Attention-GANs: An Advanced GNSS Data Augmentation Method for Improved NLOS/LOS Classification

Shichuang Nie[1], Hongzhou Yang[1]

[1] Dept. of Geomatic Engineering, University of Calgary, Calgary, AB, Canada - honyang@ucalgary.ca

Keywords: GAN, Attention, Data Augmentation, NLOS, LOS, Classification

Abstract

Global Navigation Satellite System (GNSS) positioning in urban environments remains challenging due to signal obstructions and reflections caused by tall buildings, trees, and overpasses. Non-Line-of-Sight (NLOS) propagation leads to significant positioning errors, making accurate classification of Line-of-Sight (LOS) and NLOS signals essential for robust GNSS performance. Machine learning (ML) techniques have been widely explored for NLOS/LOS classification, yet their effectiveness is constrained by data imbalance, as acquiring labeled NLOS data is more challenging than LOS data. This imbalance reduces model generalization, leading to biased predictions. To address this challenge, we propose an Attention-GAN framework for synthetic GNSS data generation, coupled with a transformer-based encoder to enhance feature extraction. The proposed Attention-GAN incorporates Multi-Head Self-Attention (MHA) in both its generator and discriminator to improve the quality of generated data. Using the UrbanNav dataset, we validate our approach by training various ML classifiers on augmented data and comparing their performance against conventional methods. Experimental results demonstrate that our approach effectively mitigates data imbalance, improves classification accuracy, and enhances GNSS positioning robustness in complex urban environments.

1. Introduction

The Global Navigation Satellite System (GNSS) plays a significant role in positioning and navigation, providing users with global, all-weather and continuous services. However, it is still challenging to achieve high accuracy in urban canyons, where GNSS signals can be inevitably blocked and reflected by tall buildings, trees and overpasses. NLOS propagation occurs when GNSS signals reflect off surfaces before reaching the receiver. The reflected signal takes longer to reach the receiver, leading to errors in distance calculations between the receiver and satellites. Consequently, this results in inaccurate positioning. When the receiver receives both LOS and indirect signals simultaneously, this phenomenon is referred to as Multipath. The receiver is able to deal with part of the multipath effect due to the correlator design, but it does not mitigate the effect of NLOS. In other words, NLOS has a greater impact on the accuracy of positioning in comparison to multipath (Hsu, 2018). Therefore, it is essential to accurately identify and account for NLOS data during the data processing stage to attain robust and accurate positioning in complicated environments.

Extensive research has been undertaken in the domain of NLOS/LOS classification, encompassing various approaches. These can be broadly categorized into antenna-related advancements, advanced receiver algorithms, sensor fusion techniques, 3D building modeling, and machine learning methodologies (Hsu, 2017). Recently, machine learning (ML) methods have gained increasing attention for their ability to process complex GNSS data. Techniques such as Support Vector Machines (SVM) (Jiao et al., 2017), Random Forests (RF) (Zhang and Hsu, 2018), and Decision Trees (DT) (Linty et al., 2019) have been explored for GNSS applications. Additionally, (Xu et al., 2024) employed several ML methods as benchmarks for NLOS/LOS classification. However, addressing the NLOS/LOS identification challenge requires a sufficiently large and well-balanced dataset of labeled NLOS and LOS observations. Moreover, labeling NLOS data is challenging because signal transmission characteristics are not directly observable. For the UrbanNav dataset, the Azimuth and Elevation angles of satellites combined with 3D models are used to identify the signal transmission types (Hsu et al., 2023) (Hsu et al., 2021). However, it is time consuming and difficult to acquire large amounts of labeled NLOS data. As a result, the amount of labeled NLOS data is much less than LOS data. The data constraint issue can lead to bad generalization performance of machine learning models (Sun et al., 2009) (Ganganwar, 2012), as they may struggle to learn robust features for NLOS classification. This class imbalance can cause models to be biased toward LOS predictions, reducing their reliability in urban environments where NLOS conditions are prevalent. (Zhou et al., 2024) proposed a Hopular based model in NLOS/LOS classification, achieving high classification and positioning accuracy using relatively small to medium-sized datasets but still faced imbalanced data issues. Some teams used generative models to deal with the imbalanced dataset in other fields (Song et al., 2018) (Tran et al., 2022), which shows the potential of generative model in the GNSS domain.

