Reduce time-consuming collecting fingerprint data in Indoor Positioning Systems with Generated Synthetic Data by Ensemble models and GANs

Prab Wongsekleo[1], Lapat Nakpaen[1], Panarat Cherntanomwong[1]*, and Charnon Pattiyanon[2]
[1]*School of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand*
[2]*CMKL University, Bangkok, Thailand*
Email: 64011583@kmitl.ac.th, 64011750@kmitl.ac.th, panarat.ch@kmitl.ac.th*, charnon@cmkl.ac.th

***Abstract*—Nowadays, the demand for IPS is growing due to the increasing need for accurate indoor location services in applications. The IPS fingerprint techniques are widely popular because they offer high accuracy. However, the process of collecting fingerprint data is labor-intensive and time-consuming. This study aims to alleviate the burden of data collection by generating synthetic data using Machine Learning (ML) and Generative Adversarial Networks (GANs). To create ML synthetic data, we used a dataset containing RSSI values and coordinates. Various regression models were trained using Randomized Search for hyperparameter tuning. The best models were then combined into an ensemble method using Voting Regressor. This ensemble model was used to predict RSSI values for new, synthetic coordinates generated around each reference point, forming the synthetic dataset. We combined synthetic data with actual data from the IPS fingerprint RSSI collecting from the mobile application to create three new datasets with varying ratios of actual to synthetic data from 90:10 to 10:90. These combined datasets were used to train models including Random Forest, Decision Tree, Linear Regression, Gradient Boosting, and K Nearest Neighbors. Our results indicate that models trained on combined datasets significantly reduce the mean distance error (MDE) compared to those trained solely on actual data. This improved performance, however, comes with trade-offs in terms of slightly increased training time, prediction time, and memory usage during both training and prediction phases.**

***Index Terms*—Indoor Positioning Systems (IPS), Generating Synthetic Data, Machine Learning (ML), Generative Adversarial Networks (GANs)**

I. Introduction

Indoor Positioning Systems (IPS) play a crucial role in environments where traditional GPS signals are ineffective, such as inside buildings. These systems are essential for applications like indoor navigation, asset tracking, and emergency response, where precise location information is critical [1], [2]. Among various IPS techniques, the fingerprinting method is widely used due to its ability to achieve high accuracy. This method relies on collecting Received Signal Strength Indicator (RSSI) values from multiple access points at different locations, creating a ”fingerprint” of the environment. However, a significant drawback of this approach is the labor-intensive and time-consuming process of collecting and maintaining up-to-date fingerprint data, especially in large or dynamically changing environments.

Traditional approaches to mitigate the effort involved in fingerprint data collection include using regression-based path loss models, hierarchical KNN methods, and adaptive path loss model interpolation [3]–[6]. While these methods offer certain advantages, they still depend heavily on extensive manual data collection and preprocessing. To address these limitations, recent studies have explored the use of synthetic data generation techniques to augment actual data, thereby reducing the manual effort required for data collection.

Our study proposes an approach to alleviate these challenges by generating synthetic fingerprint data using ensemble ML model and Generative Adversarial Networks (GANs). The synthetic data is generated by training an ensemble of regression models, including Linear Regression, Ridge, Lasso, Decision Tree Regressor, Random Forest Regressor, K Nearest Neighbors Regressor, and XGBRegressor, to predict RSSI values for synthetic coordinates. Additionally, GANs are employed to generate realistic synthetic data that mimics the distribution of actual data. By combining actual and synthetic data in various ratios, our method allows for a more scalable and less labor-intensive IPS data collection process.

We evaluated our approach by combining synthetic data with actual data from the IPS fingerprint RSSI collector mobile application [7] to create three new datasets: Actual + ML Synthetic, Actual + GANs Synthetic, and Actual + ML + GANs Synthetic. Additionally, we varied the ratio of actual to synthetic data from 90:10 to 10:90. These combined datasets were used to train models like Random Forest, Decision Tree, Linear Regression, Gradient Boosting, and K Nearest Neighbors. Our results demonstrate that models trained on combined datasets achieve lower mean distance errors (MDE) compared to those trained solely on actual data, with the optimal performance observed in the 90:10 to 70:30 ratio range.

