ELKOM (Jurnal Elektronika dan Komputer)
p-ISSN : 1907-0012 (print) e-ISSN : 2714-5417 (online)
Vol. 18, No. 1, Juli 2025
https://journal.stekom.ac.id/index.php/elkom

Predictive Modeling of Microstrip Antenna Slot Dimensions Using Random Forest Regression

Aisya Nur Aulia Yusuf[1], Elsa Sari Hayunah Nurdiniyah[2], Norma Amalia[3]
[1,2,3]Department of Electrical Engineering, Universitas Jenderal Soedirman Purbalingga, Indonesia
Email author: aisya.yusuf@unsoed.ac.id, elsa.nurdiniyah@unsoed.ac.id, norma.amalia@unsoed.ac.id

Article Info

Article history:

Received Januari 3, 2025
Revised Februari 17, 2025
Accepted June 28, 2025

Keywords:

Machine learning; Microstrip antenna; Prediction; Random forest; Regression

ABSTRACT

This study presents a machine learning approach for predicting the dimensions of microstrip antenna slots based on antenna performance parameters such as frequency, gain, directivity, return loss (S_{11}), radiation efficiency, and VSWR. A two-phase methodology was employed. In the first phase, ten regression algorithms were evaluated, and Random Forest was identified as the most effective model based on Mean Absolute Error (MAE) and R-squared (R^2) scores. In the second phase, hyperparameter tuning was conducted using Grid Search to further improve the model's performance. The optimized Random Forest model demonstrated consistent improvements in predictive accuracy, with R^2 values increasing across all output variables. These results indicate that the combination of regression-based modeling and systematic hyperparameter tuning is effective for capturing complex relationships in antenna design tasks. The proposed approach offers a promising data-driven alternative for geometric prediction in microstrip antenna development, particularly when analytical models are insufficient.

Corresponding Author:

Aisya Nur Aulia Yusuf,
Universitas Jenderal Soedirman
Jl. Raya Mayjen Sungkono No.KM 5 Purbalingga
Email: aisya.nur@unsoed.ac.id

1. INTRODUCTION

Microstrip antennas are a cornerstone in modern communication systems due to their numerous advantages, which make them highly suitable for a wide range of applications. Microstrip antennas are inherently small and lightweight, making them ideal for portable devices such as mobile phones and laptops [1], [2]. Their low profile and volume allow for easy integration into modern electronic circuits and systems, including satellites where space and weight are at a premium [3]. These antennas are cost-effective due to their simple construction, which involves a metal patch on a dielectric substrate with a ground plane [2], [4]. The ease of fabrication and compatibility with integrated circuit technology make them a preferred choice for mass production [3]. Microstrip antennas support dual and circular polarizations, dual-frequency operation, and frequency agility, which are crucial for modern communication systems [5]. They can be designed to operate in multiple bands, such as GSM, Wi-Fi, and

WiMAX, providing flexibility in application [1], [2]. Recent advancements, including metamaterial-enhanced designs, have improved their performance characteristics, such as gain and bandwidth [4]. Microstrip antennas are particularly valued for their compact size, lightweight construction, and ease of integration with electronic circuits, which are essential for portable and satellite communication devices. Their ability to operate across multiple frequency bands further enhances their utility in diverse communication protocols.

While microstrip antennas offer significant advantages, they also present several significant disadvantages that can limit their effectiveness in various applications. MSAs typically exhibit a narrow impedance bandwidth, often ranging from 1% to 5% [6]. This limitation restricts their operational frequency range, making them unsuitable for applications requiring broader bandwidths, such as satellite communications and Wi-Fi devices [6]. Microstrip antennas generally have low gain compared to traditional antennas, which can hinder their performance in long-range communication scenarios [7], [8]. Additionally, they often suffer from low radiation efficiency, which can further diminish their effectiveness in practical applications [8].

However, ongoing research and technological advancements continue to address these limitations, ensuring that microstrip antennas remain a vital component in the evolution of communication systems. One of them is the implementation of machine learning. The integration of machine learning (ML) into microstrip antenna design has seen significant advancements, particularly in enhancing design efficiency and performance optimization. Recent studies highlight various ML techniques that streamline the design process, reduce computational costs, and improve accuracy in predicting antenna characteristics. Techniques such as K-nearest neighbor (KNN), support vector machines, and random forests have been employed to predict antenna performance metrics like gain, directivity, and radiation efficiency [9], [10]. KNN achieved a gain prediction accuracy of 94.23% in sub-6 GHz ranges, while random forests demonstrated a high R-squared value of 0.94 in predicting antenna dimensions [9], [11].

