processes

MDPI

*Article*

# A Method for Predicting Surface Finish of Polylactic Acid Parts Printed Using Fused Deposition Modeling

**Meifa Huang, Shangkun Jin, Zhemin Tang, Yuanqing Chen and Yuchu Qin ***

Guangxi Key Laboratory of Manufacturing System and Advanced Manufacturing Technology, School of Mechanical and Electrical Engineering, Guilin University of Electronic Technology, Guilin 541004, China; hmhmf@guet.edu.cn (M.H.); 21012201020@mails.guet.edu.cn (S.J.); tzmaql@guet.edu.cn (Z.T.); 20012302102@mails.guet.edu.cn (Y.C.)
* Correspondence: yuchu.qin@guet.edu.cn

**Abstract:** Accurately predicting the surface finish of fused deposition modeling (FDM) parts is an important task for the engineering application of FDM technology. So far, many prediction models have been proposed by establishing a mapping relationship between printing parameters and surface roughness. Each model can work well in its specific context; however, existing prediction models cannot meet the requirements of multi-factor and multi-category prediction of surface finish and cope with imbalanced data. Aiming at these issues, a prediction method based on a combination of the adaptive particle swarm optimization and K-nearest neighbor (APSO-KNN) algorithms is proposed in this paper. Seven input variables, including nozzle diameter, layer thickness, number of perimeters, flow rate, print speed, nozzle temperature, and build orientation, are considered. The printing values of each specimen are determined using an L27 Taguchi experimental design. A total of 27 specimens are printed and experimental data for the 27 specimens are used for model training and validation. The results indicate that the proposed method can achieve a minimum classification error of 0.01 after two iterations, with a maximum accuracy of 99.0%, and high model training efficiency. It can meet the requirements of predicting surface finish for FDM parts with multiple factors and categories and can handle imbalanced data. In addition, the high accuracy demonstrates the potential of this method for predicting surface finish, and its application in actual industrial manufacturing.

**Keywords:** fused deposition modeling; surface finish; adaptive particle swarm optimization algorithm; K-nearest neighbor algorithm; multi-category prediction



**Citation:** Huang, M.; Jin, S.; Tang, Z.; Chen, Y.; Qin, Y. A Method for Predicting Surface Finish of Polylactic Acid Parts Printed Using Fused Deposition Modeling. *Processes* **2023**, *11*, 1820. https://doi.org/10.3390/pr11061820

Academic Editor: Juan García Rodríguez

Received: 12 May 2023
Revised: 5 June 2023
Accepted: 12 June 2023
Published: 15 June 2023



## 1. Introduction

Currently, additive manufacturing (AM) technologies are increasingly being used in various fields [1–4]. Compared with traditional manufacturing technologies [5], they improve production efficiency, reduce manufacturing costs, and enable personalized production, further promoting the progress of Industry 4.0. Fused deposition modeling (FDM) is a widely used AM technology due to its low manufacturing cost, fast molding speed, and simple process [6]. However, the poor surface finish of printed parts limits their application in practical engineering [7]. For example, when using FDM technology to manufacture molds in casting and injection molding processes, or when printed parts are in contact with human skin, surface finish is crucial and can affect the product's performance to meet its assembly and functional requirements [8,9]. Accurately predicting the surface finish of printed parts is one of the important tasks of FDM technology in engineering applications [10].

In recent years, the influence of FDM printing parameter settings on surface finish, including layer thickness, temperature, print speed, flow rate, build orientation, number of perimeters, etc., has been reported [11–15]. Layer thickness refers to the thickness of each layer printed by the printer nozzle along the Z-axis, which affects the details and

printing speed of the finished product. Print speed refers to the moving speed of the printer nozzle, which affects the printing time and quality of the finished product. Wall thickness refers to the thickness of the printed model shell after the removal of the infill, which can also be understood as the number of times the nozzle passes over the outer layer, or the number of perimeters. Print temperature determines the melting point and fluidity of the printing material. Flow rate compensation, also known as flow rate, mainly corrects errors in the actual extrusion amount caused by internal factors. Build orientation is the molding direction of the part and is closely related to the surface shape of the part. Currently, mainstream prediction methods aim to establish a mapping relationship between part printing parameters and part surface roughness, with printing parameters as input and predicted roughness parameters as output. Researching different mapping relationship models to improve prediction accuracy is an important direction in the prediction of surface smoothness of FDM-printed parts.

Some researchers attempted to establish models for fitting the contours of printed parts. Taufik et al. [16], Buj-Corral et al. [17], Xu et al. [18], Perez et al. [19], and Pandey et al. [20] constructed geometric models of surface contours to predict surface roughness (Ra) by studying the relationship between surface contours of different types of FDM-printed parts and printing parameters. Other researchers, such as Vahabli et al. [21], Wu et al. [22], Boschetto et al. [23], and Reddy et al. [24], generated models using machine-learning-based algorithms. The common features of these studies are threefold: First, they use commonly used printing materials such as ABS or PC. Second, the main factors considered in their models are only layer thickness and build orientation. Third, the output of their models is the surface roughness values, obtained by solving a regression problem.

Predicting the surface roughness of a part may be meaningful for certain purposes, such as accurately modeling the surfaces of printed parts. However, in actual engineering manufacturing, it is only necessary to know whether the surface finish level of the part meets the required manufacturing requirements (classification problem) [25]. In fact, the standards used until a few years ago in the industry used classes when defining surface roughness [26]. For example, N11 is a class that corresponds to a surface roughness between 12.5 and 25 μm; N12 is a class that corresponds to a surface roughness between 25 and 50 μm. Studies can be found in the literature that use classification algorithms to determine the surface finish levels of parts manufactured by other printing technologies [27]. Molero et al. [28], Barrios et al. [29], and Cerro et al. [30] compared different machine learning algorithms to verify their performance by dividing the range of the surface roughness dataset of parts into different categories. The input factors, data types, and number of classifications used by each model are shown in Table 1.

**Table 1.** Summary of current classification models.

| Authors | Method | Data Balance | Factor | Classification |
|---|---|---|---|---|
| Molero et al. [28] | ML | √ | LT, T, PS, PA, F | 2 |
| Barrios et al. [29] | DT | √ | LT, T, PS, PA, F | 2 |
| Cerro et al. [30] | ML | √ | LT, PS, NP, T, BO | 2 |

Note: ML stands for machine learning; DT stands for decision tree; Layer thickness (LT), mm; Temperature (T), °C; Print speed (PS), mm/s; Print acceleration (PA), $mm/s^2$; Flow rate (F), %; Build orientation, (BO), °; Number of perimeters (NP); "√" represents data balance.

Each of the existing classification models can work well in its specific context. However, they still have the following shortcomings: first, they do not consider imbalance in the data, resulting in low prediction accuracy for categories with fewer individuals. Second, they consider limited types of input factors. In the literature, you can find previous work in which the influence of different printing parameters on the surface roughness of 3D printed parts has been studied (Table 2). Third, they cannot meet the demand for multi-factor and multi-category prediction of surface finish. Category expansion will reduce prediction accuracy and the model will be complex, difficult to tune, and difficult to understand, making it unsuitable for practical engineering applications.

**Table 2.** Previous studies on the surface roughness of FDM PLA parts.

| Authors | Layer Thickness | Temperature | Print Orientation | Print Speed | Nozzle Diameter | Filling Density | Wall Thickness |
|---|---|---|---|---|---|---|---|
| García Plaza et al. [31] | √ | – | √ | √ | – | – | – |
| Taşcıoğlu et al. [32] | √ | √ | – | √ | – | – | – |
| Ramli et al. [33] | √ | – | – | – | – | √ | – |
| Alsoufi et al. [34] | √ | – | – | – | √ | – | – |
| Cerro et al. [30] | √ | √ | √ | √ | – | – | √ |

Note: "√" represents that the study involves this factor, while "–" represents that the study don't involve this factor.

The K-nearest neighbor algorithm (KNN) is simple, robust, and highly interpretable, and is suitable for multi-classification, nonlinear, and imbalanced data problems [35]. The basic principle of the K-nearest neighbor algorithm is to classify the *k*-closest labels to the target distance, where distance is the main factor affecting classification accuracy. However, different weights of the input variables inevitably lead to biases in distance calculation and classification errors. In [36–38], the particle swarm optimization (PSO) algorithm was used to optimize the data features and the number of neighbors of the KNN algorithm, which was applied to specific tasks such as perception, recognition, and diagnosis, and achieved good performance.