The remainder of this paper is organized as follows: Section II details the methodology, Section III presents the experimental method, Section IV presents the results and discussion, and Section V is for conclusion.

II. Methodology

Our approach to generating synthetic fingerprint data for IPS involves an combination of ML models and GANs. The actual

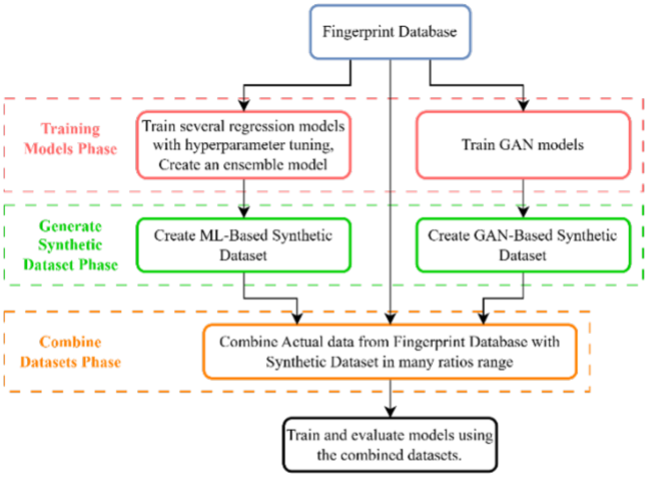
Fingerprint Database
Training Models Phase
Train several regression models with hyperparameter tuning, Create an ensemble model
Train GAN models
Generate Synthetic Dataset Phase
Create ML-Based Synthetic Dataset
Create GAN-Based Synthetic Dataset
Combine Datasets Phase
Combine Actual data from Fingerprint Database with Synthetic Dataset in many ratios range
Train and evaluate models using the combined datasets.

Fig. 1. Overall Process of Methodology

data used was collected using an IPS fingerprint RSSI collector mobile application [7], which provided RSSI values from multiple access points and corresponding coordinates (x, y, z) for each reference point. The collected dataset serves as the foundation for generating synthetic data.

A. An Overview of the Research Methodology

The methodology consists of three main phases:

1) *Training Models Phase:* This phase involves training several regression models with hyperparameter tuning to create an ensemble model, as well as training GANs models.
2) *Generate Synthetic Dataset Phase:* This phase includes creating ML-Based Synthetic Datasets and GANs Based Synthetic Datasets.
3) *Combine Datasets Phase:* This phase involves combining the actual data from the Fingerprint Database with the Synthetic Datasets in various ratios and then training and evaluating models using these combined datasets.

The overall process of our proposed method is depicted in Figure 1.

B. Machine Learning-based Data Synthesis

1) Data Preparation: The dataset containing RSSI values and corresponding coordinates (x, y, z) is prepared. The RSSI values serve as features, and the coordinates serve as targets. The dataset is then splitted into training, validation, and test sets.

2) Model Training: Regression models including Linear Regression, Ridge, Lasso, Decision Tree Regressor, Random Forest Regressor, K Nearest Neighbors Regressor, and XGBRegressor are trained. Hyperparameter tuning is performed using Randomized Search to optimize each model's performance.

3) Ensemble Model: The best-performing models from hyperparameter tuning are combined into an ensemble model using Voting Regressor. This ensemble model predicts RSSI values for synthetic coordinates generated around each reference point, forming the synthetic dataset. The result is the average predicted RSSI value from each model in Ensemble Model.

4) Synthetic Data Generation: Synthetic coordinates are generated around each reference point, with the points generated up to the midpoint between reference points every 0.1 meters, and the ensemble model predicts RSSI values for these coordinates, creating the synthetic dataset.

C. Generative Adversarial Networks (GANs)-Based Synthetic Data Generation

1) Data Preparation and Normalization: The dataset containing RSSI values and coordinates is normalized. RSSI values are scaled to the range $[0, 1]$ after being shifted, and coordinates are normalized using MinMaxScaler.

2) GAN Model Architecutre: The GAN architecture includes a Generator that produces realistic RSSI values and coordinates from random noise, and a Discriminator that distinguishes between real and synthetic data.

