MATEC Web of Conferences **395**, 01012 (2024) https://doi.org/10.1051/matecconf/202439501012
ICPCM2023

WLAN monopole antenna design by Siamese convolutional neural network and KNN exploiting Gaussian process

Yubo Tian*, and *Fei* Meng

School of Information and Communication Engineering, Guangzhou Maritime University, Guangzhou, China

Keywords: Siamese neural networks, K-nearest neighbour, Gaussian processes, Monopole antennas, Modelling, Optimization.

Abstract. In the process of antenna design, surrogate models can generally be used, but modeling requires a large number of samples. Although full wave electromagnetic simulation software can handle this task, obtaining a large number of samples is time-consuming, however too small number of sample may lead to lower accuracy of the trained surrogate model. Inspired by semi-supervised learning methods, this paper uses Siamese convolutional neural networks (SCNN) and K-nearest neighbor (KNN) algorithms to generate highly reliable virtual samples and expand the training sample set, further improving the accuracy and robustness of the surrogate model by exploiting Gaussian process (GP) models. The proposed method is named SCNN-KNN-GP, which is used for the design of WLAN dual band monopole antennas. Moreover, the relationships between the performance of the proposed model and the increased number of virtual samples and the coefficient of the KNN are studied, resulting in a more excellent surrogate model structure.

1 Introduction

Artificial neural networks (ANNs) are mathematical models that mimic the behavioral characteristics of animal neural networks and perform distributed parallel information processing. This type of network relies on the complexity of the system and adjusts the interconnected relationships between a large number of internal nodes to achieve the purpose of processing information, with self-learning and adaptive capabilities. As an important branch of neural networks, the Siamese neural network (SNN) is a coupled architecture based on two ANNs [1-2]. The twin neural network takes two samples as inputs and outputs their representations embedded in high-dimensional space to compare the similarity between the two samples. SNN can perform small sample or one-shot learning and are not easily disturbed by erroneous samples. Therefore, they can be used for problems with strict fault tolerance

* Corresponding author: tianyubo@gzmtu.edu.cn

requirements, such as user identity linkage, change detection, image classification, target tracking, etc.

To extend the user identity linkage (UIL) problem to a general scenario, reference [3] propose a Siamese neural network (NN) architecture-based UIL (SAUIL) model that learns and compares the highest-level feature representation of input web-browsing behaviors with deep NNs, in order to link the web-browsing behaviors of users, which can help to distinguish specific users from others, such as children or malicious users. Reference [4] propose a deep siamese convolutional multiple-layers recurrent neural network (RNN) (SiamCRNN) for change detection (CD) in multitemporal very-high-resolution (VHR) images, and the experimental results in two homogeneous data sets and one challenging heterogeneous VHR images data set demonstrate that the promising performances of the proposed network outperform several state-of-the-art approaches. In order to solve the problems of unbalanced sample data and the lack of consideration of temporal information in existing Siamese-based trackers, reference [5] proposes a Siamese recurrent neural network and region proposal network (Siamese R-RPN), and it has achieved state-of-the-art performance on three large tracking benchmarks—OTB 2015, VOT2016 and VOT 2018—where this verifies its effectiveness. Reference [6] propose a dual-task constrained deep Siamese convolutional network (DTCDSCN) model to tackle the problem of the extracted features not being discriminative enough resulting in incomplete regions and irregular boundaries for building change detection. Reference [7] propose a Local Semantic Siamese (LSSiam) network to extract more robust features for solving drift problems caused by partial occlusion or non-rigid appearance deformation, and the proposed tracker can run at a high-speed of 100 Frame-per-Second (FPS) far beyond real-time requirement.

In this paper, we uses Siamese convolutional neural networks (SCNN) and K-nearest neighbor (KNN) exploiting Gaussian process (GP) to model a WLAN dual frequency monopole antennas, and the simulation result shows the proposed method is with high accuracy and robustness.

2 The proposed method

The KNN algorithm is initially an effective classification algorithm based on similarity values, which required finding K nearby points in the training data whenever a new data needs to be classified. Furthermore, the KNN algorithm can be extended to the field of predicting continuous variables by representing the label of the variable by the average of K independent variables near the predicted point. The SNN is a specific application of metric learning, with having the same two network structure and sharing the weight parameters. The network can learn the similarity between two samples and use different distance metric functions for measurement. The GP is a Bayesian nonparametric regression technique, and it is defined as a set of random variables in which any finite subset is subject to a joint Gaussian distribution, determined by the mean function and the covariance function.

