The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM)
ISSN: 2602-3199

The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM), 2024

Volume 28, Pages 246-251

ICBASET 2024: International Conference on Basic Sciences, Engineering and Technology

Machine Learning Approach for Predicting Bead Geometry of Stainless Steel in Wire arc Additive Manufacturing

Harsh Shah
Pandit Deendayal Energy University

Kishan Fuse
Pandit Deendayal Energy University

Abstract: Wire arc additive manufacturing (WAAM) employs an electric arc to melt wire feedstock, making it a method within additive manufacturing (AM). It deposits material layer by layer to build up a part. The present study investigated the application of machine learning classification-based models for estimating bead width and bead height of stainless-steel parts fabricated using WAAM. The input parameters (voltage, current, wire feed rate, and travel speed) were considered as input to algorithms. Training and testing were performed for 98 experimental data sets from peer-reviewed literature. The machine learning classification models, K-nearest neighbors, decision tree with gini index as criteria, and random forest were evaluated. The ML model performance was evaluated utilizing statistical metrics, including accuracy, F1 score, precision, and recall. The decision tree classifier exhibited the highest accuracy of 87.8% for bead width and 84.7% for bead height. The findings offer valuable insights into leveraging ML techniques to enhance the performance and accuracy of predictive models within WAAM-based AM.

Keywords: Machine learning, Wire arc, Bead geometry of stainless steel,

Introduction

Traditional manufacturing techniques are gradually being replaced by additive manufacturing (AM). The AM can be divided into several kinds according to the production technique, such as binder jetting, sheet lamination, powder bed fusion, directed energy deposition (DED), etc. (Srivastava et al.,2022). In contrast to polymeric material additive manufacturing, metal AM is very new and requires careful consideration. The two primary categories of AM procedures for metal are wire-based and powder-based methods (Li, 2021). Wire arc additive manufacturing (WAAM), one of the various DED technologies available today, is the most significant DED technique. Large-scale fabrication of metallic components is made possible by WAAM, which includes GMAW, GTAW, and PAW. Aerospace, automotive, oil and gas, nuclear, shipbuilding, heavy fabrication, and other sectors stand to benefit greatly from WAAM's potential to revolutionize production processes since it uses robots with gantry systems and floor-mounted tracks to make components of any size (Yaseer, 2021). Additionally, there has been a rise in studies in the field of AM utilizing machine learning (ML). This is because ML can effectively save costs, optimize the production process, and increase quality. Numerous researchers seemed to like ML techniques such as decision tree (DT), random forest (RF), and k-nearest neighbour (KNN) for defect detection and bead geometry prediction. The prediction of bead width (BW) and bead height (BH) using the RF ML technique was done and showed higher accuracy (Sharma et al.,2023). RF and neural networks (NN) were used to predict YS and UTS for WAAM 316L steel parts. They used performance evaluators such as R2, Adj R2, and RMSE for RF (Mamedipaka & Thapliyal, 2024). For NN, performance was evaluated by loss and mean absolute error. ML models such as DT, RF, XGBoost algorithm, ANN, and linear regression were employed on the WAAM part fabricated using Al 5356 alloy. They reported better performance for ANN than XGBoost in predicting BH. For BW prediction, they found linear regression better performing ML model than

ANN (Chandra et al., 2023). The developed predictive models effectively crafted new prediction algorithms for identifying defects within WAAM deposited layer (Cheepu, 2022). Gaussian process regressor was used to predict melting efficiency in WAAM. They used input variables such as diameter of feeding wire, feeding speed of the wire, nozzle speed, and net power to calculate melting efficiency (Barrionuevo et al., 2022). Out of all the ML models, this study used the DT, RF, and KNN on 98 experimental data sets. The programming language employed was Python. The results of ML computations were visualized using packages like scikit-learn, NumPy, pandas, and Matplotlib (Oh et al., 2022).

Machine Learning Classification Models

The present study investigated three ML classification models as DT, RF, and KNN. This section highlights detail about all three ML techniques. The Decision Tree Classification Algorithm works by asking some binary questions to predict the output variable through the input variables. These questions create a hierarchical image resembling a tree, starting with the first classification at the top i.e. the root node and further into others. With each classifier, the data is divided into two subsets based on the response, which will continue until the tree's leaf node provides a definitive prediction. The main objective of the DT classifier is to formulate questions that yield the most relevant data regarding the output variable while asking very few possible questions. This is achieved by taking the optimal question to ask at each branching point in the tree, based on a predetermined criterion (e.g., Gini index). After the decision tree is generated, new data can be predicted by tracing the path from the root to the relevant leaf node. Each leaf node reflects to a specific value of the response, and the value associated with that leaf node gives the predicted output.