3) Training: The GAN architecture employed in this study comprises a Generator and a Discriminator, trained in an adversarial manner. The Adversarial Training Loop involves training the Generator and Discriminator, with the Discriminator minimizing Binary Cross-Entropy Loss [8] and the Generator maximizing the Discriminator's classification error to generate more realistic synthetic data. Both models are optimized using the Adam optimizer [9], which adjusts learning rates to improve convergence.

4) Synthetic Data Generation: The trained Generator produces synthetic RSSI values and coordinates. These values are then inverse-transformed back to their original scale and combined into a final synthetic dataset for further training and evaluation of IPS models.

III. Experimental Methods

A. Data Collection

The actual data used in this study were collected using an IPS fingerprint RSSI collector mobile application [7]. The collected data includes RSSI values from multiple access points and the corresponding coordinates (x, y, z) of each reference point.

B. Dataset Preparation

To assess the impact of synthetic data on model performance, three new datasets were created by combining actual data with synthetic data generated through ML models and GAN. The datasets include:

- *Actual + ML Synthetic:* Combining actual data with synthetic data generated using Machine Learning models.
- *Actual + GAN Synthetic:* Combining actual data with synthetic data generated using GANs.

- *Actual + ML + GAN Synthetic:* Combining actual data with both types of synthetic data (ML and GANs).

The ratio of actual to synthetic data varied from 90:10 to 10:90 to assess the effect of different combinations on model performance.

C. Model Training and Evaluation

Machine learning models, including Random Forest, Decision Tree, Linear Regression, Gradient Boosting, and K Nearest Neighbors, were trained on the combined datasets. The training and evaluation process involved:

1) Model Training: Each model was trained on the combined datasets, with hyperparameters optimized using grid search for maximum performance.

2) Evaluation Metrics: The models were assessed using the following metrics:

- *Mean Distance Error (MDE):* Measures the accuracy of the model's location predictions.
- *Training Time:* Evaluates the computational efficiency during the training process.
- *Prediction Time:* Assesses the model's speed during inference.
- *Memory Usage During Training:* Monitors resource requirements during training.
- *Memory Usage During Prediction:* Tracks resource requirements during inference.

3) Cross-Validation: Cross-validation was employed to ensure robustness in model evaluations, using a 10-fold split to obtain average performance scores.

IV. Results and Discussion

The study utilized a 13th Gen Intel Core i9-13980HX Processor, an NVIDIA GeForce RTX 4060 GPU, and 32GB DDR5-4800 memory to manage processing demands for generating and training models on synthetic datasets. The CPU's high core and thread count allowed efficient parallel processing, while the GPU accelerated GAN training. The large memory capacity ensured faster experimentation and iteration.

A. Synthezied Data Generation

1) Machine-Learning-Based Synthetic Data: The ensemble model for generating ML-based synthetic data comprises several regression models optimized using Randomized Search. The specific parameters identified for each model are shown as in Table I.

The ensemble model's accuracy is assessed by calculating the Mean Absolute Error (MAE) for each predicted RSSI value. The dataset used in this study includes 18 different RSSI values, each corresponding to a different access points within the same network. These values are crucial for accurately determining a device's position within an indoor environment. The results are reported as in Figure 2.

TABLE I
A SUMMARY OF THE PARAMETER SETUP FOR EACH MACHINE LEARNING-MODEL.

Model Name	**Parameters**
Linear Regression	Fit_intercept=True
Ridge	Alpha=0.005522
Lasso	Alpha=0.139493
Decision Tree Regressor	Max_depth=15

Random Forest Regressor	Max_depth=35,
	Max_features=0.963619,
	N_estimators=305,

K Neighbors Regressor	N_neighbors=8

XGBRegressor	colsample_bytree=0.894809,
	Gamma=0.182412,
	learning_rate=0.257210,
	max_depth=9,
	N_estimators=447.

RSSI Value 1: MAE = 3.1210	RSSI Value 10: MAE = 4.8177
RSSI Value 2: MAE = 2.9748	RSSI Value 11: MAE = 3.8708
RSSI Value 3: MAE = 3.3994	RSSI Value 12: MAE = 4.3740
RSSI Value 4: MAE = 2.4971	RSSI Value 13: MAE = 3.5709
RSSI Value 5: MAE = 1.2729	RSSI Value 14: MAE = 4.0229
RSSI Value 6: MAE = 3.5758	RSSI Value 15: MAE = 4.4044
RSSI Value 7: MAE = 3.7703	RSSI Value 16: MAE = 3.5930
RSSI Value 8: MAE = 4.4543	RSSI Value 17: MAE = 4.2355
RSSI Value 9: MAE = 2.5767	RSSI Value 18: MAE = 3.8799

Fig. 2. A summary of the MAE of RSSI values collected from 18 different access points in the target area.