Generative models offer promising solutions for image style transfer, text generation, data simulation and augmentation, which can be utilized to mitigate the challenges associated with data constraints. Compared to other models, generative models have the unique capability to generate realistic data samples, a feature that most traditional models lack. This ability allows them to overcome data scarcity by producing high-quality synthetic data that closely mimics real-world observations. Generative models can be categorized into two types: explicit density estimation and implicit estimation (Creswell et al., 2018). Explicit density estimation requires defining and solving the distribution of the data, while implicit estimation involves sampling from the data without explicitly defining it. For sampling, they can be categorized based on whether they employ Markov chains. However, Markov chains pose challenges in achieving convergence, which means the model need to run for sufficient time to converge (Creswell et al., 2018). Popular generative models include: Diffusion Models (Ho et al., 2020) which iteratively refine noisy inputs into structured outputs, Autoencoders (particularly Variational Autoencoders) (Gm et al., 2020) which encode data into latent representations and reconstruct or generate new samples, and Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) which utilize adversarial training between a generator and a discriminator to produce realistic samples. Given the complexity of GNSS data, the need for large datasets, and the difficulty of explicitly defining data distributions, an implicit generative model that does not rely on Markov assumptions is preferred. GANs exhibit these characteristics, making them a suitable choice for addressing data imbalance in GNSS positioning applications.

In this research, a novel Attention-GAN model with a transformer-based encoder is proposed for data augmentation. The attention mechanism (Vaswani et al., 2023) is introduced to enhance feature representation by focusing on crucial spatial and temporal dependencies in GNSS signals, thereby improving the quality of generated samples and reducing noise. The implementation utilizes the UrbanNav dataset from the Hong Kong Polytechnic University on Kaggle, which contains GNSS observation features along with corresponding NLOS/LOS labels. After addressing the data imbalance, several baseline machine learning classification models, including including SVM, RF, Logistic Regression (LR) and Gradient Boosting (GB), will be employed to evaluate the effectiveness of the proposed approach compared to benchmark methods.

2. Methodology

The purpose of the Attention-GAN is to simulate and generate GNSS data to balance the training dataset. Figure 1 shows the work flow of the Attention-GAN. Randomly generated noise following a normal Gaussian distribution is first fed into the generator to produce simulated data. The generated data is then integrated with the scaled real training data to form the encoded augmented dataset for training.

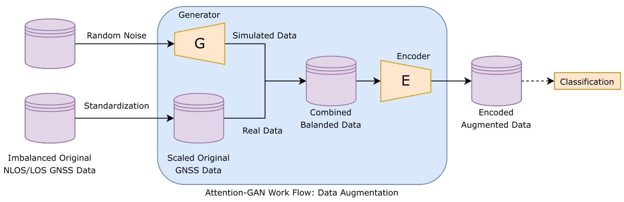

Figure 1. Attention-GAN Workflow.

In this section, the Attention-GAN training process along with the structure of the GAN and the encoder is first introduced. This is followed by the validation of the model including the steps involved in generating synthetic NLOS samples and integrating them into the dataset and how the combined dataset is used to train various machine learning models. The effectiveness of the proposed approach is then assessed using multiple evaluation metrics, ensuring a comprehensive analysis of its impact on classification performance.

2.1 Attention-GAN Training

The training data is sourced from the UrbanNav dataset, which is available at . This dataset consists of two Excel files: one containing 74,086 entries, which is used for training, and another containing 36,189 entries which is used for validation. Each entry represents an observation from a specific satellite. Each observation includes 16 features, but for this experiment, six key features were selected: elevation angle, azimuth angle, carrier-to-noise ratio (C/N_0), pseudorange residual, root of sum of square error (RSSE), and standard deviation of pseudorange error. The training dataset is used to train both the encoder and the GAN. Meanwhile, a portion of the test dataset is selected and modified to create an imbalanced NLOS/LOS ratio, which is then used to validate the proposed method's effectiveness in handling class imbalance.

2.1.1 Transformer-Based Encoder Training: The transformer-based encoder leverages a multi-head self-attention mechanism, a feedforward network, and layer normalization with residual connections to effectively capture complex dependencies within the input data.