A novel hybrid model combining Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks has been developed to enhance microstrip antenna designs, particularly for IoT and 5G applications. This model significantly reduces prediction errors, showcasing improved accuracy over traditional methods [12]. The use of AI-driven global optimization techniques, such as TR-SADEA, has enabled efficient design of complex microstrip antenna arrays. This approach has shown promising results in achieving wideband applications with substantial bandwidth and efficiency [13]. Techniques such as Bayesian optimization and genetic algorithms have been utilized to enhance antenna properties, achieving gains of 0.84 to 0.88 [14].

Despite the existence of extensive research that forecasts the dimensions of microstrip antennas, investigations specifically focused on predicting the dimensions of slots within a microstrip antenna, based on various antenna parameters such as gain, frequency, bandwidth, voltage standing wave ratio (VSWR), return loss, and radiation efficiency, remain notably limited. Therefore, this research conducts a comparative analysis of ten machine learning algorithms to predict the dimensions of two slots in a microstrip antenna using a secondary dataset [15]. The most effective algorithm is subsequently optimized to ascertain the superior evaluation metrics, thereby enhancing the accuracy of the prediction outcomes.

2. METHOD

This study adopts a two-phase methodology to develop a predictive model for estimating the slot dimensions of microstrip antennas based on a set of key antenna parameters. In the first phase, a comparative evaluation of multiple machine learning regression algorithms is conducted to identify the most suitable model based on performance metrics such as Mean Absolute Error (MAE) and R-squared (R^2). In the second phase, the best-performing algorithm is further optimized using hyperparameter tuning techniques to enhance its predictive capability and generalization performance.

The dataset used in this research comprises multiple features commonly associated with microstrip antenna characteristics, including frequency, gain, directivity, return loss (S_{11}), radiation efficiency, and voltage standing wave ratio (VSWR). These features serve as the input variables, while the output targets are the geometric dimensions of the antenna slots, namely Slot1_W, Slot1_L, Slot2_W, and Slot2_L. All codes were executed in the Google Colab environment using Python and Scikit-learn libraries for implementation, modeling, and evaluation.

2.1. Comparative Evaluation of Regression Algorithms

The first phase of the research involves evaluating the performance of ten different machine learning regression models to identify the most accurate and robust algorithm for predicting antenna slot dimensions. The models include: Linear Regression, Ridge Regression, Lasso Regression, Elastic Net, Polynomial Regression, Decision Tree Regressor, Random Forest Regressor, Support Vector Regression (SVR), Gradient Boosting Regressor, K-Nearest Neighbors (KNN).

The data preprocessing pipeline starts with uploading the dataset. The dataset is then cleaned by removing missing values, and the features and targets are extracted based on user-defined column names. All input features are standardized using StandardScaler to ensure uniform data scaling, which is especially important for distance-based and regularized models. The dataset is split into training and testing sets using an 80:20 ratio. Each model is implemented using a *multi-output regression* strategy by wrapping the base regressor in Scikit-learn's MultiOutputRegressor, allowing simultaneous prediction of all four target variables. Polynomial Regression is treated as a special case by constructing a pipeline that includes PolynomialFeatures followed by linear regression.
Each model is trained and tested, and evaluated based on five key regression metrics:

- Mean Absolute Error (MAE): measures average absolute error between predicted and actual values.
- Mean Squared Error (MSE) and Root Mean Squared Error (RMSE): penalize larger errors more heavily.
- R-squared (R^2): indicates how much variance in the target variables is explained by the features.
- Explained Variance Score: quantifies how well future samples are likely to be predicted by the model.
- Mean Absolute Percentage Error (MAPE) is also calculated, though handled cautiously to avoid division by zero.

In addition to these evaluation metrics, each model is subjected to 5-fold cross-validation to assess consistency and generalization. The mean and standard deviation of R^2 scores are reported. This comparative analysis serves as a basis for selecting the most promising algorithm, which in this case is Random Forest due to its high R^2 performance across output variables.

2.2. Hyperparameter Optimization of Random Forest

In the second phase of the study, the selected model (Random Forest) is further enhanced through systematic hyperparameter tuning using GridSearchCV. The objective is to identify the optimal configuration of the model's parameters that yields the best predictive performance. A parameter grid is defined to explore the impact of several critical hyperparameters:

- n_estimators: number of decision trees in the forest (100, 200)
- max_depth: maximum depth of each tree (None, 10, 20)
- min_samples_split: minimum number of samples required to split an internal node (2, 5)
- min_samples_leaf: minimum number of samples required to be at a leaf node (1, 2)
- max_features: number of features to consider when looking for the best split ('sqrt')

GridSearchCV is configured with 5-fold cross-validation and R^2 as the scoring metric. This grid search systematically evaluates all parameter combinations to identify the best-performing model configuration. Upon completion, the optimal parameters are extracted, and the best estimator is retrained on the full training set.