B. GAN-based Synthetic Data Generation

The GANs model successfully generated realistic RSSI values and coordinates. The Generator and Discriminator were trained in an adversarial manner. The accuracy of the GANs model can be visually represented by the distribution of synthetic coordinates, as shown in Figure 3. This data synthesis is aimed to generate data for IPS of two floors in a target building. This plot indicates that the synthetic data closely mimics the spatial distribution of the actual data, suggesting that the GANs model effectively captures the patterns.

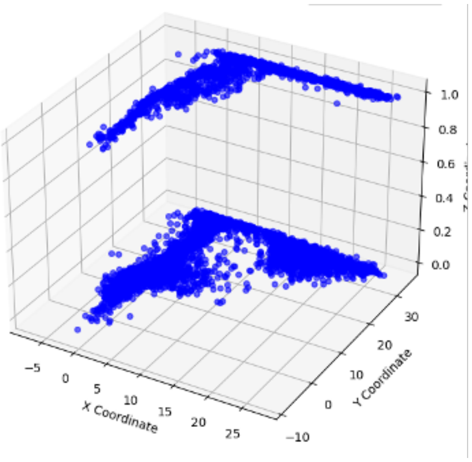

Fig. 3. 3D scatter plot of GANs-based synthetic data, showing an alignment of data points to the actual location.

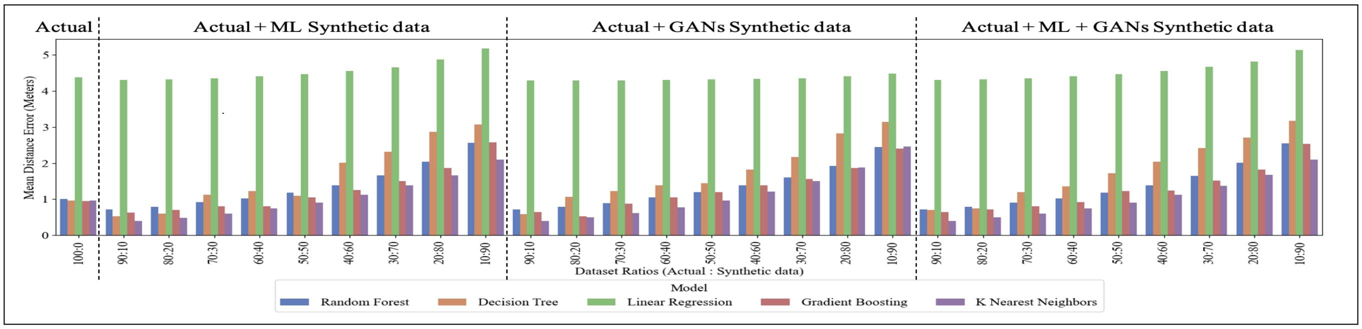

Fig. 4. A comparisong of MDE of each model with different training dataset and actual/synthesis data ratios.