Random Forest is a widely utilized supervised machine learning framework constructed upon an ensemble of multiple trees. Renowned for its rapid processing speed and robustness against outliers, RF excels in analyzing intricate nonlinear relationships. Throughout the training phase, RF discerns patterns to establish connections between the input (X) and output (Y) variables, overseeing the learning process (Mamedipaka & Thapliyal, 2024).

The KNN algorithm runs by computing the distance between n training samples and the new specimen, subsequently considering the K samples with the smaller distances to the new specimen. The parameter K (k=10), defined by the user, determines the number of training samples utilized for prediction. Euclidean distance can be gained by computing the distance between the new specimen and a training sample. Subsequently, upon identifying the nearest K training samples, the predominant label among these K samples is used to establish the predicted labels. For example, the new specimen will be labelled as 0 if most of the nearest K samples have a label of 0; similarly, the new specimen will be labelled as 1, if majority have a label of 1 (Sharma et al., 2023).

Methodology

In the present study, 98 datasets from peer-reviewed literatures (Chaudhari et al., 2022; Le et al.,2021; Kumar & Maji, 2020; Sharma et al., 2023b; Vora et al., 2022; Yadav et al., 2024) was analysed using 10-fold cross-validation. Further, the dataset was subdivided into 10 subsets. This made it possible to train and test the created model ten times, striking a compromise between computing efficiency and variance reduction. Bead width and bead height were considered to be the response parameters in the experimental dataset. The schematic of the BW and BH is as shown in Figure 1. The input parameters in the dataset were wire feed rate (m/min), travel speed (mm/min), current (A), and voltage (V). The output parameters were bead width (mm) and bead height (mm). Before evaluation, the datasets underwent pre-processing, wherein missing values for input variables were substituted with their respective mean values.

The dataset was subjected to machine learning classification models i.e. DT, RF and KNN. The output was classified as 0 or 1 based upon corresponding average values. The average of bead width was 6.78mm. The BW was classified as 1 when the it was less than 6.78 mm and 0 when greater than 6.78 mm. The average of bead height was 3.8mm. The BH was classified as 1, when it was less than 3.8 mm and 0 when it was greater than 3.8 mm. To assess and collate the effectiveness of these classification models, four primary metrics were considered: accuracy, F1 score, precision, and recall. The accuracy evaluates algorithm's performance by measuring the proportion of correctly classified instances among all evaluated instances. The ratio of true positives to all projected positives is called precision, while the ratio of genuine positives to all actual positives is called recall. An overall evaluation of the performance of the classification algorithm was provided by the F1

score, which is a harmonic mean of precision and recall. Additionally, a confusion matrix, which represents a matrix that contrasts the genuine labels of a particular test dataset with the labels predicted by the algorithm. It is a necessary evaluation tool for classification algorithms. A confusion matrix as shown in Figure 3 was generated for all the instances to get the insights of the DT, RF, and KNN algorithm (Mishra et al., 2024).