Model performance is evaluated using the same set of metrics as before (MAE, MSE, RMSE, R^2, Explained Variance), but now based on predictions from the optimized model. A separate function, plot_results(), is used to create scatter plots that compare predicted and actual values for each output variable, offering an intuitive visualization of model accuracy. This tuning process resulted in noticeable improvements in prediction accuracy and generalization capability, confirming the effectiveness of model optimization in machine learning pipelines.

3. RESULT AND ANALYSIS

p-ISSN: 1907-0012
e-ISSN: : 2714-5417

This study is divided into two phases. The initial phase involves identifying the most suitable machine learning algorithm through a comparative analysis of ten diverse regression methodologies: Linear Regression, Ridge Regression, Lasso Regression, Elastic Net, Polynomial Regression, Decision Tree, Random Forest, Support Vector Regression, Gradient Boosting, and K-Nearest Neighbors (KNN). The subsequent phase involves refining the most effective algorithm. The algorithm exhibiting the highest efficacy is subsequently fine-tuned to enhance its output, as evidenced by improvements in various performance indicators, such as Mean Absolute Error (MAE) and the R-squared value. In this analysis, the performance metrics of the microstrip antenna, derived from the dataset (frequency in GHz, gain, directivity, return loss, radiation efficiency, and VSWR), serve as the input data for the machine learning model. In contrast, the dimensions of the antenna slot (length and width of slot 1, length and width of slot 2) constitute the output data.

3.1. Performance comparison of 10 regression machine learning algorithms

The results of the MAE comparison of 10 regression model algorithms against four output variables can be seen in Figure 1 below.