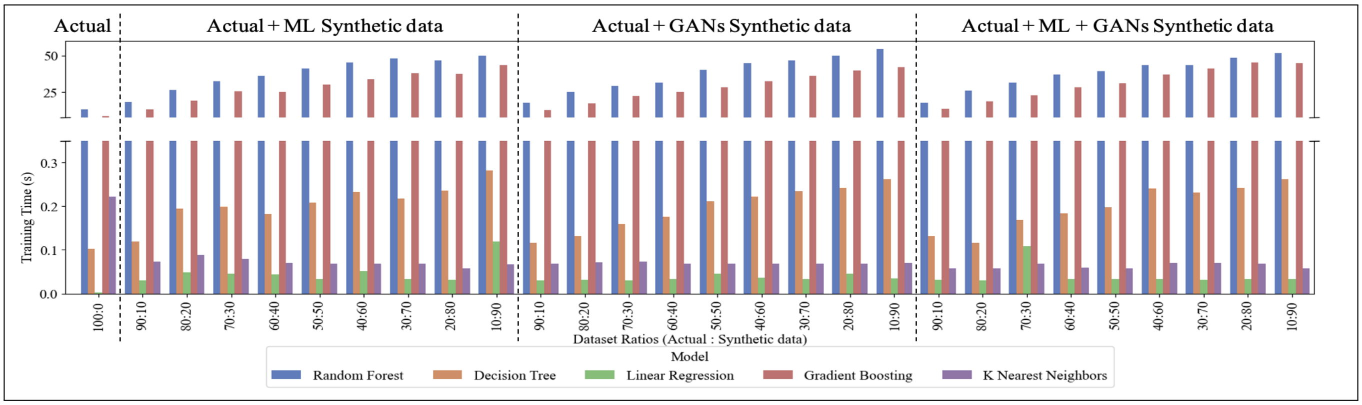

Fig. 5. A comparisong of model training time (in seconds) of each model with different training dataset and actual/synthesis data ratios.

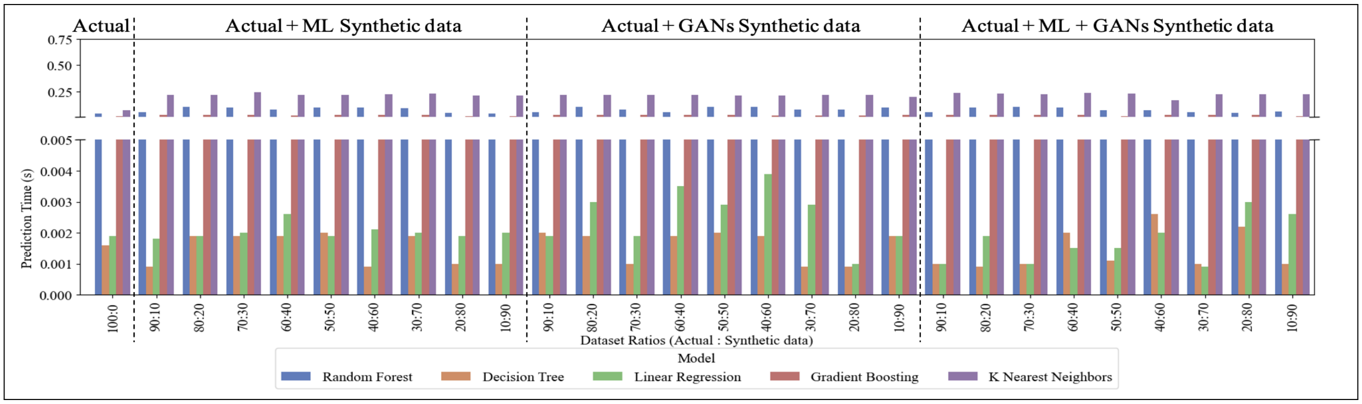

Fig. 6. A comparisong of model prediction/inference time (in seconds) of each model with different training dataset and actual/synthesis data ratios.

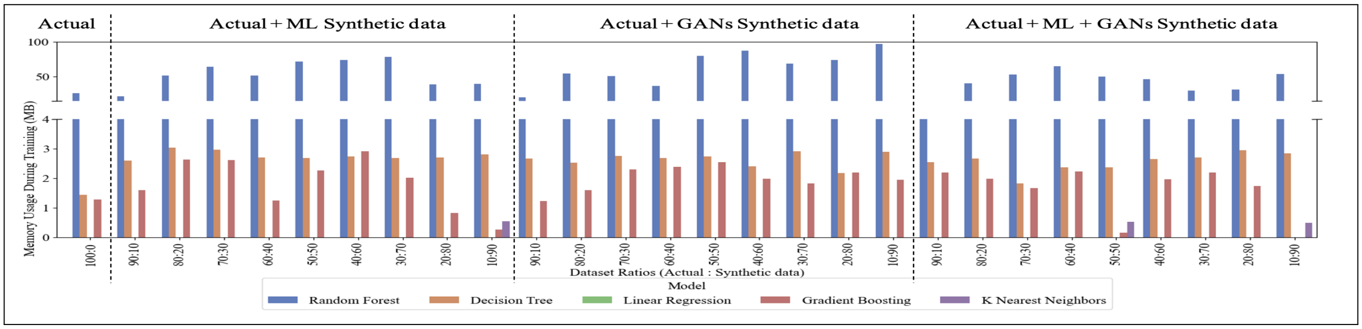

Fig. 7. A comparisong of memory usage for model training (in megabytes) of each model with different training dataset and actual/synthesis data ratios.