Inspired by semi-supervised learning methods, SCNN and KNN algorithms are used to jointly generate samples similar to real data, and the accuracy of GP model training is improved by expanding the training set. The proposed SCNN-KNN-GP model aims to map the return loss curve of electromagnetic devices in high-dimensional space to their physical parameters in low-dimensional space, while maintaining a certain degree of feature extraction accuracy. The specific operation process is as follows.

On the basis of the original set S_{true} of true return loss curves, a new set S^* of return loss curves is randomly generated, then combining a mixed sample set $\left\{S_{true}, S^*\right\}$. The elements in the set are randomly combined in pairs to form N group of sample pairs, and the Euclidean distance between the sample pairs is calculated and sorted in ascending order. The first N_1 group of sample pairs is selected, and their label value is defined as 1, which is a similar sample pair. The remaining sample pairs are defined as their label value is 0, which is a non similar sample pair.

Select appropriate SCNN structures and parameters, and train the SCNN using similar and dissimilar sample pairs.

Use the trained SCNN to determine whether the new generated return loss curve is similar to the original one S_{true}. If yes, add it to form a new similar set $\{SimilarSet\}$, otherwise, delete it. Repeat the process until the number of samples in the similar set $\{SimilarSet\}$ is N_2.

Select an appropriate K in the KNN algorithm.

On the basis of the original training sample set, the KNN algorithm is used to calculate the electromagnetic physical parameters in similar sets $\{SimilarSet\}$, and the elements in the set $\{SimilarSet\}$ are combined with their corresponding electromagnetic physical parameter values to form an additional sample set $\{AddedSet\}$.

Train the GP model using the N_3 samples in the $\{AddedSet\}$ including the original training samples and added samples, and calculate the testing error of the GP model at the end of the training.

3 Design of dual-band monopole antenna

3.1 The antenna model and the dataset preparation

As shown in Fig. 1, the WLAN Dual Band Monopole Antenna (WLAN-DBMA) has two L-shaped branch arms, which the right one is for low-frequency with 2.4 GHz, and the left one is for high-frequency with 5.2 GHz. For the HFSS simulation, the frequency range is set to 1GHz~6GHz with the step size 0.001GHz, therefore a return loss curve with 701 sampling frequency points will be generated. For the antenna, some of the physical parameters have been fixed, including Lg =11, Wf =3.5, L =40, W =40, $L1$ =14, $W3$ =2. Other four physical parameters $[L2,L3,W1,W2]$ have a significant impact on the resonant frequency band of the designed antenna, and they are selected to be the output of the proposed model. The purpose of the experiment is to use the proposed SCNN-KNN-GP model to extract and compress the return loss curve of the WLAN-DBMA with 701 dimensions into the physical parameters in the 4-dimensional space.

Variable parameters values of the WLAN-DBMA are listed in Table 1, and 81 samples of different combinations are generated, in which randomly selected 72 samples are computed in the HFSS-MATLAB-API joint script to obtain their corresponding return loss. Among them, 48 samples $\{S_{true}\}$ are randomly selected as the training data, whereas other 24 samples are the testing data. On the basis of the original samples $\{S_{true}\}$, a large number of new sets $\{S^*\}$ of return loss are randomly generated to obtain a mixed sample set $\{S_{true}, S^*\}$. After pairing the elements in the set $\{S_{true}, S^*\}$, 720 sets of sample pairs are randomly selected, and the corresponding Euclidean distance is calculated and sorted in ascending order. The $N_1\%$ previous sample pairs are defined as similar sample pairs, meaning that the two curves have a high degree of similarity, and their label value is defined as 1. The remaining sample pairs are defined as non-similar sample pairs, meaning that the similarity between the two curves is relatively low, and their label value is defined as 0. The specific percentages are set to 10%, 20%, 30%, 40%, 50%, and 60% during the experiment.