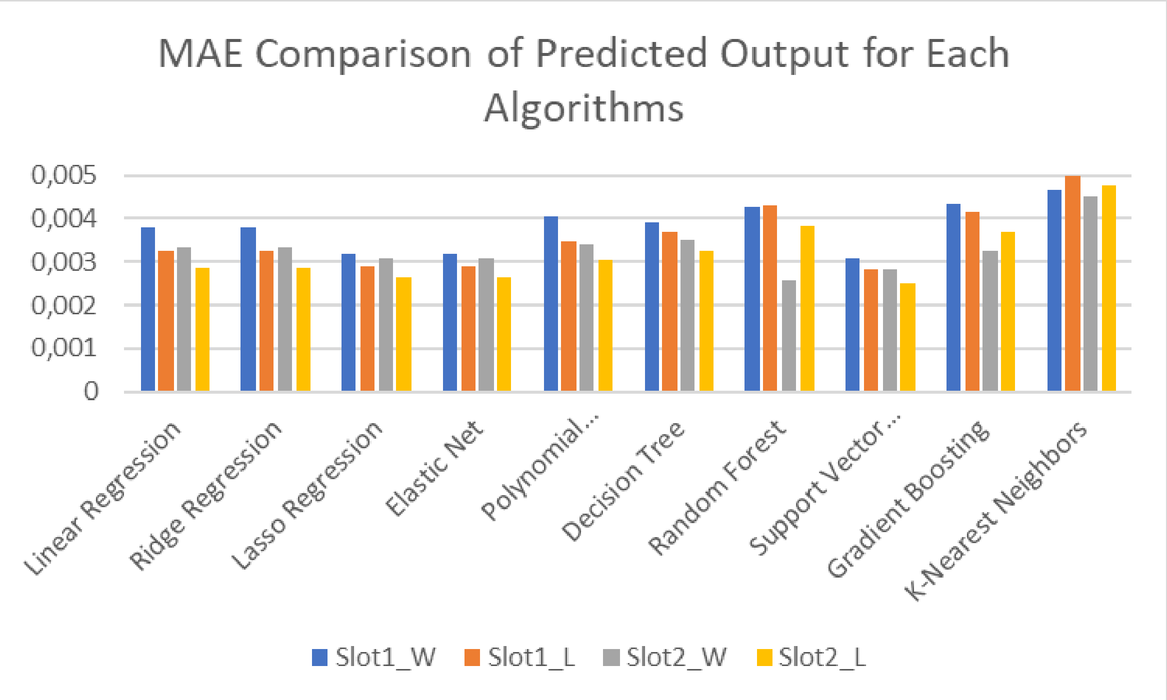

Figure 1. MAE Comparison of Predicted Output for Each Algorithms

The figure presents a bar chart comparing the Mean Absolute Error (MAE) values of predicted outputs across four variables: Slot1_W, Slot1_L, Slot2_W, and Slot2_L. Each output variable was predicted using ten different machine learning algorithms. MAE indicates the average absolute difference between predicted values and actual values, where lower MAE values correspond to higher prediction accuracy. Each algorithm is represented by four bars, with each bar indicating the MAE for one of the output variables. The color scheme is as follows: blue for Slot1_W, orange for Slot1_L, gray for Slot2_W, and yellow for Slot2_L. Overall, the results indicate that Support Vector Regression and Elastic Net demonstrate the best performance, as evidenced by consistently low MAE values—below 0.0035—for all output variables. In contrast, the K-Nearest Neighbors algorithm shows the poorest performance, with MAE values reaching or approaching 0.005 across all outputs, particularly for Slot1_L and Slot2_L. Random Forest and Polynomial Regression also exhibit relatively high MAE values for certain outputs, whereas Linear Regression, Ridge Regression, and Lasso Regression tend to fall in the mid-range. Gradient Boosting demonstrates stable performance, with MAE values ranging from approximately 0.0035 to 0.004. These findings highlight the importance of appropriate algorithm selection, where models such as SVR and Elastic Net can be considered more suitable choices for minimizing prediction errors in similar datasets.

Meanwhile, Figure 2 shows the R-squared (R^2) comparison of 10 regression model algorithms against four output variables.