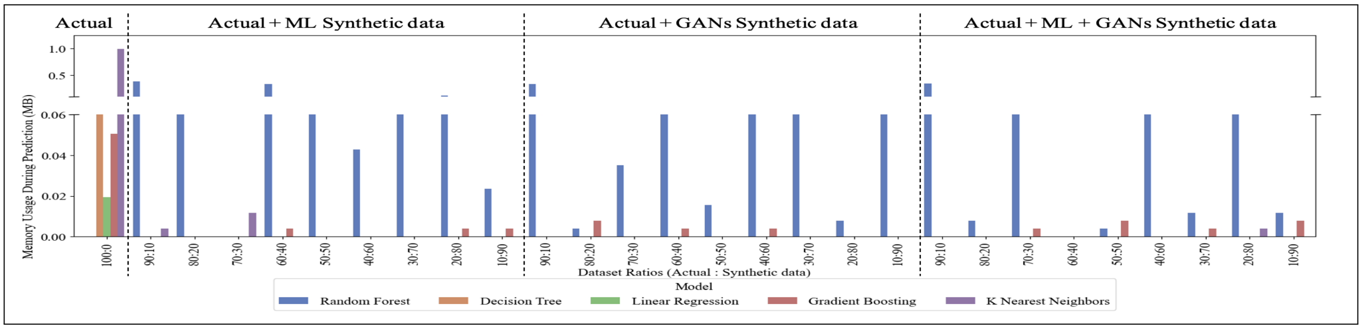

Fig. 8. A comparisong of memory usage for prediction or model inference (in megabytes) of each model with different training dataset and actual/synthesis data ratios.

C. *Training with Different Datasets and Ratios*

The MDE results, illustrated in Figure 4, demonstrate the impact of incorporating synthetic data on model accuracy across various datasets and ratios.

The Random Forest model exhibited substantial improvements in MDE when trained with synthetic data. For the Actual + ML Synthetic dataset, the MDE improvement was 29.27% at a 90:10 ratio, 21.97% at an 80:20 ratio, and 9.56% at a 70:30 ratio. Similar improvements were observed with the Actual + GANs Synthetic dataset, where the MDE improved by 29.69% at a 90:10 ratio, 22.50% at an 80:20 ratio, and 11.61% at a 70:30 ratio. When combining both ML and GANs synthetic data, the Random Forest model achieved improvements of 29.34%, 21.61%, and 10.61% at the respective ratios.

The Decision Tree model also showed notable MDE improvements, particularly with synthetic data at higher ratios. With the Actual + ML Synthetic dataset, the MDE improved by 44.82% at a 90:10 ratio and 37.52% at an 80:20 ratio. However, at a 70:30 ratio, the improvement dropped to 17.04%, indicating a potential overfitting issue as the proportion of synthetic data increased. The Actual + GANs Synthetic dataset showed a similar pattern, with improvements of 39.53% at a 90:10 ratio, but a decrease in performance with improvements of only 11.16% and -28.11% at the 80:20 and 70:30 ratios, respectively. When combining both ML and GANs synthetic data, the Decision Tree model showed improvements of 27.17%, 21.53%, and -25.54% at the respective ratios, indicating that while synthetic data can enhance performance, there is a need to carefully manage the ratio to avoid degradation in model accuracy.

The Linear Regression model exhibited more modest improvements in MDE compared to other models. With the Actual + ML Synthetic dataset, the MDE improved by 1.77% at a 90:10 ratio, but the improvements gradually diminished as the ratio of synthetic data increased, with improvements of 1.24% at an 80:20 ratio and 0.56% at a 70:30 ratio. A similar trend was observed with the Actual + GANs Synthetic dataset, where the MDE improvements were 1.94%, 1.83%, and 0.49% at the respective ratios. Combining both ML and GANs synthetic data resulted in MDE improvements of 1.80%, 1.21%, and 0.48% at the respective ratios.