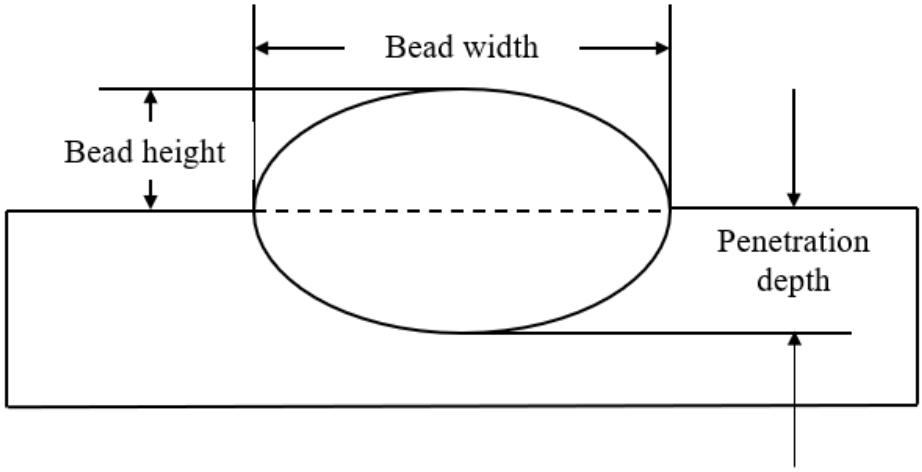

Figure 1. Schematic of bead width and bead height

Hyper-parameters Tuning

The scikit-learn library comes with pre-set hyperparameters for all machine learning algorithm, yet it doesn't ensure the optimal parameter values for a specific problem. Consequently, tuning hyperparameters becomes crucial in enhancing the predictive accuracy of ML models. A variety of hyperparameter optimisation algorithms, such as grid search, randomised search, and bayesian search are available with the scikit-learn package. To obtain the best hyperparameters for the ML algorithms, the present study utilized both manual techniques and scikit-learn packages. Table 1 lists the optimised values of critical hyper parameters for each ML model. Furthermore, to enhance the ML algorithms adaptability, K-fold cross-validation was utilised during implementation. The training dataset was subjected to a tenfold cross-validation for both training and validation purposes for all three studied ML models.

Table 1. Hyper-parameter optimized for ML algorithms

SN	ML Model	Hyper Parameter
1	Decision Tree	criterion: gini, splitter: best, max_feature: auto
2	Random Forest	n_estimators:100, criterion: gini, max_depth: 2
3	K-Nearest Neighbour	n_neighbour:10, n_jobs: -1

Results and Discussion

Prediction of Bead Width

To forecast bead width, the statistical metrics (Precision, Recall, and F1-score) evaluating the performance of three ML models on both training and testing data are consolidated in Table 2. KNN obtained the lowest value of the key metrics which indicated the poor labelling of algorithm and poor overall performance. While DT obtained the highest value of key metrics which can be validated using confusion matrix shown in Figure 3. The highest accuracy in predicting the bead width was resulted by DT is 87.8%, followed by RF 77.4% and KNN being the lowest at 70.2%. A comparison of accuracy for all the three algorithms is shown in Figure 2.

Table 2. Precision, recall and F1-score of different classifier model for bead width

Classifier	Precision	Recall	F1-Score
Decision Tree	0.907	0.875	0.876
Random Forest	0.831	0.774	0.762
K-Nearest Neighbour	0.770	0.702	0.702

Figure 2. Bar chart comparison of the accuracy of different ML models for (a) bead width (b) bead height