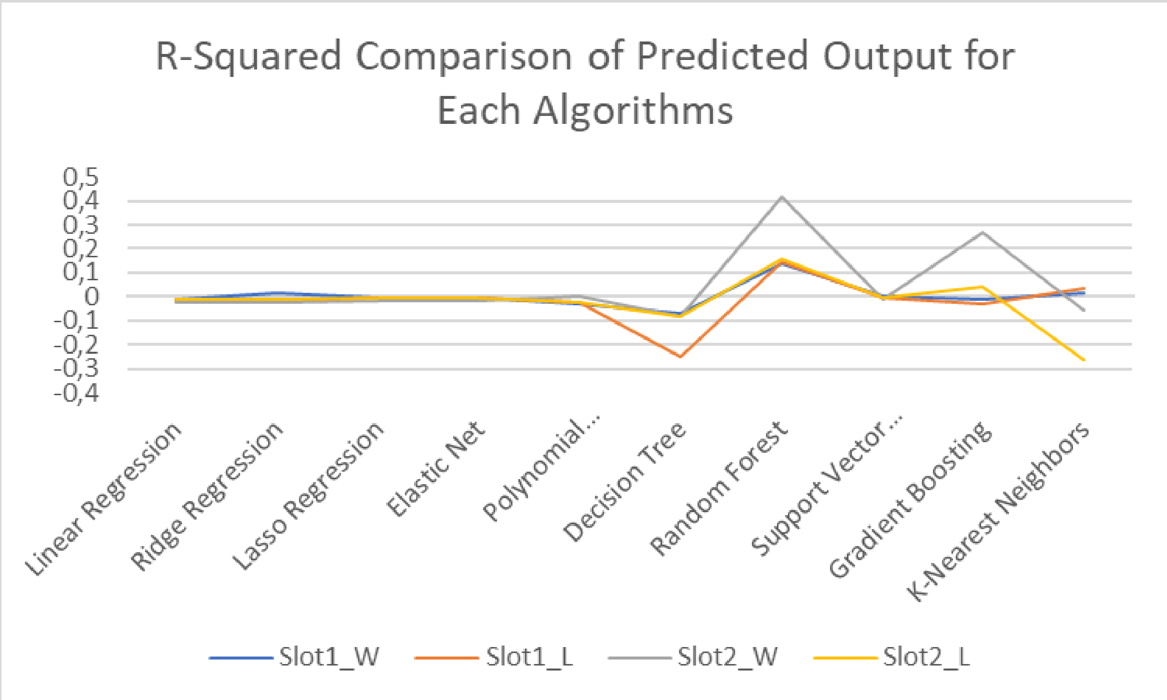

Figure 2. R-Squared (R^2) Comparison of Predicted Output for Each Algorithms

The figure presents a line chart comparing the R-squared (R^2) values of predicted outputs for four different variables using ten machine learning algorithms. The R-squared metric measures the proportion of variance in the dependent variable that is predictable from the independent variables, where a value closer to 1 indicates a stronger predictive relationship. In contrast, values close to or below zero reflect poor model performance, sometimes even worse than a simple mean prediction. From the chart, it is evident that most algorithms result in R-squared values that are either close to zero or negative, suggesting limited effectiveness in modeling the data. Among all models tested, Random Forest stands out as the most effective, achieving the highest R-squared value, particularly for the Slot2_W variable, with a value above 0.4. This algorithm also performs relatively well on Slot1_L, reaching slightly above 0.1. Gradient Boosting follows, showing moderate performance across output variables, especially Slot2_W and Slot2_L. Conversely, the K-Nearest Neighbors algorithm demonstrates the weakest performance, with a highly negative R-squared value for Slot2_L, indicating poor predictive capability. Traditional linear models such as Linear Regression, Ridge Regression, Lasso Regression, and Elastic Net consistently produce R-squared values near or below zero, reflecting their inability to capture the complexity of the data relationships.

Table 1 below shows a comparison of MAE and R^2 values of 10 machine learning regression algorithms.

Table 1. MAE and R^2 Comparison of Predicted Output for Each Algorithms

Algorithms	Slot1_W		Slot1_L		Slot2_W		Slot2_L	
	MAE	R^2	MAE	R^2	MAE	R^2	MAE	R^2
Linear Regression	0,003817	-0,013268	0,003269	-0,015554	0,003317	-0,026021	0,002864	-0,012131

Algorithms	Slot1_W		Slot1_L		Slot2_W		Slot2_L	
	MAE	R^2	MAE	R^2	MAE	R^2	MAE	R^2
Ridge Regression	0,003816	0,013229	0,003269	-0,015537	0,003316	-0,02598	0,002864	-0,012123
Lasso Regression	0,003174	-0,001797	0,00291	-0,002138	0,003069	-0,017207	0,002663	-0,007213
Elastic Net	0,003174	-0,001797	0,00291	-0,002138	0,003069	-0,017207	0,002663	-0,007213
Polynomial Regression	0,004056	-0,032288	0,003464	-0,023193	0,003419	0,003551	0,003028	-0,025004
Decision Tree	0,0039	-0,069467	0,003709	-0,248245	0,003509	-0,080684	0,003269	-0,0791
Random Forest	0,004252	0,136262	0,004289	0,143967	0,002585	0,414614	0,003838	0,158025
Support Vector Regression	0,003091	-0,00115	0,002818	-0,001368	0,002818	-0,011006	0,002515	-0,004614
Gradient Boosting	0,004332	-0,008263	0,004176	-0,032784	0,003275	0,269468	0,003701	0,038662
K-Nearest Neighbors	0,004661	0,013499	0,005	0,031958	0,004515	-0,05702	0,004764	-0,265468

Overall, the result suggests that ensemble-based models like Random Forest are better suited for predicting the given output variables, while simpler or linear models fall short. The use of the R-squared metric in this analysis complements error-based evaluations and reinforces the importance of selecting models that can effectively explain the variability in the data. This study identifies the Random Forest algorithm as the most suitable among the ten algorithms evaluated, thus leading to its selection for subsequent optimization aimed at enhancing performance outcomes.

3.2. Performance of Random Forest Regressor Hyperparameter Tuning

Following the initial evaluation, in which the Random Forest algorithm demonstrated the best performance among the ten regression models compared, the next step involves conducting hyperparameter tuning to further optimize the model's predictive capability. The objective of this tuning process is to enhance both the accuracy and efficiency of the model by exploring the optimal combination of parameters tailored to the characteristics of the dataset. The tuning focuses on several key parameters that significantly influence the model's outcome. The search space for these parameters is summarized in Table 2 below.

Table 2. Hyperparameter Search Space for Random Forest Tuning

Hyperparameter	Values Explored
n_estimators	100, 200
max_depth	None, 10, 20
min_samples_split	2, 5
min_samples_leaf	1, 2
max_features	sqrt

Table 2 outlines the set of hyperparameters and their corresponding values that were explored during the tuning of the Random Forest regression model using Grid Search. The tuning process involved systematically evaluating all possible combinations of selected hyperparameters, which include the number of estimators (n_estimators), maximum depth of the trees (max_depth), minimum number of samples required to split an internal node (min_samples_split), minimum number of samples required to be at a leaf node (min_samples_leaf), and the number of features considered at each split (max_features). This comprehensive search aimed to identify the optimal configuration that yields the best predictive performance while maintaining computational efficiency. As a result of this process, the best-performing parameter combination was found to be: n_estimators = 200, max_depth = 20, min_samples_split = 2, min_samples_leaf = 1, and max_features = 'sqrt'. This configuration was subsequently used to retrain the model and evaluate its final performance.