In the population-based optimization algorithms, the PSO algorithm has the advantages of simple principles, few adjustable parameters, low computational complexity, and strong global optimization capability [39]. The optimization method has been widely used in various fields and effectively solved corresponding problems in specific fields [40–42]. However, the fast convergence speed of the PSO algorithm can easily cause the parameter search to fall into a local optimum, leading to premature convergence. To address this issue, we use clustering to adaptively divide the particle swarm into different populations and guide the populations by applying different update strategies [43]. This enhances the diversity of particles and helps particles jump out of a local optimum.

Based on the above considerations, this paper introduces the adaptive particle swarm optimization algorithm into the K-nearest neighbor classification algorithm and proposes a method for predicting the surface finish of FDM PLA parts based on this. By introducing the adaptive particle swarm optimization algorithm into the distance formula of the K-nearest neighbor algorithm, random weight values are generated for each input variable to adjust the distance calculation deviation. This method reduces the impact of data imbalance on the prediction results and meets the needs of multi-factor and multi-category prediction of surface finish in engineering practice, providing a powerful reference for the surface finish of FDM-printed parts in engineering manufacturing.

The remainder of this paper is organized as follows: Section 2 explains the details of the proposed method. Section 3 presents the experimental design, dataset, and model framework. Section 4 shows the results and discussions. Finally, Section 5 provides a conclusion, including future directions for further work.

## 2. Materials and Methods

This method introduces the adaptive particle swarm optimization algorithm into the distance formula of the K-nearest neighbor algorithm, generating optimal weights for each input variable to adjust the calculation bias of the distance formula. Its framework diagram is shown in Figure 1. The principles are analyzed in detail below.

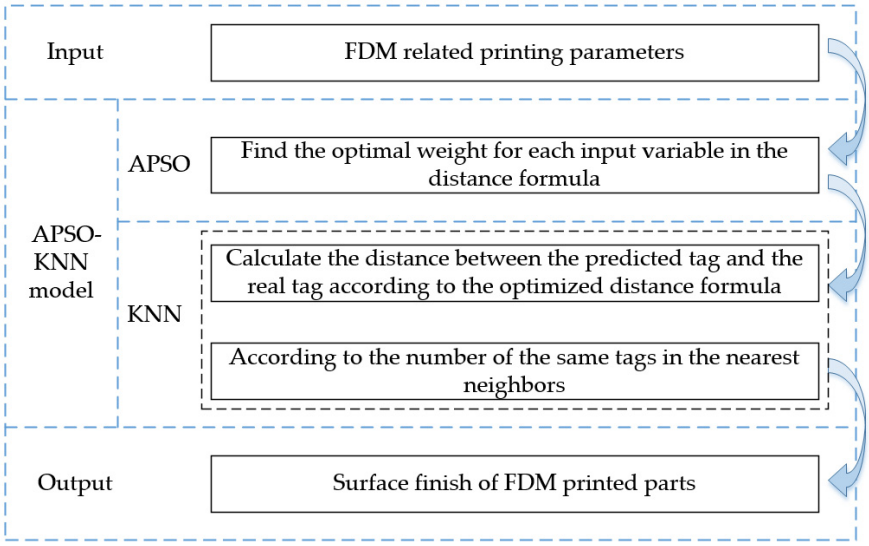



**Figure 1.** Framework of the method proposed in this paper.

### *2.1. K-Nearest Neighbors*

The K-nearest Neighbor (KNN) algorithm is an instance-based machine learning algorithm and one of the simplest and least parameterized classification algorithms [44]. The KNN algorithm is simple, robust, and highly interpretable, and is suitable for multi-classification, nonlinear, and imbalanced data problems. The idea behind the KNN algorithm is to find the $k$ closest labels to the label to be classified in the training set, and to determine the classification of the label to be classified by counting the categories of these $k$ labels [45]. Distance is the main factor affecting classification accuracy.

The neighbors of a label can be defined based on the Euclidean distance, assuming all labels are in $m$-dimensional space, and any set $x$ can be represented as a feature vector $x = (x_1, x_2, \cdots, x_m)$, where $x_i^l$ represents the feature value of the $i$-th example. The distance between example $x_i$ and set $x_j$ is defined as the Euclidean distance $d(x_i,x_j)$, as shown in the following formula.

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{m} (x_i^l - x_j^l)^2} \tag{1}$$

However, different weights of the input variables inevitably lead to biases in distance calculation and classification errors. To address this issue, a weight $\alpha$ is assigned to each feature value used in the distance formula, adjusting the calculation bias. The above formula can be written as:

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{m} \left(\alpha \left(x_i^l - x_j^l\right)\right)^2} \tag{2}$$

### *2.2. Adaptive Particle Swarm Optimization*

Adaptive Particle Swarm Optimization (APSO) is an evolutionary algorithm [46,47]. Based on the Basic Particle Swarm Optimization algorithm, APSO introduces adaptive mechanisms and polynomial mutation strategies. The updated rules for particle velocity and position are the same as those in the Basic Particle Swarm Optimization algorithm, but the algorithm dynamically adjusts the inertia weights by introducing an adaptive mechanism to improve the search efficiency and convergence performance and avoid getting trapped in local optima.

The updated rules for particle velocity and position are as follows:

$$\begin{cases} x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \quad j = 1, 2, 3, \cdots, d \\ v_{i,j}(t+1) = \omega \cdot v_{i,j}(t) + c_1 r_1 [p_{i,j} - x_{i,j}(t)] + c_2 r_2 [p_{g,j} - x_{i,j}(t)] \end{cases} \tag{3}$$

In Equation (3), $x_{i,j}$ and $v_{i,j}$ represent the numerical values of particle displacement and velocity; $\omega$ is the inertia weight; $c_1$ and $c_2$ are the learning factors; $r_1$ and $r_2$ are uniformly distributed random numbers within [0, 1]; $p_{i,j}$ is the individual best value of the particle; $p_{g,j}$ is the global best value of the particle.

The adaptive mechanism is expressed as follows:

$$\omega = \begin{cases} \omega_{min} - \frac{(\omega_{max}-\omega_{min})\times(f-f_{min})}{f_{avg}-f_{min}} & f \le f_{avg} \\ \omega_{max} & f > f_{avg} \end{cases} \quad (4)$$

In Equation (4), $f$ represents the real-time objective function value of the particle; $f_{avg}$ and $f_{min}$ represent the average value and the minimum target value of all particles at present; $\omega_{max}$ and $\omega_{min}$ represent the maximum and minimum values of the inertia weight, and the value range is 0.4–0.8.

### 2.3. APSO-KNN

In this study, the APSO algorithm is incorporated into the KNN algorithm to search the weight of each feature value in Formula (2). Additionally, the selection of fitness function in APSO is crucial. The KNN classification accuracy function is chosen as the fitness function.

$$acc = \frac{NC}{NT} \quad (5)$$

In Equation (5), *acc* represents the accuracy, and its value range is 0–1; *NC* represents the number of correct classifications; *NT* represents the total number of classification labels.

By learning the weight value "$\alpha$", the fitness function value can be maximized to the greatest extent possible. The workflow of the APSO-KNN algorithm is shown in Figure 2.

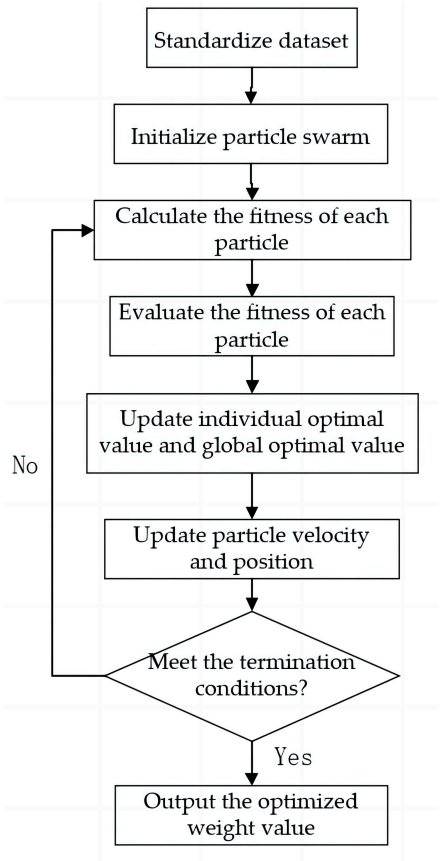



**Figure 2.** Flowchart of the APSO-KNN algorithm.

Step 1: Standardize the dataset. Standardizing the dataset can reduce the differences between certain attributes and decrease the errors in the classification accuracy of the dataset.