The Gradient Boosting model benefited significantly from the inclusion of synthetic data. For the Actual + ML Synthetic dataset, the MDE improved by 33.14% at a 90:10 ratio, 25.63% at an 80:20 ratio, and 15.15% at a 70:30 ratio. With the Actual + GANs Synthetic dataset, the model achieved MDE improvements of 31.68% at a 90:10 ratio, 44.39% at an 80:20 ratio, and 7.08% at a 70:30 ratio. The combination of ML and GANs synthetic data resulted in MDE improvements of 32.22%, 25.28%, and 16.01% at the respective ratios, indicating the model's strong ability to leverage synthetic data for improved accuracy.

The KNN model showed the most significant improvements in MDE among all models. For the Actual + ML Synthetic dataset, the MDE improved by 59.06% at a 90:10 ratio, 50.52% at an 80:20 ratio, and 37.25% at a 70:30 ratio. The Actual + GANs Synthetic dataset showed similar improvements, with MDE reductions of 58.84% at a 90:10 ratio, 48.48% at an 80:20 ratio, and 35.79% at a 70:30 ratio. When combining ML and GANs synthetic data, the KNN model achieved MDE improvements of 59.20%, 49.25%, and 37.30% at the respective ratios.

These findings indicate that the integration of synthetic data, particularly in optimal ratios with actual data, can substantially enhance the performance of IPS models. The Random Forest and Gradient Boosting models showed significant improvements in MDE, highlighting their effectiveness in leveraging synthetic data for improved accuracy.

Training time results, depicted in Figure 5, varied significantly across the different models and synthetic data ratios. Ensemble methods like Random Forest and Gradient Boosting required more time to train due to their computational complexity, especially as the proportion of synthetic data increased. For instance, the Random Forest model trained with a 10:90 ratio of actual to ML synthetic data required approximately 49.87 seconds, significantly longer than the 13.19 seconds required for training with actual data alone. This increase in training time can be attributed to the additional variability and noise introduced by synthetic data, which necessitates more computational effort to achieve convergence.

On the other hand, simpler models like Linear Regression and KNN demonstrated relatively stable training times across different ratios. For example, the training time for Linear

Regression with a 10:90 ratio of actual to ML synthetic data was approximately 0.12 seconds, only a slight increase from the 0.03 seconds required for actual data alone. This stability is due to the straightforward nature of these models, which are less sensitive to the increased complexity introduced by synthetic data.

The trends observed in the training time results highlight a clear trade-off between computational resources and model accuracy. While ensemble models like Random Forest and Gradient Boosting achieve higher accuracy with the inclusion of synthetic data, this comes at the cost of increased training time. The choice of synthetic data ratio thus becomes a critical consideration, with higher ratios offering greater accuracy improvements but at the expense of longer training times. For applications where training time is a significant constraint, a more balanced ratio, such as 70:30 or 80:20, may offer a reasonable compromise between accuracy and computational efficiency.

In this context, relying on human effort to collect data for IPS can be reduced using computational power to synthesize data. Manually collecting fingerprint data is labor-intensive and time-consuming, especially in large environments. Machine learning and GAN techniques can facilitate the data generation and drastically overcome the drawbacks of the manual collection. Although there is an increase in training time, the benefits of this trade-off outweigh the time and human labor required. Modern computational resources can handle large datasets and execute complex algorithms with minimal human intervention, making them a more scalable solution.

As shown in Figure 6, the prediction time remained relatively stable across different ratios of actual to synthetic data. The K Nearest Neighbors model exhibited the highest prediction time, consistent with its computational complexity, which involves calculating distances for each prediction. Overall, the inclusion of synthetic data did not significantly impact the prediction time, ensuring that the models maintain efficient real-time performance.

The Memory Usage During Training results, shown in Figure 7, indicate that the Random Forest model consistently consumed more memory across all datasets and ratios, reflecting its complexity as an ensemble method. As the ratio of synthetic data increased, especially in higher ratios like 70:30 or 60:40, memory usage escalated further due to the larger dataset size and added complexity. In contrast, simpler models like Linear Regression and KNN demonstrated relatively stable and low memory usage during training, highlighting their efficiency in handling larger datasets without significantly increasing resource consumption.