To assess the impact of the selected hyperparameter configuration, the Random Forest model was retrained using the optimal parameters obtained from the Grid Search. The model's performance was then re-evaluated using the same validation procedure applied in the initial model comparison phase. The evaluation results demonstrate a noticeable improvement in predictive accuracy, indicating that hyperparameter tuning plays a crucial role in enhancing model performance. A detailed comparison between the model's performance before and after tuning is presented in Table 3, highlighting the effect of each optimized parameter on the regression metrics.

Table 3. Comparison of Model Performance Before and After Hyperparameter Tuning

Predicted Output	MAE		R^2	
	Before	After	Before	After
Slot1_W	0,004252	0,0041	0,136262	0,1743
Slot1_L	0,004289	0,0042	0,143967	0,1696
Slot2_W	0,002585	0,0026	0,414614	0,4673
Slot2_L	0,003838	0,0035	0,158025	0,2394

The performance comparison in Table 3 demonstrates the effectiveness of the hyperparameter tuning process in improving the predictive accuracy of the Random Forest model across all target outputs. After tuning, a consistent reduction in Mean Absolute Error (MAE) is observed for all predicted outputs. For instance, the MAE for Slot1_W decreased from 0.004252 to 0.0041, while Slot2_L showed an even more substantial improvement, from 0.003838 to 0.0035. Although the improvement in MAE may appear relatively small in absolute terms, such refinement is meaningful in regression tasks with fine-grained continuous outputs.

In terms of the R^2 metric, all output variables exhibit an increase, indicating better model fit. The Slot2_W output achieved the highest R^2 value after tuning, increasing from 0.4146 to 0.4673. Similarly, Slot2_L improved from 0.1580 to 0.2394, showing a notable gain in the model's explanatory power. Although the overall R^2 scores remain moderate, the positive trend across all outputs confirms that the tuned model captures the underlying patterns in the data more effectively than the initial configuration. These results validate that hyperparameter tuning not only refines prediction precision (as reflected in MAE) but also enhances the model's generalization ability (as indicated by R^2), making the model more reliable for downstream tasks or further deployment.

4. DISCUSSION

The findings from this study provide valuable insights into the application of machine learning models for predicting microstrip antenna slot dimensions based on key antenna parameters. The evaluation process was structured in two main phases: the comparative analysis of ten regression algorithms and the optimization of the best-performing model through hyperparameter tuning.

In the first phase, while Support Vector Regression and Elastic Net demonstrated lower MAE values across all predicted targets, their R-squared (R^2) scores remained relatively low or even negative for several outputs. This indicates that although they yielded smaller average errors, their ability to explain variance in the target data was limited. In contrast, the Random Forest algorithm achieved significantly higher R^2 values, particularly for the Slot2_W output, which reached an R^2 of 0.4146. This suggests that Random Forest was more effective at capturing the underlying relationships between antenna parameters and slot dimensions. Building upon these results, the second phase involved a systematic hyperparameter tuning process to further improve the Random Forest model's performance. The optimization utilized a grid search across several key hyperparameters, including the number of estimators, tree depth, and minimum samples for splitting and leaf nodes. As a result, the best configuration achieved measurable performance improvements.

This improvement is clearly visualized in Figure 3 and Figure 4, which compare the R^2 and MAE values for each output variable before and after tuning.

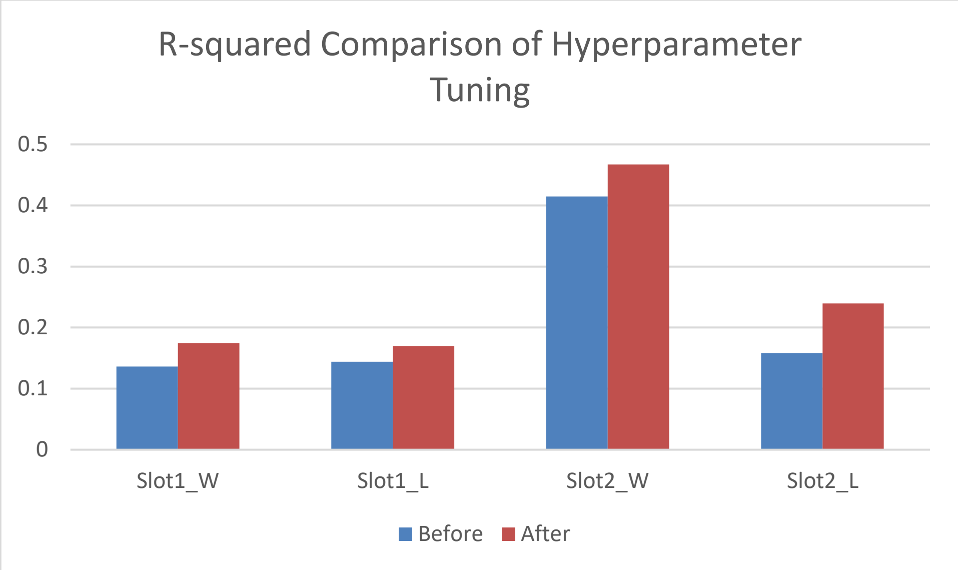

Figure 3. R^2 Comparison Before and After Hyperparameter Tuning for Each Output Variable