Step 2: Initialize particle swarm. Using the weight $\alpha$ for each feature value as the optimization variable, the position and velocity of each particle in the population are randomly initialized.

Step 3: Calculate the distance between each label and the predicted label vector using Formula (2) and calculate the fitness value using Formula (5).

Step 4: Evaluate the fitness of each particle, store the particle's position and fitness value in its personal best $p_{best}$, and save the position and fitness value of the best individual from $p_{best}$ in the global best $g_{best}$.

Step 5: Update each particle using the updated Formulas (3) and (4) and evaluate the fitness of each particle.

Step 6: Compare the fitness value of each particle with its personal best position, and if they are close, update the personal best position with the current value. Compare all $p_{best}$ and $g_{best}$, and update $g_{best}$.

Step 7: Stop the search and output the results if the termination condition is met; otherwise, return to Step 5 to continue the search.

## 3. Experiments

An industrial-grade FDM BM345 (Blue Maker, China) printer (Figure 3) was used for 3D printing. It is equipped with nozzles with diameters of 0.4 mm, 0.6 mm, and 0.8 mm, and the maximum molding size is 350 × 350 × 450 mm. The recommended printing speed is 60 mm/s, with a maximum of 150 mm/s. A blower fan is installed next to the extruder for accelerated cooling of the printed object. The recommended slicing software is BlueMaker3D (version 18.10), which allows the use of STL and G-code standards.

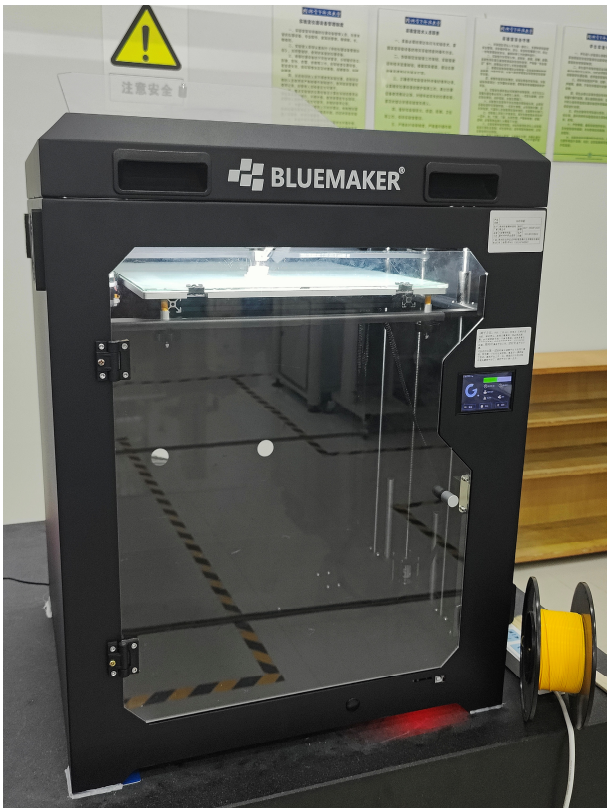


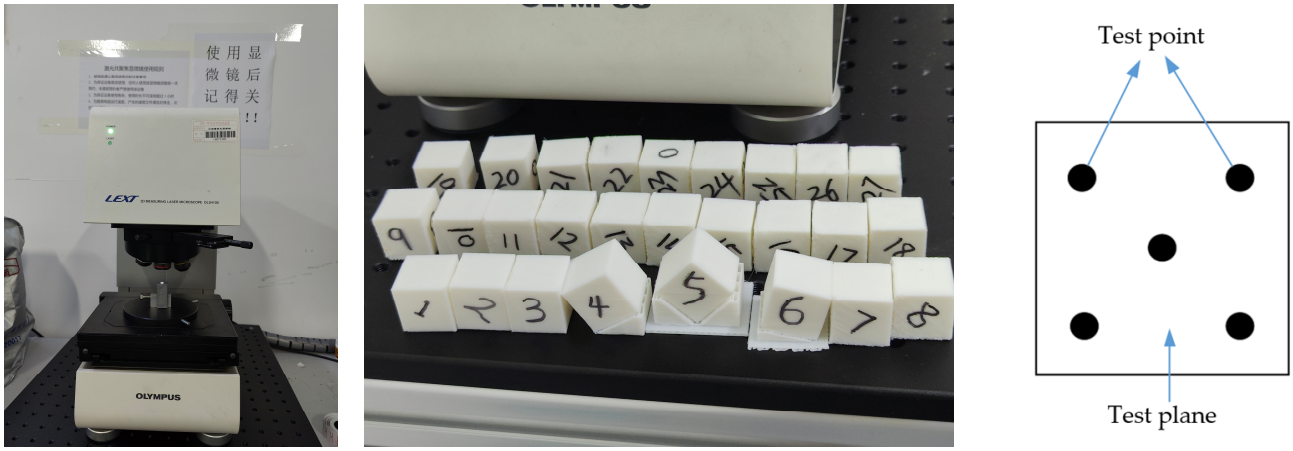

**Figure 3.** BM345 printer and PLA filament.

Polylactic Acid (PLA) was used as the base material. Its main properties are density, temperature, diameter, and color [48]. These properties have an impact on the FDM printing parameter settings, as well as the quality and strength of printed parts. The specific parameters are shown in Table 3. The material is provided by Smart Materials 3D company (Alcalá la Real, Spain) with a diameter of 1.75 mm and a diameter variation range of ±0.03 mm. The material has stable performance and is expected not to warp.

**Table 3.** Material properties.

| Chemical Name | Composition | Density ($g/mm^3$) | Printing Temperature (°C) | Diameter (mm) | Color |
|---|---|---|---|---|---|
| Polylactic Acid | PLA (Polylactide Resin) 99% | 1.24 | 220 ± 20 | 1.75 ± 0.03 | white |

Surface roughness measurement is defined in ISO 21920-2:2021 [49] and has different parameters. Ra is the most commonly used parameter in determining the surface texture state, as it provides the most general and adequate information about surface roughness. The surface shapes of 27 samples were observed and the surface roughness (Ra) was measured using OLS40-SU laser confocal microscope (Figure 4a). It can magnify at a range of 108×–17,280×, with a display resolution of 1 nm. The highest precision of the X/Y axis is 0.12 μm, and the optimal measurement range is 1 μm–1.5 mm. The highest precision of the Z axis is 0.01 μm, and the optimal measurement range is 0.5 μm–3 mm. The roughness measurement range is above 0.1 μm, and it has a built-in Gaussian filter. The objective lens used is 10×. The samples shown in Figure 4b were measured using this technique. Each sample was placed horizontally with the surface to be measured facing upwards, and five positions were taken on the plane, four on the edge and one in the center (Figure 4c). The surface roughness value was the average roughness measured at the five points.




(**a**) (**b**) (**c**)

**Figure 4.** (**a**) Surface roughness measuring instrument; (**b**) Experimental printing samples; (**c**) Point method.

The Taguchi L27 experimental design [50] was used in this study to print 27 specimens with dimensions of 20 × 20 × 20 mm (Figure 4b), using 7 factors with 3 levels (Table 4). The factors studied were nozzle diameter (ND), layer thickness (LT), number of perimeters (NP), flow rate (F), print speed (PS), nozzle temperature (T), and build orientation (BO), with the number of perimeters associated with the layer width. Table 5 shows the settings for each parameter.

