, X_i) + b$$

where αi are the Lagrange multipliers obtained from the optimization.

- **Output**: A Support Vector Regression model that can be used to predict the target variable yyy for new input data.

E: K-Nearest Neighbors (KNN) Regressor Algorithm

- **Input**:
    - Dataset with 4 input variables X (where X=[X1,X2,X3,X4] and one output variable y.
    - Specify the number of neighbors k.
- **Data Preprocessing:**
    - Normalize or standardize the input features X if necessary to ensure all features contribute equally to the distance calculations.
- **Distance Calculation**:
    - For a new input Xnew=[X1,$_{\text{new}}$,X2,$_{\text{new}}$,X3,$_{\text{new}}$,X4,$_{\text{new}}$]
    - to each training sample Xi. Common distance metrics include:
        - **Euclidean Distance**:

$$d(X_{\text{new}}, X_i) = \sqrt{\sum_{j=1}^{4} \left(X_{j,\text{new}} - X_{j,i}\right)^2}$$

        - **Manhattan Distance**:

$$d(X_{\text{new}}, X_i) = \sum_{j=1}^{4} \left|X_{j,\text{new}} - X_{j,i}\right|$$

- **Find Neighbors**:
    - Identify the k training samples closest to Xnew based on the calculated distances.
- **Prediction**:
    - Calculate the predicted output y^\hat{y}y^ as the mean of the output values y of the k nearest neighbors:

$$\hat{y} = \frac{1}{k} \sum_{j=1}^{k} y_j$$

- **Output**: A KNN regression model that can predict the target variable y for new input data.