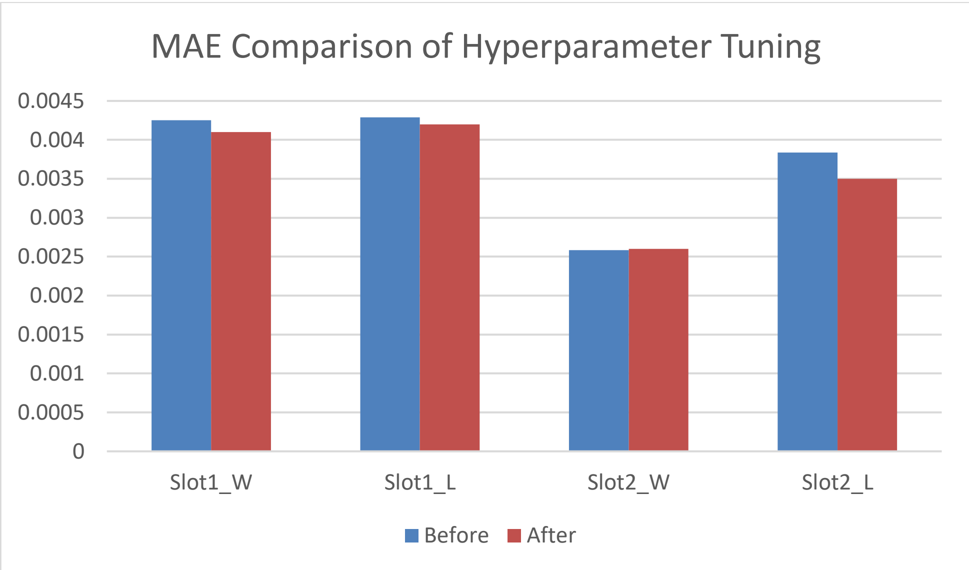

Figure 4. MAE Comparison Before and After Hyperparameter Tuning for Each Output Variable

As shown in Figure 3, the R^2 scores increased for all output variables after tuning. Notably, Slot2_W improved from 0.4146 to 0.4673, while Slot2_L improved from 0.1580 to 0.2394. Even modest gains in Slot1_W and Slot1_L indicate a better overall fit of the model to the data. These results confirm that the tuned model was more capable of explaining the variance in the target variables and likely to generalize better to unseen data. Similarly, as illustrated in Figure 4, MAE values decreased across all target outputs after hyperparameter tuning. For instance, the MAE for Slot2_L decreased from 0.003838 to 0.0035, while Slot1_W and Slot1_L experienced slight but consistent reductions. Although the numerical differences in MAE are relatively small, they are significant in the context of precise antenna design, where millimeter-level accuracy is essential.

Overall, the results affirm that hyperparameter optimization plays a crucial role in refining model performance. The Random Forest algorithm, when fine-tuned, balances both prediction accuracy (low MAE) and explanatory power (high R^2), making it a strong candidate for predictive modeling in antenna design tasks. Grid Search proved to be a systematic and effective approach for identifying optimal hyperparameter settings, with meaningful improvements in performance metrics. The tuned model, with parameters such as n_estimators = 200, max_depth = 20, and max_features = 'sqrt', demonstrated enhanced predictive capabilities compared to its untuned counterpart. This suggests that, in practice, even robust models like Random Forest benefit significantly from well-planned hyperparameter optimization, especially when applied to complex regression problems in antenna design.

5. CONCLUSION

This study has presented a machine learning approach for predicting the slot dimensions of microstrip antennas using a secondary dataset composed of various antenna parameters. A comparative analysis involving ten regression algorithms revealed that Random Forest offered the best overall performance, particularly in terms of R^2 scores, highlighting its suitability for capturing nonlinear patterns in the data.

Subsequent hyperparameter tuning further improved the model's predictive accuracy. The optimized Random Forest regressor showed consistent enhancements in both MAE and R^2 metrics across all predicted outputs, confirming the value of model tuning in practical applications. The results validate that hyperparameter tuning not only minimizes prediction error but also strengthens the model's generalization ability.