**Table 4.** Factors and levels in the design of experiments (DOE).

| Factors | Level 1 | Level 2 | Level 3 |
|---|---|---|---|
| Nozzle diameter, ND (mm) | 0.40 | 0.60 | 0.80 |
| Layer thickness, LT (mm) | 0.10 | 0.25 | 0.35 |
| Number of perimeters, NP | 2 | 3 | 4 |
| Flow rate, F (%) | 90 | 100 | 110 |
| Printing speed, PS (mm/s) | 40 | 60 | 80 |
| Temperature, T (°C) | 190 | 210 | 230 |
| Build orientation, BO (°) | 30 | 45 | 75 |

**Table 5.** Taguchi L27 experimental design.

| No. | ND (mm) | LT (mm) | NP | F (%) | PS (mm/s) | T (°C) | BO (°) |
|---|---|---|---|---|---|---|---|
| 1 | 0.40 | 0.10 | 2 | 90 | 40 | 190 | 30 |
| 2 | 0.40 | 0.10 | 2 | 90 | 60 | 210 | 45 |
| 3 | 0.40 | 0.10 | 2 | 90 | 80 | 230 | 75 |
| 4 | 0.40 | 0.25 | 3 | 100 | 40 | 190 | 30 |
| 5 | 0.40 | 0.25 | 3 | 100 | 60 | 210 | 45 |
| 6 | 0.40 | 0.25 | 3 | 100 | 80 | 230 | 75 |
| 7 | 0.40 | 0.35 | 4 | 110 | 40 | 190 | 30 |
| 8 | 0.40 | 0.35 | 4 | 110 | 60 | 210 | 45 |
| 9 | 0.40 | 0.35 | 4 | 110 | 80 | 230 | 75 |
| 10 | 0.60 | 0.10 | 3 | 110 | 40 | 210 | 75 |
| 11 | 0.60 | 0.10 | 3 | 110 | 60 | 230 | 30 |
| 12 | 0.60 | 0.10 | 3 | 110 | 80 | 190 | 45 |
| 13 | 0.60 | 0.25 | 4 | 90 | 40 | 210 | 75 |
| 14 | 0.60 | 0.25 | 4 | 90 | 60 | 230 | 30 |
| 15 | 0.60 | 0.25 | 4 | 90 | 80 | 190 | 45 |
| 16 | 0.60 | 0.35 | 2 | 100 | 40 | 210 | 75 |
| 17 | 0.60 | 0.35 | 2 | 100 | 60 | 230 | 30 |
| 18 | 0.60 | 0.35 | 2 | 100 | 80 | 190 | 45 |
| 19 | 0.80 | 0.10 | 4 | 100 | 40 | 230 | 45 |
| 20 | 0.80 | 0.10 | 4 | 100 | 60 | 190 | 75 |
| 21 | 0.80 | 0.10 | 4 | 100 | 80 | 210 | 30 |
| 22 | 0.80 | 0.25 | 2 | 110 | 40 | 230 | 45 |
| 23 | 0.80 | 0.25 | 2 | 110 | 60 | 190 | 75 |
| 24 | 0.80 | 0.25 | 2 | 110 | 80 | 210 | 30 |
| 25 | 0.80 | 0.35 | 3 | 90 | 40 | 230 | 45 |
| 26 | 0.80 | 0.35 | 3 | 90 | 60 | 190 | 75 |
| 27 | 0.80 | 0.35 | 3 | 90 | 80 | 210 | 30 |

## 4. Results and Discussion

### *4.1. Results of Surface Roughness Measurements*

Following the measurement method shown in Figure 4c, the surface roughness (Ra) values were measured at five points, and the average value was calculated. However, due to the large variation in roughness values at different measurement points on the same plane, the results had high variability. This variability made it difficult to draw conclusions about surface roughness based on measurement location. Calculating the standard deviations for all test points can clearly show the measurement performance. The standard deviations (SD) and average values are listed in Table 6 in units of μm.

**Table 6.** Results of surface roughness measurements.

| Test | $Ra_1$ (μm) | $Ra_2$ (μm) | $Ra_3$ (μm) | $Ra_4$ (μm) | $Ra_5$ (μm) | Ra (μm) | SD (μm) |
|---|---|---|---|---|---|---|---|
| 1 | 15.43 | 14.24 | 16.50 | 18.32 | 15.22 | 15.94 | 1.55 |
| 2 | 16.93 | 12.60 | 12.63 | 15.21 | 15.50 | 14.58 | 1.90 |
| 3 | 10.90 | 10.67 | 9.89 | 10.72 | 11.73 | 10.78 | 0.65 |
| 4 | 44.53 | 44.61 | 44.25 | 45.79 | 44.11 | 44.66 | 0.66 |
| 5 | 34.19 | 31.29 | 32.69 | 34.22 | 29.11 | 32.30 | 2.15 |
| 6 | 24.58 | 24.91 | 26.7 | 28.17 | 21.46 | 25.17 | 2.52 |
| 7 | 52.91 | 49.05 | 54.76 | 51.82 | 48.08 | 51.32 | 2.75 |
| 8 | 43.75 | 41.16 | 43.25 | 48.18 | 41.05 | 43.48 | 2.89 |
| 9 | 37.76 | 34.28 | 35.22 | 35.29 | 36.61 | 35.83 | 1.35 |
| 10 | 9.03 | 8.92 | 7.97 | 9.88 | 7.98 | 8.76 | 0.80 |
| 11 | 18.74 | 14.19 | 15.40 | 19.86 | 14.64 | 16.57 | 2.56 |
| 12 | 11.23 | 12.48 | 13.81 | 10.90 | 11.56 | 11.99 | 1.17 |
| 13 | 25.71 | 21.30 | 21.67 | 23.15 | 24.35 | 23.24 | 1.84 |
| 14 | 37.56 | 43.28 | 42.84 | 43.96 | 39.59 | 41.45 | 2.74 |
| 15 | 34.02 | 29.64 | 32.96 | 30.23 | 31.59 | 31.69 | 1.83 |
| 16 | 31.66 | 30.21 | 31.80 | 28.21 | 32.16 | 30.81 | 1.62 |
| 17 | 52.47 | 50.76 | 53.15 | 49.82 | 52.84 | 51.81 | 1.44 |
| 18 | 38.20 | 37.42 | 39.64 | 43.37 | 41.49 | 40.02 | 2.43 |
| 19 | 18.01 | 17.81 | 16.59 | 16.67 | 16.07 | 17.03 | 0.83 |
| 20 | 7.30 | 8.11 | 7.43 | 8.83 | 7.41 | 7.81 | 0.65 |
| 21 | 24.90 | 22.98 | 23.81 | 25.22 | 24.16 | 24.21 | 0.89 |
| 22 | 29.83 | 28.42 | 30.70 | 29.89 | 28.89 | 29.55 | 0.90 |
| 23 | 20.02 | 16.33 | 20.91 | 21.98 | 22.95 | 20.44 | 2.55 |
| 24 | 33.60 | 33.57 | 34.31 | 33.90 | 34.87 | 34.05 | 0.54 |
| 25 | 41.54 | 40.78 | 41.91 | 41.96 | 40.38 | 41.31 | 0.70 |
| 26 | 30.42 | 32.67 | 29.87 | 30.30 | 28.26 | 30.30 | 1.57 |
| 27 | 59.04 | 58.72 | 55.41 | 52.35 | 52.88 | 55.68 | 3.14 |

### *4.2. Experimental Data Classification*

From the experimental data, it can be seen that the range of variation in Ra mean values is 7–55 μm. In order to achieve surface finish classification, the values of the arithmetic mean deviation Ra of the contours were divided into four levels according to the standard (ISO 1302:2002 [26]), namely N10, N11, N12, and N13, as shown in Table 7. Then, the 27 sets of experimental data were classified according to the grading ranges, as shown in Table 8.

**Table 7.** Classification ranges for surface roughness.

| Roughness Value (μm) | Class Label |
|---|---|
| $Ra \leq 12.5$ | N10 |
| $12.5 < Ra \leq 25$ | N11 |
| $25 < Ra \leq 50$ | N12 |
| $50 < Ra$ | N13 |

### *4.3. Evaluation of Predictive Model Performance*

In this section, the performance of APSO-KNN in predicting surface finish levels of FDM parts is evaluated and compared with existing mainstream classifiers, including Support Vector Machine (SVM), Artificial Neural Network (ANN), K-nearest neighbor (KNN), and Decision Tree. All models were developed using MATLAB R2021a [51].