The Memory Usage During Prediction results, shown in Figure 8, reveal that memory consumption remained relatively low and stable across most models. The Random Forest model consumed the most memory during prediction, though this was still lower than its memory usage during training. The introduction of synthetic data had a minimal impact on prediction memory usage, with only a slight increase observed in the KNN model due to its reliance on storing the entire training dataset in memory to compute distances. Similarly, the Gradient Boosting model exhibited a slight increase in memory usage, though it remained lower than during training, ensuring efficient real-time application.

V. Conclusion and Future Work

This study explored the use of synthetic data generated by ensemble ML model and GAN to enhance the accuracy of Indoor IPS while mitigating the labor-intensive process of fingerprint data collection. By integrating synthetic data with actual data, models trained on these combined datasets, such as Random Forest and Gradient Boosting, demonstrated significant improvements in MDE compared to models trained solely on actual data. However, these accuracy gains came with increased computational demands, resulting in longer training times and higher resource usage, particularly for more complex models. These findings underscore the importance of balancing accuracy improvements with computational efficiency. Future work could focus on optimizing synthetic data generation and model training to reduce computational costs and further validating this approach in real-world IPS environments.

References

[1] N. M. Tiglao, M. Alipio, R. Dela Cruz, F. Bokhari, S. Rauf, and S. A. Khan, “Smartphone-based indoor localization techniques: State-of-the-art and classification,” *Meas. J. Int. Meas. Confed.*, vol. 179, no. 109349, 2021, doi: 10.1016/j.measurement.2021.109349.

[2] R. S. Naser, M. C. Lam, F. Qamar, and B. B. Zaidan, “Smartphone-Based Indoor Localization Systems: A Systematic Literature Review,” *Electron.*, vol. 12, no. 8, pp. 1–32, 2023, doi: 10.3390/electronics12081814.

[3] D. J. Suroso, F. Y. M. Adiyatma, P. Cherntanomwong, and P. Sooraksa, “Fingerprint Database Enhancement by Applying Interpolation and Regression Techniques for IoT-based Indoor Localization,” *Emerg. Sci. J.*, vol. 4, pp. 167–189, 2022, doi: 10.28991/esj-2021-sp1-012.

[4] F. Y. M. Adiyatma, D. J. Suroso, and P. Cherntanomwong, “Fingerprint Database Enhancement using Spatial Interpolation for IoT based Indoor Localization,” in *Proc. Int. Comput. Sci. Eng. Conf. (ICSEC 2022)*, pp. 192–197, 2022, doi: 10.1109/ICSEC56337.2022.10049367.

[5] F. Y. M. Adiyatma, D. J. Suroso, and P. Cherntanomwong, “Regression-based Path Loss Model Correction to Construct Fingerprint Database for Indoor Localization,” *ACM Int. Conf. Proceeding Ser.*, pp. 181–186, 10.1145/3592307.3592335.

[6] F. Y. M. Adiyatma, S. Sunimit, T. Chokporntaveesuk, K. Lualum, N. Chaisang, and P. Cherntanomwong, “Hierarchical KNN for Smartphone-based 3D Indoor Positioning,” in *Proc. 39th Int. Techn. Conf. on Circuits/Systems, Comput., and Commun. (ITC-CSCC 2024)*, 2-5 July 2024, Okinawa, Japan.

[7] K. Lualum, N. Chaisang, S. Sunimit, and T. Chokporntaveesuk, “Smart phone-based Indoor Localization System within a Multi-Storey Building using WiFi and Machine Learning,” *B.S. Thesis*, School of International and Interdisciplinary Engineering, King Mongkut’s Institute of Technology Ladkrabang, 2024.

[8] I. J. Goodfellow et al., “Generative Adversarial Nets,” *Adv. in Neural Info. Proc. Sys.*, Jun. 2014, doi: 10.1145/3422622.

[9] D. P. Kingma and J. L. Ba, “ADAM: A Method For Stochastic Optimization,” in *Proc. Int. Conf. Learning Rep. (ICLR 2015)*, San Diego, USA, 2015 [Online]. Available: https://arxiv.org/pdf/1412.6980.pdf