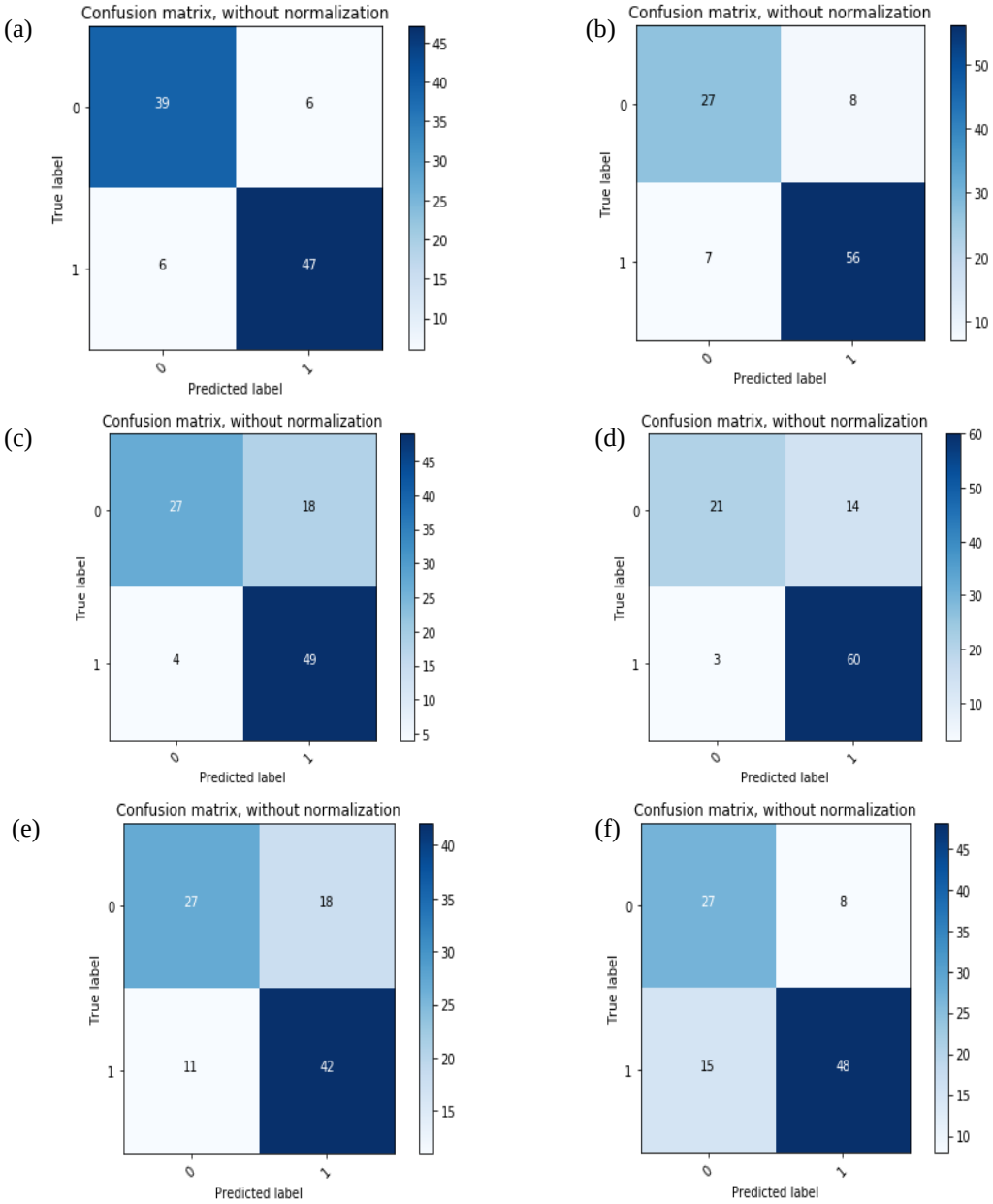

Figure 3. Confusion matrix comparing the actual and predicted values from the A) Decision Tree for bead width B) Decision Tree for bead height C) Random Forest for bead width D) Random Forest for bead height E) K-Nearest Neighbour for bead width F) k-Nearest Neighbour for bead height

Prediction of Bead Height

For predicting the bead height, the value of statistical metrics (Precision, Recall, and F1-score) for measuring the performance of three ML models for both training and testing are outlined in Table 3. The lowest value of key metrics is obtained for KNN which can be due to imbalanced data, whereas DT showed great results in labelling the algorithm. DT achieved the highest accuracy of 84.7% for prediction of bead height, followed by RF with 82.6% accuracy and KNN having the lowest accuracy of 76.4%. Figure 2 shows the accuracy comparison of all the three algorithms.

Table 3. Precision, recall and F1-score of different classifier model for bead height

Classifier	Precision	Recall	F1-Score
Decision Tree	0.856	0.847	0.843
Random Forest	0.841	0.826	0.807
K-Nearest Neighbour	0.822	0.764	0.770

Conclusion

In summary, this research has effectively explored the utilization of supervised ML algorithms for estimating the bead width and bead height of WAAM parts fabricated using stainless steel. By investigating 98 dataset and utilizing input variable such as voltage, current, wire feed rate and travel speed, we have evaluated the accuracy and effectiveness of three distinct supervised classification algorithms: Decision Tree, Random Forest and KNN. The conclusion from the present study are:

- The result illustrates that decision tree outperforms the other algorithm with the accuracy of 87.8% for bead width and 84.7% for bead height. This underscores the decision tree algorithm's exceptional capacity to distinguish between the two categories of bead geometry in the dataset, rendering it the preferred option for classification within this research framework.
- This study sets the foundation for forthcoming research aimed at enhancing these algorithms, fine-tuning variable, and broadening the utilization of ML in additive manufacturing. The ultimate goal is to enhance the standard and reliability of 3D-printed components.

Scientific Ethics Declaration

The authors declare that the scientific ethical and legal responsibility of this article published in EPSTEM journal belongs to the authors.