The multi-head self-attention mechanism is a key component that enhances the model's ability to identify intricate relationships between different features within the input data. Instead of computing a single attention score for each input element, this mechanism employs multiple attention heads, each learning to focus on different aspects of the input. This allows the model to capture diverse dependencies across features, improving its ability to extract meaningful representations. In detail, each input feature vector is linearly transformed into three different vectors: Query (Q), Key (K) and Value (V). If we have n inputs with d_k dimensions, the Q, K and V are all in the shape of $n * d_k$. The results are calculated as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \tag{1}$$

where d_k = the dimensionality of the key vectors

The scaling by $\sqrt{d_k}$ prevents overly large dot-product values which can lead to unstable gradients. Instead of using a single set of Q, K, and V, the model applies multiple attention heads in parallel, each with independent learned weights.

The outputs from all attention heads are concatenated and projected back into the original feature space with the feedforward network, allowing the model to integrate multiple perspectives on feature interactions. The attention mechanism plays a crucial role in identifying relationships between different elements of the input, enabling the model to focus on the most relevant features while preserving contextual information. In the case of GNSS data, it helps the encoder emphasize key attributes such as signal strength variations, geometric relationships (e.g., azimuth and elevation angles), and error characteristics, ensuring that significant patterns are effectively captured. By leveraging multi-head self-attention along with residual connections and normalization, the transformer-based encoder enhances its ability to generalize across different GNSS observations, making it well-suited for improving classification performance in NLOS/LOS scenarios.

The procedure of the encoder's training is shown in Figure 2. The sixteen features are firstly scaled by StandardScaler. This ensures that all features contribute equally to the model and prevents any one feature from dominating due to differing scales. The StandardScaler is a commonly used data normalization technique that transforms each feature to have a mean of

0 and a standard deviation of 1. It achieves this by applying the following transformation to each feature:

$$X_{scaled} = \frac{X - \mu}{\sigma}, \tag{2}$$

where X = original feature value
μ= mean of the feature across all samples
σ = the standard deviation of the feature

After scaling, the processed data is passed into a transformer encoder-based network which contains multi-head attention mechanism for feature extraction. Following this, a fully connected linear layer is applied to map the extracted representations back to the original feature space, reconstructing the input data. To evaluate the effectiveness of the encoder, the reconstructed data is compared to the original standardized features using Mean Squared Error (MSE) loss. This loss function quantifies how well the encoder preserves essential feature information. Lower MSE values indicate that the encoder successfully retains the key characteristics of the GNSS features. Once the encoder is trained and optimized, it is used to encode the entire scaled training dataset to train the machine learning models for NLOS/LOS classification.

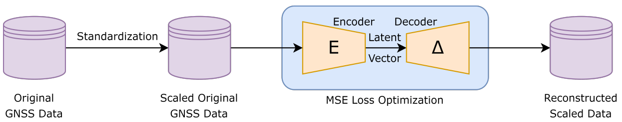

Figure 2. Encoder Training Procedure.

2.1.2 GAN Training The GAN contains two parts: the generator with multi-head attention (MHA) and discriminator with MHA. The generator is designed to transform an input noise vector into a meaningful high-dimensional representation, which mimics the data distribution. The noise vectors are sampled from a standard normal distribution to ensure a diverse range of initial latent vectors.

The generator architecture transforms a random noise vector, sampled from a standard normal distribution into synthesized data through a series of structured operations. Initially, the noise input is processed through three fully connected layers with progressively increasing dimensionality, each followed by a Rectified Linear Unit (ReLU) activation function to introduce non-linearity and facilitate complex feature extraction. The output is then reshaped and passed through a multi-head self-attention layer, which captures long-range dependencies in the feature space by computing self-attention scores across multiple attention heads. This attention mechanism, configured with an embedding dimension of 256 and four attention heads, enhances feature representation by preserving contextual relationships. Subsequently, the attended features are processed through a fully connected layer to generate the synthesized data. To ensure numerical stability and improve training dynamics, the generated output is standardized using a StandardScaler, which normalizes feature distributions and facilitates model convergence (Ahsan et al., 2021).

The discriminator acts as a binary classifier, distinguishing between real and generated samples. The input data first undergoes three fully connected layers with decreasing dimensions, each followed by a LeakyReLU activation function with a negative slope of 0.2. This allows for efficient gradient flow while mitigating issues related to vanishing gradients (Xu et al., 2020). Then the extracted features are passed through a multi-head self-attention layer with an embedding dimension of 64 and four attention heads. This mechanism enables the discriminator to focus on the most critical features, improving its ability to detect generated samples. The final layer consists of a fully connected layer with a sigmoid activation function, producing a probability score that indicates the likelihood of the input belonging to the real dataset.