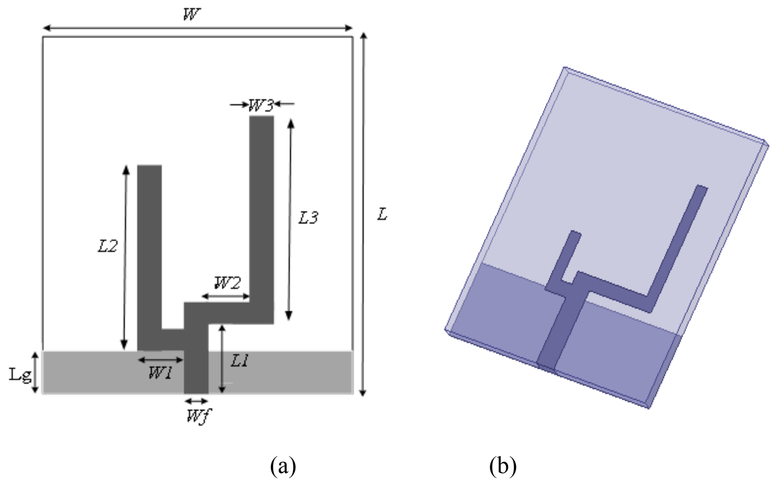

Fig. 1. The WLAN dual-band monopole antenna.

Table 1. Variable parameters of the WLAN-DBMA 111.

Parameters	Values (mm)
L2	8,9,10
L3	18,19,20
W1	4,4.5,5
W2	10,10.5,11

In the SCNN, the convolution layer 1 is with the size of kernels 6*6 and the number 60, the pooling layer 1 is with the size of window 2. The convolution layer 2 is with the size of kernels 4*4 and the number 80. In the fully connected layer, the number of nodes is 100. The SCNN is trained using the 720 sample pairs mentioned above. After the training, it is used to determine whether the randomly generated return loss curve meets the similarity range of the original real samples $\{S_{true}\}$. If satisfied, it is added to the set $\{SimilarSet\}$ until there are 48 elements in the set. Using the KNN algorithm and the original training samples calculates the physical parameters of these 48 samples, and then form a set $\{AddedSet\}$. The original training data and the set $\{AddedSet\}$ are used to train the GP model, respectively, and calculate the testing error of the two trained GP models. Among them, the coefficients in the KNN algorithm are set to 4, 6, 8, 10, and 12, respectively, to explore its impact on the performance of the proposed SNN-KNN-GP model. Without the sample set $\{AddedSet\}$, the traditional GP model predicts mean absolute percent error (MAPE) of 3.223%.

3.2 The experiment results

The impact of changing the coefficient K and increasing the number of samples on the performance of the proposed model is explored. Under the same coefficient K, the number of generated samples is multiplied to 12, 24, 36, 48. Originally, we have 48 training samples and 24 test samples. The prediction results in Table 2 show that when K=6, the accuracy of the model prediction has been improving from 12 to 48 samples. For K=4, 10, 12, when increasing the samples from 12 to 36, the accuracy of the model prediction has been improving. However, when increasing to 48, the accuracy has decreased, but it is within an acceptable range. When K=8, the model has the highest prediction accuracy with adding 12 samples, then it fluctuates with the change of added samples. However, 85% of the predicted MAPE results in Table 2 are better than those of the GP model directly trained without using additional samples. This shows that increasing the samples has a certain degree of credibility, and the proposed SNN-KNN-GP model has improved compared to the original GP model. Among them, when K=10 and increasing 36 samples, the trained model has the best effect, marked with red font in Table 2, with MAPE of 2.624%, which is 18.38% higher than the original method.

Table 2. Testing results when changing added sample number under different coefficient K.

10% similar sample pairs	Number of added samples	Testing result with MAPE
K=4	12	3.213%
	24	3.260%
	36	2.964%
	48	3.380%
K=6	12	3.284%
	24	3.019%
	36	3.100%
	48	2.998%
K=8	12	2.764%
	24	3.022%
	36	2.998%
	48	3.195%
K=10	12	3.003%
	24	2.885%
	36	2.624%
	48	2.917%
k=12	12	2.980%
	24	2.881%
	36	2.698%
	48	3.158%

Following, the impact of changing the percentage of defined similar samples and increasing the number of samples on the performance of the proposed model is explored. With fixed K=10, the percentage of similar samples is changed to 10%, 20%, 30%, 40%, 50%, and 60%. The model prediction results in Table 3 show that when the number of samples is increased to 36, the accuracy of the model is still relatively highest under all conditions. Among them, it is found that when the defined similarity percentage is 10%, the model has the best effect, marked with red font in Table 3.