**Table 8.** Measurement values and category comparison table.

| Test | Ra (μm) | Class Label | Sign |
|---|---|---|---|
| 1 | 15.94 | N11 | 2 |
| 2 | 14.58 | N11 | 2 |
| 3 | 10.78 | N10 | 1 |
| 4 | 44.66 | N12 | 3 |
| 5 | 32.30 | N12 | 3 |
| 6 | 25.17 | N12 | 3 |
| 7 | 51.32 | N13 | 4 |
| 8 | 43.48 | N12 | 3 |
| 9 | 35.83 | N12 | 3 |
| 10 | 8.76 | N10 | 1 |
| 11 | 16.57 | N11 | 2 |
| 12 | 11.99 | N10 | 1 |
| 13 | 23.24 | N11 | 2 |
| 14 | 41.45 | N12 | 3 |
| 15 | 31.69 | N12 | 3 |
| 16 | 30.81 | N12 | 3 |
| 17 | 51.81 | N13 | 4 |
| 18 | 40.02 | N12 | 3 |
| 19 | 17.03 | N11 | 2 |
| 20 | 7.81 | N10 | 1 |
| 21 | 24.21 | N11 | 2 |
| 22 | 29.55 | N12 | 3 |
| 23 | 20.44 | N11 | 2 |
| 24 | 34.05 | N12 | 3 |
| 25 | 41.31 | N12 | 3 |
| 26 | 30.30 | N12 | 3 |
| 27 | 55.68 | N13 | 4 |

Performance evaluation includes two aspects: comparing the accuracy of classification prediction and comparing model running time. First, the dataset was divided into a training set and a test set, with 70% of the dataset used for training the prediction model and 30% used for verifying the accuracy of the trained model. Then, the performance of each prediction model was examined through the minimum classification error chart and confusion matrix. The minimum error chart is a two-dimensional graph of classification error versus iteration number, showing the convergence speed of each algorithm, while the confusion matrix can show the predictive ability of each algorithm for each category. The results were divided into true positive rate (TPR) and false negative rate (FNR), where TPR represents the proportion of correctly predicted categories, while FNR represents the proportion of misclassified categories. Finally, the model running times were compared.

Comparative analysis was conducted using different classification indicators. The minimum prediction error of the APSO-KNN model is shown in Figure 5. Each model was trained using 19 data points (3-N10, 5-N11, 9-N12, and 2-N13) and tested using 8 data points (1-N10, 2-N11, 4-N12, and 1-N13) for surface finish prediction. The results showed that the APSO-KNN model converged after two iterations, with a minimum classification error of 0.01. The Decision Tree and ANN models achieved minimum classification errors of 0.32 and 0.32 after four and eight iterations, respectively. The number of layers of neurons in the ANN model is one of the important factors affecting its prediction accuracy. When the ANN model performs classification prediction, it optimizes the number of layers of neurons to search for the globally optimal number of layers that minimize the classification error. The model automatically adjusts the number of layers of neurons during iterations 19–30, but this is not the optimal number, which is why it jumps.

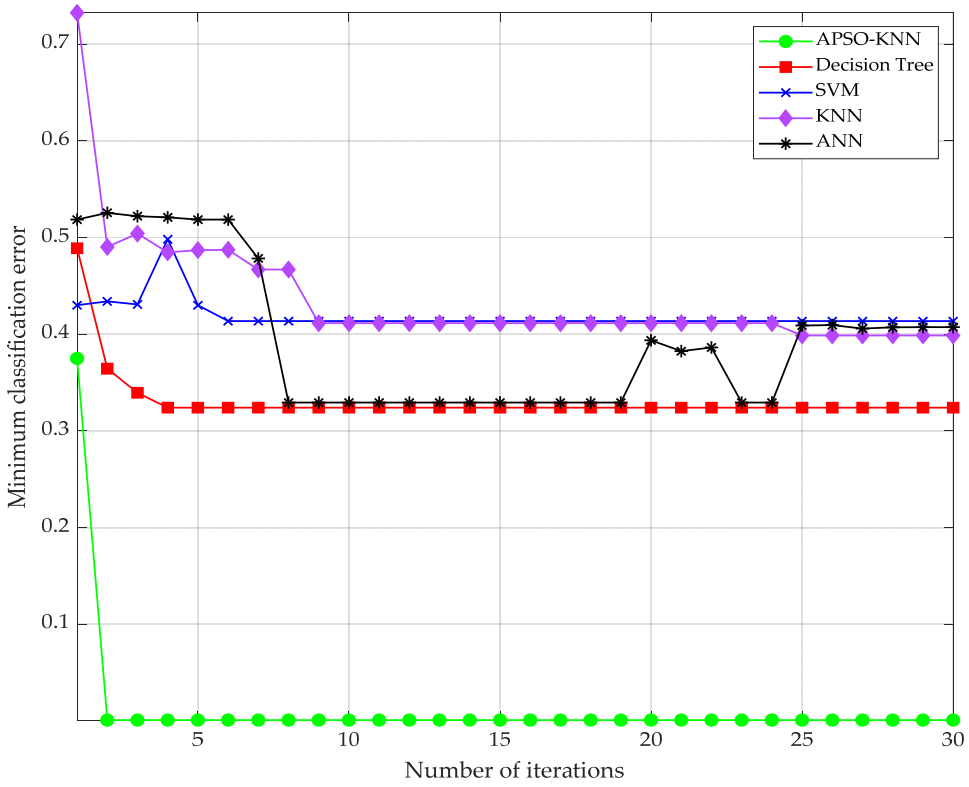



**Figure 5.** Minimum classification error: APSO-KNN, Decision tree, SVM, KNN, and ANN.

The optimal hyperparameter settings required for each model to achieve the minimum classification error are listed in Table 9.

**Table 9.** Summary of the training of each model.

| Model | Hyperparameter | Minimum Error | Training Accuracy | Training Time (s) |
|---|---|---|---|---|
| APSO-KNN | c1:2; c2:2; wmax:0.8; wmin:0.6; M:2; D:7; k:1 | 0.01 | 99.0% | 14.40 |
| Decision Tree | Maximum number of divisions: 1.<br>Split criterion: Gini diversity index.<br>Optimizer: Bayesian optimization | 0.32 | 67.5% | 14.18 |
| SVM | Kernel function: linear.<br>Box constraint level: 1.<br>Multiple methods: one-to-one.<br>Standardized data: true.<br>Optimizer: Bayesian optimization | 0.41 | 59.3% | 36.63 |
| KNN | k:1.<br>Distance Metric: Euclidean.<br>Distance weight: equidistant.<br>Standardized data: true.<br>Optimizer: Bayesian optimization | 0.39 | 60.1% | 24.88 |
| ANN | Number of layers: 3.<br>Activation function: ReLU.<br>Iteration limit: 1000.<br>Standardized data: true.<br>Optimizer: Bayesian optimization | 0.32 | 67.1% | 39.92 |

Figures 6–10 display the confusion matrices of the test data for each model under the hyperparameter settings that resulted in the minimum classification error. The areas where the classifier performs poorly can be found by the percentage and color displayed within the cell, with orange representing predicted classes that do not match true classes. The higher the percentage, the darker the color inside the cell. It can be observed that only the APSO-KNN, SVM, and ANN models are capable of predicting the highest level of surface finish for class 1 (N10) with a small amount of data. However, the Decision Tree model

predicts class 1 (N10) as a similar class 2 (N11), and the KNN model predicts it as class 3 (N12) with a larger amount of data. For class 2 (N11), only the APSO-KNN model can make accurate predictions, while the KNN model deviates completely by predicting the class with the largest amount of data, class 3 (N12). Other models have a TPR of 50%, and some will be predicted incorrectly (FNR is 50%). For class 3 (N12) with the largest amount of data, the SVM and ANN models are still affected by the excessive number of predicted categories, resulting in a large prediction error, with TPRs of 25% and 50%, respectively. The APSO-KNN model can accurately predict class 4 (N13), while other models predict it as class 3 (N12). Figure 11 shows the TPR of each model, with the APSO-KNN model having the best predictive performance. When the input variable weights are not searched using the APSO algorithm, the KNN model's prediction results are biased toward the class with the largest amount of data (Figure 9). This problem is resolved by introducing the APSO algorithm, which accurately predicts and classifies imbalanced data with multiple factors and categories.