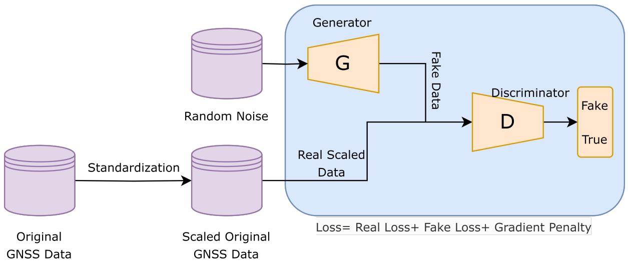

Figure 3. Attention-GAN Training Procedure.

The training procedure is shown in Figure 3. To train the Attention-GAN, the raw features are scaled by StandardScaler first. The scaled data is then used to train the GAN, consisting of the generator and the discriminator. The discriminator is optimized using real encoded samples and synthetic samples generated by the generator, with a loss function that encourages the discriminator to assign higher scores to real data and lower scores to fake data. Traditional GANs use a binary cross-entropy loss to minimize the difference between two distributions, equivalent to minimizing the Kullback-Leibler (KL) Divergence. The discriminator is minimizing:

$$\begin{aligned} L^D\left(X_r, X_g\right) =& -\mathbb{E}_{x_r \sim X_r}\left[\log\left(D\left(x_r\right)\right)\right] \\ & -\mathbb{E}_{x_g \sim X_g}\left[\log\left(1 - D\left(x_g\right)\right)\right] \end{aligned} \tag{3}$$

And the generator is minimizing:

$$L^D\left(X_r, X_g\right) = -\mathbb{E}_{x_g \sim X_g}\left[\log\left(D\left(x_g\right)\right)\right] \tag{4}$$

where D is the discriminator function
x_r is sampled from the real distribution X_r
x_g is sampled from the generated distribution X_g

To address issues such as vanishing gradients, which may cause the generator to stop learning, and mode collapse, where the generator produces only a limited set of outputs instead of capturing the full data distribution (Huang and Jafari, 2023)(Mariani et al., 2018), Wasserstein GAN is introduced. This approach utilizes the Wasserstein distance to mitigate these challenges, helping to stabilize training and improve output diversity(Gulrajani et al., 2017). Wasserstein distance, also known as Earth-Mover Distance (EMD), quantifies the minimum cost of transporting a mass from one distribution to another. EMD is a continuous and differentiable function, ensuring that its gradients are always meaningful and contributing to the stability of GAN training (Huang and Jafari, 2023). Aligned with the theory of WGAN, the generator will eventually converge to the performance of the discriminator. Consequently, WGAN necessitates a deep architecture for the discriminator. The EMD distance is defined as:

$$W\left(X_r, X_g\right) = \inf_{\gamma \sim \Pi\left(X_r, X_g\right)} \mathbb{E}_{\left(x_r, x_g\right) \sim \gamma} \left\| x_r - x_g \right\|, \quad (5)$$

where $\Pi\left(X_r, X_g\right)$ denotes all the joint distributions between the real distribution X_r and the generated data distribution X_g. However, it is not feasible to try all pairs to find the smallest EMD. By using Kantorovich-Rubinstein duality, it is equivalent to find the upper bound in:

$$\begin{aligned} W\left(X_r, X_g\right) = \sup_{\|D\|_L \leq 1} & \left(\mathbb{E}_{x_r \sim X_r}\left[D\left(x_r\right)\right]\right. \\ & \left. -\mathbb{E}_{x_g \sim X_g}\left[D\left(x_g\right)\right]\right), \end{aligned} \quad (6)$$

where $||D||_L <= 1$ ensures that D belongs to the space of 1-Lipschitz functions. In the absence of this constraint, the discriminator's objective function simplifies to maximize:

$$W^{(D)}\left(X_r, X_g\right) = \mathbb{E}_{X_r \sim X_r}\left[D\left(x_r\right)\right] - \mathbb{E}_{x_g \sim X_g}\left[D\left(x_g\right)\right], \quad (7)$$

In contrast to the original GAN, the discriminator in WGAN employs an unconstrained real number as the criterion for evaluating the quality of real/fake data, rather than classification probability. Additionally, a gradient penalty (GP) is applied to enforce Lipschitz continuity, ensuring stable training (Gulrajani et al., 2017). The 1-Lipschitz constraint equals to ensure the normality of gradients $\|\nabla_x D(x)\|_2$ everywhere. The GP is:

$$GP = \mathbb{E}_{x \sim X}\left[\left(\|\nabla_x D(x)\|_2 - 1\right)^2\right], \quad (8)$$