The findings suggest promising prospects for integrating machine learning into antenna design workflows, particularly when empirical modeling or analytical derivation becomes complex or infeasible. Future research may consider expanding the dataset to include a wider range of antenna types, exploring other advanced optimization techniques such as Bayesian optimization, or integrating deep learning methods to handle more intricate design variables and constraints.

REFERENCES

[1] A. Nella and A. S. Gandhi, "A survey on microstrip antennas for portable wireless communication system applications," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017, pp. 2156–2165. doi: 10.1109/ICACCI.2017.8126165.

[2] S. Virnave, "Design of Antenna for Multipurpose Wireless Communication: Microstrip Antenna," *International Journal of Science and Research (IJSR)*, vol. 13, no. 5, pp. 869–873, May 2024, doi: 10.21275/sr24514141704.

[3] M. M. Amrita and M. Iyengar, "Microstrip Antennas For Wireless Communication," Sep. 2019, doi: http://dx.doi.org/10.2139/ssrn.3456925.

[4] L. A. de Asis, "Multi Purpose Wireless Communication Antenna Design: Microstrip Antenna," *Journal of Research in Science and Engineering*, vol. 6, no. 12, pp. 8–12, Dec. 2024, doi: 10.53469/jrse.2024.06(12).02.

[5] K. R. Urgunde, "A review on gain & bandwidth enhancement techniques of microstrip patch antenna," *Int J Eng Sci Res Technol*, pp. 684–687, 2014.

[6] S. Ghosh, A. Chakraborty, and S. Chakraborty, "Optimized design of wide band polarization diversified compact microstrip transponder antenna for satellite communication," *Microw Opt Technol Lett*, vol. 65, no. 7, pp. 2066–2075, 2023, doi: https://doi.org/10.1002/mop.33669.

[7] Z. Rasin and N. H. H. Khamis, "Proximity coupled stacked configuration for wideband microstrip antenna," in *2007 Asia-Pacific Conference on Applied Electromagnetics*, 2007, pp. 1–4. doi: 10.1109/APACE.2007.4603938.

[8] R. Dass, A. Kumar, N. Kumar, and R. Yadav, "MICROSTRIP ANTENNA: DESIGN ASPECTS," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 1, no. 5, pp. 130–133, Dec. 2012, [Online]. Available: http://warse.org/pdfs/ijatcse01152012.pdf

[9] P. L. Prasanna Kumar, R. Padmasree, K. Kiran, and B. Sudheer, "Machine learning-driven design and performance analysis of microstrip antennas for sub-6 GHz/mm Wave 5G networks," *International Journal of Informatics and Communication Technology (IJ-ICT)*, vol. 13, no. 3, p. 462, Dec. 2024, doi: 10.11591/ijict.v13i3.pp462-469.

p-ISSN: 1907-0012
e-ISSN: : 2714-5417

[10] K. Muthumeenakshi and S. Radha, "Comprehensive Analysis on the Performance of Antenna Design Using Machine Learning Techniques," 2023, pp. 231–241. doi: 10.1007/978-981-99-1410-4_19.

[11] C.-H. Huang, H.-H. Tsao, C.-C. Hsu, A. Ali, and S.-Y. Liao, "Antenna Structure Prediction and Optimization Based on Machine Learning and Grid Search," in *2024 IEEE International Workshop on Electromagnetics: Applications and Student Innovation Competition (iWEM)*, IEEE, Jul. 2024, pp. 1–3. doi: 10.1109/iWEM59914.2024.10649305.

[12] S. Guntupalli, M. S. Alam, and B. Jackson, "Enhanced Antenna Design Accuracy Using a Hybrid Deep Learning Model," in *2024 IEEE Latin-American Conference on Communications (LATINCOM)*, IEEE, Nov. 2024, pp. 1–6. doi: 10.1109/LATINCOM62985.2024.10770659.

[13] M. O. Akinsolu, K. K. Mistry, B. Liu, P. I. Lazaridis, and P. Excell, "Machine Learning-assisted Antenna Design optimization: A Review and the State-of-the-art," in *14th European Conference on Antennas and Propagation (EuCAP)*, 2020.

[14] K. Raveendra, "Enhancing 5G Sub-3.5 GHz Networks through Optimization of Microstrip Antenna Arrays with Sequential Rotation-Based MIMO using Machine Learning Techniques," *Journal of Electrical Systems*, vol. 20, no. 2s, pp. 370–379, Apr. 2024, doi: 10.52783/jes.1149.

[15] Md. F. H. Mihad, M. M. Islam, and F. H. Sojib, "Antenna Parameters Dataset," 2024, *Mendeley Data*: 2.