In order to further investigate the relationship between the percentage of similar sample pairs and the predictive performance of the trained SNN-KNN-GP model, the Euclidean distance relationship between the probability density function of the trained SCNN for determining the normal distribution of similar and non-similar sample pairs is studied. We find when the first 10% of 720 sample pairs are defined as similar, the SCNN is more likely to judge the sample pairs as similar when the Euclidean distance of the sample pairs is less than 0.3. As the ratio of defined percentages increasing, the ability to determine whether the samples are similar gradually weakens. When $N_1\% = 60\%$, the accuracy of the trained model in determining whether the samples are similar or not is far inferior.

Table 3. Testing results with different percentages under the same coefficient K=10.

Percentage of similar samples	Number of added samples	Testing result with MAPE
10%	12	3.003%
	24	2.885%
	36	2.624%
	48	2.917%
20%	12	3.047%
	24	3.022%
	36	2.769%
	48	3.145%
30%	12	3.265%
	24	2.890%
	36	2.824%
	48	3.122%
40%	12	3.127%
	24	3.085%
	36	2.836%
	48	3.380%
50%	12	3.254%
	24	3.164%
	36	2.952%
	48	3.154%
60%	12	3.588%
	24	3.003%
	36	2.885%
	48	2.624%

In summary, when the percentage of similar samples is 10%, the nearest neighbor coefficient K is 10, and the number of virtual samples 36, the proposed SNN-KNN-GP model has the highest prediction accuracy.

4 Conclusion

The synthesis is a challenge topic in the electromagnetics domain, and finding an effective design method is particularly crucial. This article is dedicated to studying the inverse surrogate model method for antennas design, and proposes a SCNN-KNN-GP model, which generate high reliability samples through SCNN and KNN algorithms. Moreover, we study the influence of coefficient K in KNN algorithm on the proposed model, and suitable similarity percentage ratio and the tolerance value for adding reliability samples are also obtained, which reflects the mapping relationship between antenna return loss and physical parameters well. For the proposed SCNN-KNN-GP model, the next step will be to investigate the impact of SCNN network structure on the proposed model performance.

This work was supported by the Guangdong Basic and Applied Basic Research Foundation under Grant No. 2023A1515011272, the special project in key fields of Guangdong Universities of China under No. 2022ZDZX1020, and the Tertiary Education Scientific research project of Guangzhou Municipal Education Bureau of China under No. 202234598.

Reference

1. Bromley J, Guyon I, LeCun Y, Säckinger E and Shah R 1994 Signature verification using a" siamese" time delay neural network *In Advances in neural information processing systems* pp 737-744
2. Chopra S, Hadsell R, LeCun Y 2005 Learning a similarity metric discriminatively with application to face verification *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* 25 July 2005
3. Yuanyuan Qiao, Yuewei Wu, Fan Duo, Wenhui Lin, Jie Yang 2020 Siamese Neural Networks for User Identity Linkage Through Web Browsing *IEEE Transactions on Neural Networks and Learning Systems* (Volume: 31, Issue: 8) pp 2741-2751
4. Hongruixuan Chen, Chen Wu, Bo Du, Liangpei Zhang, Le Wang 2020 Change Detection in Multisource VHR Images via Deep Siamese Convolutional Multiple-Layers Recurrent Neural Network *IEEE Transactions on Geoscience and Remote Sensing* (Volume: 58, Issue: 4) pp 2848 - 2864
5. Xuechen Zhao, Yaoming Liu, Guang Han 2021 Cooperative Use of Recurrent Neural Network and Siamese Region Proposal Network for Robust Visual Tracking *IEEE Access* (Volume: 9) pp 57704 - 57715
6. Yi Liu, Chao Pang, Zongqian Zhan, Xiaomeng Zhang, Xue Yang 2021 Building Change Detection for Remote Sensing Images Using a Dual-Task Constrained Deep Siamese Convolutional Network Model *IEEE Geoscience and Remote Sensing Letters* (Volume: 18, Issue: 5) pp 811 - 815
7. Zhiyuan Liang, Jianbing Shen 2019 Local Semantic Siamese Networks for Fast Tracking *IEEE Transactions on Image Processing* (Volume: 29) pp 3351-3364