,

As a result, with the extra gradient penalty term, the discriminator is to minimize:

$$\begin{aligned} W^{(D)}\left(X_r, X_g\right) = & \mathbb{E}_{x_r \sim X_r}\left[D\left(x_r\right)\right] - \mathbb{E}_{x_g \sim X_\xi}\left[D\left(x_g\right)\right] \\ & + \lambda \mathbb{E}_{\hat{x} \sim \hat{X}}\left[\left(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\right)^2\right] \end{aligned} \quad (9)$$

Since gradient penalty is only applied into the discriminator loss, the loss function for generator remains the same as equation 4 (Huang and Jafari, 2023). The generator is updated by minimizing the negative discriminator score on generated samples, effectively learning to produce realistic feature distributions. The training follows an alternating approach where the discriminator is trained for multiple iterations per generator update, stabilizing the learning process. In the real experiment though the WGAN-GP is applied, the training process is still unstable. This may be due to the high variance in real GNSS data, which makes it difficult for the discriminator to enforce a smooth decision boundary. To address this, gradient clipping is applied to prevent exploding gradients throughout training (Zhang et al., 2020), ensuring the norm of the gradients of the discriminator does not exceed $max_{norm} = 1.0$. If the gradient norm is greater than 1.0, all gradients are scaled down proportionally to keep the norm within the limit.

2.2 Validation On the Machine-Learning Models

To evaluate the effectiveness of the transformer-based encoder and the attention-GAN, two experimental setups were designed. First, a baseline evaluation was conducted using traditional machine learning models, including SVM, RF, LR and GB. These models were trained on a dataset comprising 2,000 randomly selected Line-of-Sight (LOS) samples and 200 Non-Line-of-Sight (NLOS) samples, which is imbalanced to establish benchmark performance. While the transformer-based encoder was applied to encode the dataset before training, as shown in Figure 4. The second experiment introduced the Attention-GAN, which was used to generate 800 synthetic NLOS samples to balance the dataset. Finally, the augmented dataset, incorporating both real and synthetic data, is used to train the previously mentioned machine learning models. These two experiments effectively evaluate the capability of the Attention-GAN in mitigating class imbalance and improving classification performance.

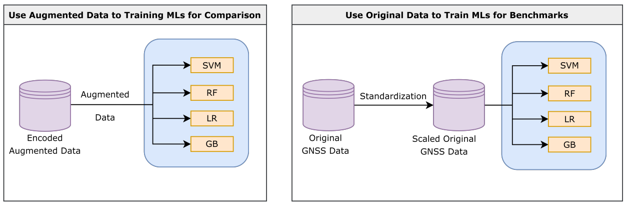

Figure 4. Augmented Data Validation Procedure.

For NLOS/LOS classification, four evaluation metrics are employed to assess the model's effectiveness. The first metric is overall accuracy, which measures the proportion of correctly classified samples out of the total dataset. Although accuracy provides a straightforward measure of model performance, it may not fully reflect classification effectiveness when dealing with imbalanced datasets, as it can be dominated by the majority class. Therefore, additional metrics such as recall, precision, and F1-score are used to provide a more comprehensive evaluation.

One of these metrics is recall, which measures the model's ability to correctly identify all relevant instances within a specific class. In the context of NLOS/LOS classification, recall for the NLOS class indicates the proportion of actual NLOS samples that are correctly classified as NLOS, while recall for the LOS class reflects the proportion of actual LOS samples that are correctly identified as LOS. A higher recall value suggests that the model effectively captures more instances of the target class, reducing the likelihood of misclassification and improving overall detection performance. And precision measures the proportion of correctly predicted instances of a class among all instances predicted as that class.

F1-score is a balanced metric that takes both precision and recall into account, providing a more comprehensive evaluation of classification performance. It is calculated as the harmonic mean of precision and recall, ensuring that both false positives and false negatives are considered in the assessment. In the context of NLOS/LOS classification, a high F1-score indicates that the model not only correctly identifies a large proportion of NLOS and LOS instances (high recall) but also minimizes incorrect classifications (high precision). This metric is particularly useful when dealing with imbalanced datasets, as it assesses the model's performance comprehensively when faced with biased datasets.