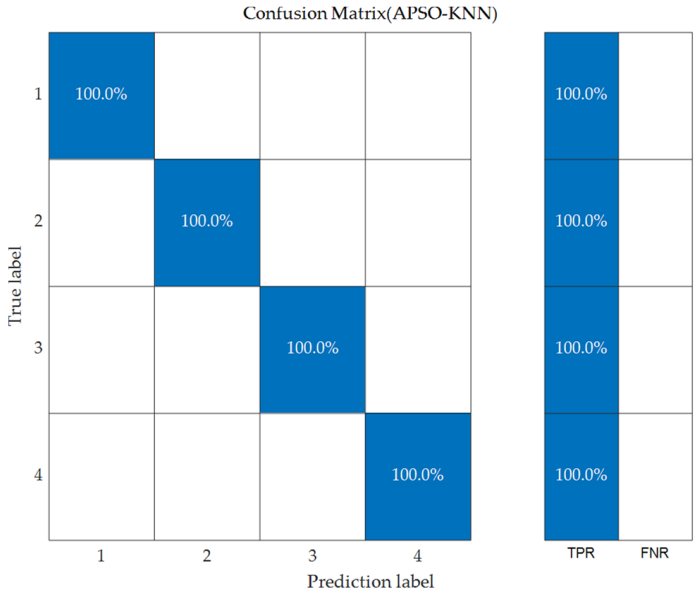



**Figure 6.** APSO–KNN model test of the confusion matrix.

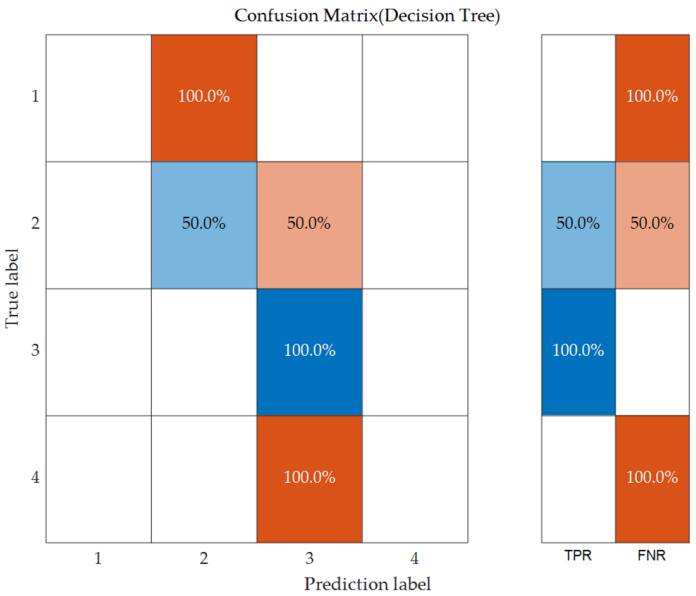



**Figure 7.** Decision Tree model test of the confusion matrix.

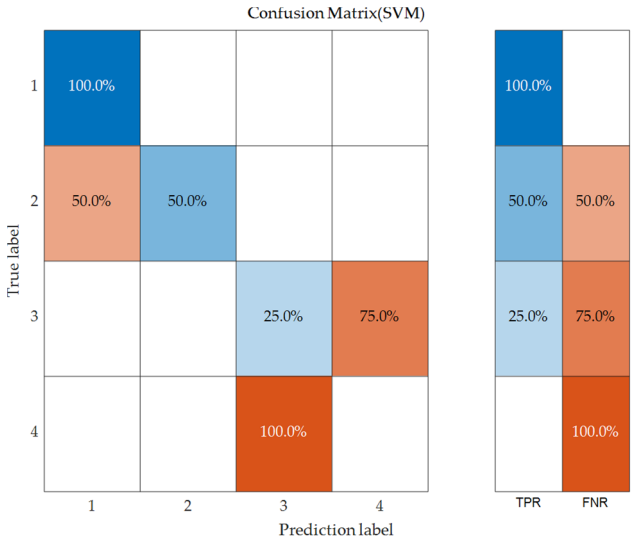



**Figure 8.** SVM model test of the confusion matrix.

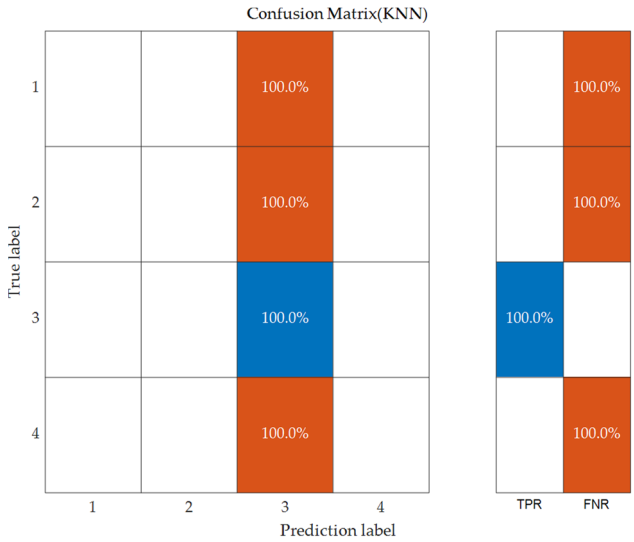



**Figure 9.** KNN model test of the confusion matrix.

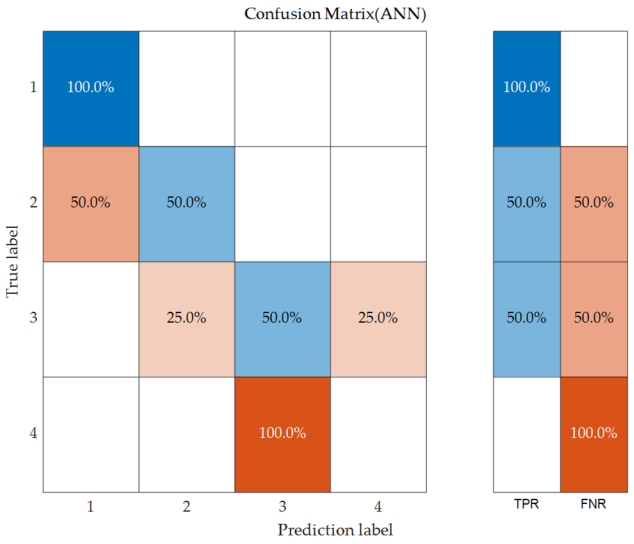



**Figure 10.** ANN model test of the confusion matrix.

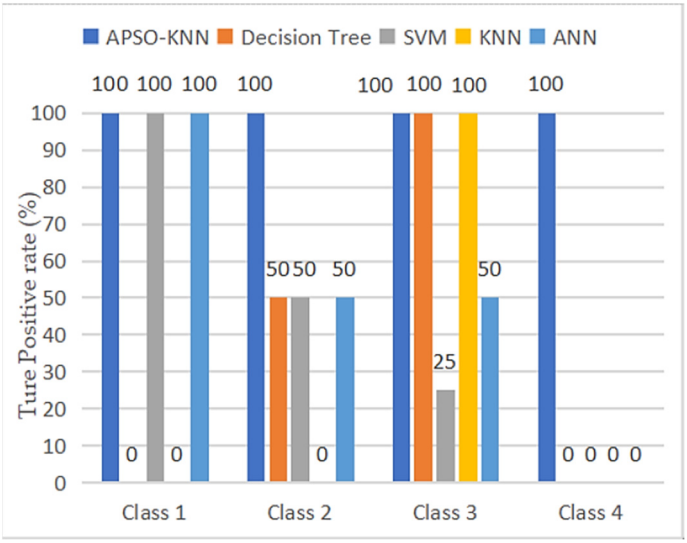



**Figure 11.** Summary graph of TPR for each model.

From Table 9, it can be observed that the training time of the APSO-KNN model is optimized compared to the KNN model, with a relative reduction of 10 s, which enhances the model's predictive efficiency. The experiment results demonstrate that the proposed prediction model has good applicability in multi-factor and multi-class prediction and imbalanced data situations. It can provide instructive predictive data for the surface finish of FDM-printed parts in engineering practice.

## 5. Conclusions

In the current work, a method for predicting the surface finish of FDM PLA parts based on APSO-KNN is developed. The input attributes considered in this method include nozzle diameter, layer thickness, number of perimeters, flow rate, printing speed, nozzle temperature, and build orientation. The output variable is the discrete values of Ra measured on the sample surface: N10 (surface roughness value less than 12.5 μm), N11 (surface roughness value between 12.5 μm and 25 μm), N12 (surface roughness value between 25 μm and 50 μm), and N13 (surface roughness value greater than 50 μm). These values are taken from ISO 1302:2002 [26] and are commonly used in actual industrial production. The printing values of each sample were determined using the L27 Taguchi experimental design, and 27 samples were printed. The experimental data for the 27 samples were used to train and validate the method. The performance was compared with the most commonly used classifiers in existing research, including Decision Tree, SVM, KNN, and ANN. The actual classification accuracy and training time were used as comparison criteria for each model.