Acknowledgements or Notes

* This article was presented as an oral presentation at the International Conference on Basic Sciences, Engineering and Technology (www.icbaset.net) held in Alanya/Turkey on May 02-05, 2024.

References

Barrionuevo, G. O., Sequeira-Almeida, P. M., Ríos, S., Ramos-Grez, J. A., & Williams, S. W. (2022). A machine learning approach for the prediction of melting efficiency in wire arc additive manufacturing. *The International Journal of Advanced Manufacturing Technology, 120*(5), 3123–3133.

Chandra, M., Vimal, K. E. K., & Rajak, S. (2023). A comparative study of machine learning algorithms in the prediction of bead geometry in wire-arc additive manufacturing. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 1-14.

Chaudhari, R., Parmar, H., Vora, J., & Patel, V. K. (2022). Parametric study and investigations of bead geometries of GMAW-based wire–arc additive manufacturing of 316L stainless steels. *Metals, 12*(7), 7.

Cheepu, M. (2023). Machine learning approach for the prediction of defect characteristics in wire arc additive manufacturing. *Transactions of the Indian Institute of Metals*, *76*(2), 447–455.

Kumar, A., & Maji, K. (2020). Selection of process parameters for near-net shape deposition in wire arc additive manufacturing by genetic algorithm. *Journal of Materials Engineering and Performance, 29,* 3334–3352.

Le, V. T., Mai, D. S., Doan, T. K., & Paris, H. (2021). Wire and arc additive manufacturing of 308L stainless steel components: Optimization of processing parameters and material properties. *Engineering Science and Technology:An International Journal, 24*(4), 1024-1025.

Li, Y. (2021). *Machine learning based defect detection in robotic wire arc additive manufacturing.* (Doctoral dissertation). University of Wollongong Thesis Collection 2017.

Mamedipaka, R., & Thapliyal, S. (2024). Data-driven model for predicting tensile properties of wire arc additive manufactured 316l steels and its validation. *Journal of Materials Engineering and Performance, 33*(3), 1083–1091.

Mishra, A., Jatti, V. S., Messele -Sefene, E., Jatti, A. V., Sisay, A. D., Khedkar, N. K., Salunkhe, S., Pagáč, M., & Abouel-Nasr, E. S. (2024). Machine learning-assisted pattern recognition algorithms for estimating ultimate tensile strength in fused deposition modelled polylactic acid specimens. *Materials Technology, 39*(1), 2295089.

Oh, W. J., Lee, C. M., & Kim, D. H. (2022). Prediction of deposition bead geometry in wire arc additive manufacturing using machine learning. *Journal of Materials Research and Technology, 20,* 4283–4296.

Sharma, R., Raj Paul, A. R., Mukherjee, M., Vadali, S., Kumar Singh, R., & Sharma, A. K. (2023a). Forecasting of process parameters using machine learning techniques for wire arc additive manufacturing process. *Materials Today: Proceedings, 80,* 248–253.

Srivastava, M., Rathee, S., Patel, V., Kumar, A., & Koppad, P. G. (2022). A review of various materials for additive manufacturing: Recent trends and processing issues, *Journal of Materials Research and Technology, 21,* 2612-2641.

Vora, J., Parmar, H., Chaudhari, R., Khanna, S., Doshi, M., & Patel, V. (2022). Experimental investigations on mechanical properties of multi-layered structure fabricated by GMAW-based WAAM of SS316L. *Journal of Materials Research and Technology, 20,* 2748–2757.

Yadav, A., Srivastava, M., Jain, P. K., & Rathee, S. (2024). Investigation of bead morphology and mechanical behaviour for metal inert gas welding-based WAAM in pulsed mode metal transfer on 316LSi stainless steel. *Journal of Adhesion Science and Technology, 38*(5),738-769.

Yaseer, A. (2021). *Process planning of robotic wire arc additive manufacturing (WAAM) using machine learning*. (Unpublished Master's thesis). Texas State University, Texas.

Author Information

Harsh Shah
Pandit Deendayal Energy University, Gandhinagar, 382007, India
Contact e-mail: *Harshkshah02@gmail.com*

Kishan Fuse
Pandit Deendayal Energy University, Gandhinagar, 382007, India

To cite this article:

Shah, H., & Fuse, K. (2024). Machine learning approach for predicting bead geometry of stainless steel in wire arc additive manufacturing. *The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM), 28,* 246-251.