3. Results and Analysis

The benchmarks where machine learning models are trained solely on the imbalanced original dataset are shown in Table 1. In comparison, the SVM and RF models trained on the GAN-augmented data experience an improvement in overall accuracy, while the LR and GB models trained on the GAN-augmented data show a decline. However, overall accuracy alone does not fully capture the model's performance. When the training data is highly imbalanced, the model may struggle to classify underrepresented categories. For instance, in the original dataset, the SVM model achieved an accuracy of 0.67, yet its recall for the NLOS class was 0, meaning it completely failed to detect NLOS instances. This justifies the introduction of additional evaluation metrics.

Specifically, for NLOS classification, although precision generally decreases across models, recall and F1-score significantly improve. For example, the SVM's recall for NLOS increases from 0.00 to 0.29, and the RF's recall for NLOS improves from 0.22 to 0.46, an increase of 24 percentage points, which is approximately 109% relative to the benchmarks, indicating a stronger ability to identify NLOS cases. Additionally, the corresponding F1-scores improve by 0.39, 0.19, 0.60, and 0.60 for SVM, RF, LR, and GB, respectively. For the last three models, these improvements correspond to relative increases of approximately 0.53%, 0.18%, and 0.15% compared to their initial values, reflecting a more balanced overall performance across all four models. Similarly, for LOS classification, most models maintain high recall and f1-score, with some improvement in precision. These results suggest that the adjustments in training help models achieve better generalization, particularly in recognizing NLOS instances, thereby enhancing overall classification effectiveness.

Models	Overall Accuracy	NLOS/LOS Metrics		
		Precision	Recall	F1-score
SVM	0.67	0.67/0.67	0.00/0.10	0.00/0.80
RF	0.74	0.97/0.73	0.22/0.10	0.36/0.84
LR	0.73	0.97/0.72	0.20/0.10	0.33/0.84
GB	0.75	0.93/0.73	0.26/0.99	0.40/0.84

Table 1. Original Selected Dataset Training Results

Models	Overall Accuracy	NLOS/LOS Metrics		
		Precision	Recall	F1-score
SVM	0.71	0.61/0.73	0.29/0.91	0.39/0.81
RF	0.76	0.69/0.77	0.46/0.90	0.55/0.83
LR	0.72	0.70/0.73	0.27/0.94	0.39/0.82
GB	0.73	0.67/0.74	0.35/0.92	0.46/0.82

Table 2. Augmented Dataset Training Results.

4. Conclusion and Future Work

In this work, we propose a novel attention-GAN together with an encoder for GNSS data augmentation to address data constraint issues in deep learning applications. A multi-head attention mechanism is applied to the encoder, generator and discriminator to extract more representative features from the data. For training, we employ the GAN with a penalty term and gradient clipping to stabilize the training process and achieve better results.

In the classification task of NLOS and LOS, the GAN is used to generate synthetic data to mitigate dataset imbalance, as NLOS samples are significantly underrepresented compared to LOS. To evaluate the effectiveness and quality of the generated data, we train multiple machine learning models including SVM, RF, LR and GB models on both the augmented and original datasets for comparison. The results demonstrate that the models trained on the augmented dataset achieve superior performance in classifying NLOS data in the test set, improving by 39%, 19%, 6% and 6% in f1-score for NLOS respectively. The results highlight the potential of generative models in the GNSS domain.

Future work will focus on redesigning the GAN architecture to better align with GNSS data characteristics. Furthermore, a more diverse dataset—including dynamic data, receiver-level signals, and other variations—will be incorporated for training. Beyond NLOS/LOS classification, the generated synthetic data has broad applicability across various domains, especially when needing a large-scale and diverse dataset. For instance, simulated GNSS data can be leveraged to generate large-scale datasets tailored to diverse environmental conditions, addressing data scarcity issues in real-world applications. This capability is particularly beneficial for training end-to-end GNSS positioning models, enhancing their robustness and generalization across different urban and rural scenarios. Additionally, synthetic GNSS data can support the development of autonomous driving models by providing diverse and realistic navigation data, improving the reliability of localization and sensor fusion algorithms in challenging conditions.

5. Acknowledgment

Thanks Penghui Xu from Hongkong Poly University for providing the training and testing dataset. Thanks my teammate Paul Dobre for refining the writing. And thanks the Natural Sciences and Engineering Research Council of Canada (NSERC) for providing the funding for my research.