The comparative results show that the APSO-KNN prediction model performs the best, completing almost all predictions for the eight test datasets (1-N10, 2-N11, 4-N12, and 1-N13), with a TPR of 100% for N10, N11, N12, and N13. Moreover, the APSO-KNN model completed the prediction after only two iterations, and the training time of the model was only 14 s, indicating a satisfactory running efficiency. This model can solve the problems caused by data imbalance and multiple factors and multiple categories in predicting the surface finish of FDM-printed parts.

A limitation of the proposed method lies in the selection of the $k$ value, which is a controversial issue in academia. Adjustments need to be made based on the actual situation, as too large or too small $k$ values may affect classification accuracy, leading to a lack of uniformity in $k$ value selection. In future work, deep learning methods will be explored to address this issue and increase the uniformity and robustness of the proposed method across different datasets. This would provide strong data support for the development of a

surface finish prediction system for FDM-manufactured parts, improve the efficiency of actual engineering production, and reduce unnecessary material waste.

**Author Contributions:** Conceptualization, M.H., S.J. and Y.Q.; methodology, M.H., S.J. and Y.Q.; software, S.J.; validation, S.J. and Y.C.; formal analysis, S.J.; investigation, S.J.; resources, S.J. and Z.T.; data curation, S.J.; writing—original draft preparation, M.H., S.J. and Y.Q.; writing—review and editing, M.H., S.J. and Y.Q.; visualization, S.J.; supervision, project administration, funding acquisition, Y.Q. and M.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China (No. 52165064 and No. 52105511). The APC was funded by the National Natural Science Foundation of China (No. 52105511).

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

| | |
|---|---|
| AM | Additive Manufacturing |
| FDM | Fused Deposition Modeling |
| ML | Machine Learning |
| DT | Decision Tree |
| LH | Layer Height |
| T | Temperature |
| ND | nozzle diameter |
| PS | Print Speed |
| PA | Print Acceleration |
| F | Flow Rate |
| WA | Wall Angle |
| NP | Number of Perimeters |
| BO | Build Orientation |
| KNN | K-nearest Neighbor |
| APSO | Adaptive Particle Swarm Optimization |
| PSO | Particle Swarm Optimization |
| PLA | Polylactic Acid |
| SVM | Support Vector Machine |
| ANN | Artificial Neural Network |
| TPR | True Positive Rate |
| FNR | False Negative Rate |
| SD | Standard Deviations |
| Ra | Surface Roughness |

## Nomenclature

| | |
|---|---|
| $x_{i,j}$ | The numerical values of particle displacement (no unit) |
| $v_{i,j}$ | The numerical values of particle velocity (no unit) |
| $k$ | The number of neighbors (no unit) |
| $\omega$ | The inertia weight (no unit) |
| $c_1$ | The learning factors (no unit) |
| $c_2$ | The learning factors (no unit) |
| $r_1$ | Uniformly distributed random numbers within [0, 1] (no unit) |
| $r_2$ | Uniformly distributed random numbers within [0, 1] (no unit) |
| $p_{i,j}$ | The individual best value of the particle (no unit) |
| $p_{g,j}$ | The global best value of the particle (no unit) |
| $f$ | The real-time objective function value of the particle (no unit) |
| $f_{avg}$ | The average value of all particles at present (no unit) |
| $f_{min}$ | The minimum target value of all particles at present (no unit) |
| $\omega_{max}$ | The maximum values of the inertia weight (no unit) |

| | |
|---|---|
| $\omega_{min}$ | The minimum values of the inertia weight (no unit) |
| *acc* | The value of accuracy (no unit) |
| *NC* | The number of correct classifications (no unit) |
| *NT* | The total number of classification labels (no unit) |

## References


1. Blakey-Milner, B.; Gradl, P.; Snedden, G.; Brooks, M.; Pitot, J.; Lopez, E.; Leary, M.; Berto, F.; du Plessis, A. Metal additive manufacturing in aerospace: A review. *Mater. Des.* **2021**, *209*, 110008. [CrossRef]
2. Sun, B.; Ma, Q.; Wang, X.; Liu, J.; Rejab, M.R.M. Additive manufacturing in medical applications: A brief review. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1078*, 12007. [CrossRef]
3. Khorasani, M.; Loy, J.; Ghasemi, A.H.; Sharabian, E.; Leary, M.; Mirafzal, H.; Cochrane, P.; Rolfe, B.; Gibson, I. A review of Industry 4.0 and additive manufacturing synergy. *Rapid Prototyp. J.* **2022**, *28*, 1462–1475. [CrossRef]
4. Nair, A.; Aditya, S.D.; Adarsh, R.N.; Nandan, M.; Dharek, M.S.; Sreedhara, B.M.; Prashant, S.C.; Sreekeshava, K.S. Additive Manufacturing of Concrete: Challenges and opportunities. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *814*, 12022. [CrossRef]
5. Gao, W.; Zhang, Y.; Ramanujan, D.; Ramani, K.; Chen, Y.; Williams, C.B.; Wang, C.C.L.; Shin, Y.C.; Zhang, S.; Zavattieri, P.D. The status, challenges, and future of additive manufacturing in engineering. *Comput. Aided Des.* **2015**, *69*, 65–89. [CrossRef]
6. Kumar, L.J.; Pandey, P.M.; Wimpenny, D.I. *3D Printing and Additive Manufacturing Technologies*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 311.
7. Sathies, T.; Senthil, P.; Anoop, M.S. A review on advancements in applications of fused deposition modelling process. *Rapid Prototyp. J.* **2020**, *26*, 669–687. [CrossRef]
8. Parada, L.R.; Mayuet, P.F.; Gámez, A.J. Industrial product design: Study of FDM technology for the manufacture of thermoformed prototypes. *Procedia Manuf.* **2019**, *41*, 587–593. [CrossRef]
9. Kechagias, J.D.; Zaoutsos, S.P. Optimising fused filament fabrication surface roughness for a dental implant. *Mater. Manuf. Process.* **2023**, *38*, 954–959. [CrossRef]
10. Di Angelo, L.; Di Stefano, P.; Marzola, A. Surface quality prediction in FDM additive manufacturing. *Int. J. Adv. Manuf. Technol.* **2017**, *93*, 3655–3662. [CrossRef]
11. Ferretti, P.; Leon-Cardenas, C.; Santi, G.M.; Sali, M.; Ciotti, E.; Frizziero, L.; Donnici, G.; Liverani, A. Relationship between FDM 3D Printing Parameters Study: Parameter Optimization for Lower Defects. *Polymers* **2021**, *13*, 2190. [CrossRef]
12. Buj-Corral, I.; Sánchez-Casas, X.; Luis-Pérez, C.J. Analysis of AM Parameters on Surface Roughness Obtained in PLA Parts Printed with FFF Technology. *Polymers* **2021**, *13*, 2384. [CrossRef] [PubMed]
13. Pérez, M.; Medina-Sánchez, G.; García-Collado, A.; Gupta, M.; Carou, D. Surface Quality Enhancement of Fused Deposition Modeling (FDM) Printed Samples Based on the Selection of Critical Printing Parameters. *Materials* **2018**, *11*, 1382. [CrossRef] [PubMed]
14. Huang, M.; Zheng, N.; Qin, Y.; Tang, Z.; Zhang, H.; Fan, B.; Qin, L. Description Logic Ontology-Supported Part Orientation for Fused Deposition Modelling. *Processes* **2022**, *10*, 1290. [CrossRef]
15. Qin, Y.; Qi, Q.; Shi, P.; Scott, P.J.; Jiang, X. Status, issues, and future of computer-aided part orientation for additive manufacturing. *Int. J. Adv. Manuf. Technol.* **2021**, *115*, 1295–1328. [CrossRef]
16. Taufik, M.; Jain, P.K. A Study of Build Edge Profile for Prediction of Surface Roughness in Fused Deposition Modeling. *J. Manuf. Sci. Eng.* **2016**, *138*, 4032193. [CrossRef]
17. Buj-Corral, I.; Domínguez-Fernández, A.; Durán-Llucià, R. Influence of Print Orientation on Surface Roughness in Fused Deposition Modeling (FDM) Processes. *Materials* **2019**, *12*, 3834. [CrossRef]
18. Xu, F.; Wong, Y.S.; Loh, H.T. Toward generic models for comparative evaluation and process selection in rapid prototyping and manufacturing. *J. Manuf. Syst.* **2001**, *19*, 283–296. [CrossRef]
19. Luis Perez, C.J.; Vivancos, J.; Sebastián, M.A. Surface roughness analysis in layered forming processes. *Precis. Eng.* **2001**, *25*, 1–12. [CrossRef]
20. Pandey, P.M.; Venkata Reddy, N.; Dhande, S.G. Improvement of surface finish by staircase machining in fused deposition modeling. *J. Mater. Process. Technol.* **2003**, *132*, 323–331. [CrossRef]
21. Vahabli, E.; Rahmati, S. Application of an RBF neural network for FDM parts' surface roughness prediction for enhancing surface quality. *Int. J. Precis. Eng. Manuf.* **2016**, *17*, 1589–1603. [CrossRef]
22. Wu, D.; Wei, Y.; Terpenny, J. Predictive modelling of surface roughness in fused deposition modelling using data fusion. *Int. J. Prod. Res.* **2019**, *57*, 3992–4006. [CrossRef]
23. Boschetto, A.; Giordano, V.; Veniali, F. Surface roughness prediction in fused deposition modelling by neural networks. *Int. J. Adv. Manuf. Technol.* **2013**, *67*, 2727–2742. [CrossRef]
24. Reddy, V.; Flys, O.; Chaparala, A.; Berrimi, C.E.; Amogh, V.; Rosen, B.G. Study on surface texture of Fused Deposition Modeling. *Procedia Manuf.* **2018**, *25*, 389–396. [CrossRef]
25. Meng, L.; McWilliams, B.; Jarosinski, W.; Park, H.; Jung, Y.; Lee, J.; Zhang, J. Machine Learning in Additive Manufacturing: A Review. *JOM* **2020**, *72*, 2363–2377. [CrossRef]
26. *UNE-EN ISO 1302-2002*; Geometrical Product Specifications (GPS)—Indication of Surface Texture in Technical Product Documentation. International Organization for Standardization: Geneva, Switzerland, 2002.

27. Aoyagi, K.; Wang, H.; Sudo, H.; Chiba, A. Simple method to construct process maps for additive manufacturing using a support vector machine. *Addit. Manuf.* **2019**, *27*, 353–362. [CrossRef]
28. Molero, E.; Fernández, J.J.; Rodríguez-Alabanda, O.; Guerrero-Vaca, G.; Romero, P.E. Use of Data Mining Techniques for the Prediction of Surface Roughness of Printed Parts in Polylactic Acid (PLA) by Fused Deposition Modeling (FDM): A Practical Application in Frame Glasses Manufacturing. *Polymers* **2020**, *12*, 840. [CrossRef] [PubMed]
29. Barrios, J.M.; Romero, P.E. Decision Tree Methods for Predicting Surface Roughness in Fused Deposition Modeling Parts. *Materials* **2019**, *12*, 2574. [CrossRef] [PubMed]
30. Cerro, A.; Romero, P.E.; Yiğit, O.; Bustillo, A. Use of machine learning algorithms for surface roughness prediction of printed parts in polyvinyl butyral via fused deposition modeling. *Int. J. Adv. Manuf. Technol.* **2021**, *115*, 2465–2475. [CrossRef]
31. García Plaza, E.; Núñez López, P.J.; Caminero Torija, M.Á.; Chacón Muñoz, J.M. Analysis of PLA Geometric Properties Processed by FFF Additive Manufacturing: Effects of Process Parameters and Plate-Extruder Precision Motion. *Polymers* **2019**, *11*, 1581. [CrossRef] [PubMed]
32. Taşcıoğlu, E.; Kıtay, Ö.; Keskin, A.Ö.; Kaynak, Y. Effect of printing parameters and post-process on surface roughness and dimensional deviation of PLA parts fabricated by extrusion-based 3D printing. *J. Braz. Soc. Mech. Sci.* **2022**, *44*, 139. [CrossRef]
33. Ramli, F.R.; Faudzie, M.; Nazan, M.A.; Alkahari, M.R.; Sudin, M.N.; Mat, S.; Khalil, S.N. Dimensional accuracy and surface roughness of part features manufactured by open source 3D printer. *ARPN J. Eng. Appl. Sci.* **2018**, *13*, 1139–1144.
34. Alsoufi, M.S.; Elsayed, A.E. How surface roughness performance of printed parts manufactured by desktop FDM 3D printer with PLA+ is influenced by measuring direction. *Am. J. Mech. Eng.* **2017**, *5*, 211–222.
35. Wu, Q.; Liu, H.; Yan, X. Multi-label classification algorithm research based on swarm intelligence. *Clust. Comput.* **2016**, *19*, 2075–2085. [CrossRef]
36. Wu, S. Simulation of classroom student behavior recognition based on PSO-kNN algorithm and emotional image processing. *J. Intell. Fuzzy Syst.* **2021**, *40*, 7273–7283. [CrossRef]
37. Han, J.; Abdelaziz, M. The Method for Identifying Employees' Emotions in Adverse States Incorporating PSO-kNN Algorithm and Multiple Physiological Parameters. *Comput. Intel. Neurosci.* **2022**, *2022*, 4371162. [CrossRef]
38. Sasirekha, K.; Thangavel, K. Optimization of K-nearest neighbor using particle swarm optimization for face recognition. *Neural Comput. Appl.* **2019**, *31*, 7935–7944. [CrossRef]
39. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In Proceedings of the MHS'95 Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995; IEEE: Piscataway, NJ, USA, 1995; pp. 39–43.
40. Song, X.; Zhang, Y.; Gong, D.; Gao, X. A fast hybrid feature selection based on correlation-guided clustering and particle swarm optimization for high-dimensional data. *IEEE Trans. Cybern.* **2021**, *52*, 9573–9586. [CrossRef] [PubMed]
41. Ji, X.; Zhang, Y.; Gong, D.; Sun, X. Dual-surrogate-assisted cooperative particle swarm optimization for expensive multimodal problems. *IEEE Trans. Evol. Comput.* **2021**, *25*, 794–808. [CrossRef]
42. Zhang, Y.; Gong, D.; Ding, Z. A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch. *Inf. Sci.* **2012**, *192*, 213–227. [CrossRef]
43. Deng, H.; Peng, L.; Zhang, H.; Yang, B.; Chen, Z. Ranking-based biased learning swarm optimizer for large-scale optimization. *Inf. Sci.* **2019**, *493*, 120–137. [CrossRef]
44. Kramer, O.; Kramer, O. K-nearest Neighbors. In *Dimensionality Reduction with Unsupervised Nearest Neighbors*; 2013; Volume 51, pp. 13–23. Available online: https://link.springer.com/chapter/10.1007/978-3-642-38652-7_2 (accessed on 5 June 2023).
45. Flach, P. *Machine Learning: The Art and Science of Algorithms That Make Sense of Data*; Cambridge University Press: Cambridge, UK, 2012.
46. Clerc, M. *Particle Swarm Optimization*; John Wiley & Sons: Hoboken, NJ, USA, 2010; Volume 93.
47. Ravikumar, G.; Gomathi, D.; Kalaiselvi, M.; Uma, C. Adaptive particle swarm optimization. *Electr. Eng. Jpn.* **2010**, *151*, 41–49.
48. Gao, G.; Xu, F.; Xu, J.; Liu, Z. Study of Material Color Influences on Mechanical Characteristics of Fused Deposition Modeling Parts. *Materials* **2022**, *15*, 7039. [CrossRef] [PubMed]
49. *ISO 21920-2:2021*; Geometrical Product Specifications (GPS)—Surface Texture: Profile—Part 2: Terms, Definitions and Surface Texture Parameters. International Organization for Standardization: Geneva, Switzerland, 2021.
50. Roy, R.K. *A Primer on the Taguchi Method*; Society of Manufacturing Engineers: Southfield, MI, USA, 2010.
51. Matlab, S. *Matlab*; The MathWorks: Natick, MA, USA, 2021.