References

Ahsan, M. M., Mahmud, M. P., Saha, P. K., Kishor, D. G., Siddique, Z., 2021. Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance. *Technologies*, 9(3), 52. https://www.mdpi.com/2227-7080/9/3/52.

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A. A., 2018. Generative Adversarial Networks: An Overview. *IEEE Signal Processing Magazine*, 35(1), 53–65. http://dx.doi.org/10.1109/MSP.2017.2765202.

Ganganwar, V., 2012. An Overview of Classification Algorithms for Imbalanced Datasets. *International Journal of Emerging Technology and Advanced Engineering*, 2(4), 42–47.

Gm, H., Gourisaria, M. K., Pandey, M., Rautaray, S. S., 2020. A Comprehensive Survey and Analysis of Generative Models in Machine Learning. *Computer Science Review*, 38.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial networks.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A. C., 2017. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

Ho, J., Jain, A., Abbeel, P., 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 6840–6851.

Hsu, L.-T., 2017. Gnss multipath detection using a machine learning approach. *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Yokohama, 1–6.

Hsu, L.-T., 2018. Analysis and Modeling GPS NLOS Effect in Highly Urbanized Area. *GPS Solutions*, 22(7).

Hsu, L.-T., Huang, F., Ng, H.-F., Zhang, G., Zhong, Y., Bai, X., Wen, W., 2023. Hong Kong UrbanNav: An Open-Source Multisensory Dataset for Benchmarking Urban Navigation Algorithms. *NAVIGATION: Journal of the Institute of Navigation*, 70(4).

Hsu, L.-T., Kubo, N., Wen, W., Chen, W., Liu, Z., Suzuki, T., Meguro, J., 2021. Urbannav: An open-sourced multisensory dataset for benchmarking positioning algorithms designed for urban areas. *Proceedings of the 34th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2021)*, 226–256.

Huang, G., Jafari, A. H., 2023. Enhanced balancing GAN: Minority-class image generation. *Neural computing and applications*, 35(7), 5145–5154.

Jiao, Y., Hall, J. J., Morton, Y. T., 2017. Automatic Equatorial GPS Amplitude Scintillation Detection Using a Machine Learning Algorithm. *IEEE Transactions on Aerospace and Electronic Systems*, 53(1), 405–418.

Linty, N., Farasin, A., Favenza, A., Dovis, F., 2019. Detection of GNSS Ionospheric Scintillations Based on Machine Learning Decision Tree. *IEEE Transactions on Aerospace and Electronic Systems*, 55(1), 303-317.

Mariani, G., Scheidegger, F., Istrate, R., Bekas, C., Malossi, C., 2018. Bagan: Data augmentation with balancing gan. *arXiv preprint arXiv:1803.09655*.

Song, B., Li, S. L., Tan, M., Ren, Q. H., 2018. A Fast Imbalanced Binary Classification Approach to NLOS Identification in UWB Positioning. *Mathematical Problems in Engineering*, 2018.

Sun, Y., Wong, A. K., Kamel, M. S., 2009. Classification of Imbalanced Data: A Review. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(4), 687–719.

Tran, D. H., Chung, B., Jang, Y. M., 2022. Gan-based data augmentation for uwb nlos identification using machine learning. *Proceedings of the 2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, IEEE, Jeju Island, Korea, Republic of, 417–420.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I., 2023. Attention is all you need.

Xu, J., Li, Z., Du, B., Zhang, M., Liu, J., 2020. Reluplex made more practical: Leaky relu. *2020 IEEE Symposium on Computers and communications (ISCC)*, IEEE, 1–7.

Xu, P., Zhang, G., Yang, B., Hsu, L.-T., 2024. Machine Learning in GNSS Multipath/NLOS Mitigation: Review and Benchmark. *IEEE Aerospace and Electronic Systems Magazine*, 39, 26–44.

Zhang, G., Hsu, L.-T., 2018. Intelligent GNSS/INS Integrated Navigation System for a Commercial UAV Flight Control System. *Aerospace Science and Technology*, 80, 368–380.

Zhang, J., He, T., Sra, S., Jadbabaie, A., 2020. Why gradient clipping accelerates training: A theoretical justification for adaptivity.

Zhou, Z., Liu, B., Yang, H., 2024. A Hopular-Based Weighting Scheme for Improving Kinematic GNSS Positioning in Deep Urban Canyon. *Measurement Science and Technology*, 35.