remote sensing

MDPI

Technical Note

Predicting the Spatial Distribution of VLF Transmitter Signals Using Transfer Learning Models

Hanqing Shi [1], Wei Xu [1,2,*], Binbin Ni [1,2], Xudong Gu [1,2], Shiwei Wang [1], Jingyuan Feng [1], Wen Cheng [1], Wenchen Ma [1], Haotian Xu [1], Yudi Pan [1] and Dongfang Zhai [1]

[1] Department of Space Physics, School of Electronic Information, Wuhan University, Wuhan 430072, China; shq1108@whu.edu.cn (H.S.); bbni@whu.edu.cn (B.N.); guxudong@whu.edu.cn (X.G.); wangsw@whu.edu.cn (S.W.); jingyuanf@whu.edu.cn (J.F.); wencheng@whu.edu.cn (W.C.); mawenchen@whu.edu.cn (W.M.); xuht18@whu.edu.cn (H.X.); panyudi@whu.edu.cn (Y.P.); dongfangz@whu.edu.cn (D.Z.)

[2] Hubei Luojia Laboratory, Wuhan 430072, China

* Correspondence: wei.xu@whu.edu.cn

Abstract: The D-region ionosphere (60–100 km altitude) is critical for radio communication and space weather research but cannot be easily measured because it is too low for satellites and too high for balloons. The most effective technique is to remotely sense by measuring Very-Low-Frequency (VLF, 3–30 kHz) waves emitted from man-made transmitters, a technique that was traditionally utilized to estimate the average ionospheric condition between the transmitter and receiver. Recently, various methods have been proposed to remotely sense the D-region ionosphere in large areas using network observation of VLF transmitter signals. The key component of these methods is the VLF propagation model, and the Long-Wavelength Propagation Capability (LWPC) model is employed in most cases due to its relatively fast computation speed. However, it is still too long and thus insufficient for real-time remote sensing. To overcome this limitation, we have proposed a neural network model to replace the LWPC model and to shorten the computation time of VLF propagation. This model is specifically obtained using the transfer learning method by retraining the last three layers of the well-established VGG16, GoogLeNet, and ResNet architectures. We have tested different methods to organize the input data for these neural network models and verified their performance using the validation dataset and real measurements. Among the three models, GoogLeNet outperforms the other two, and the root mean squared error (RMSE), with respect to LWPC results, is as low as 0.334. Moreover, the proposed neural network model can dramatically reduce the computation time. The computation time to calculate the signal distribution near the transmitter is 1184 s if one uses the LWPC model but 0.87 s if the present neural network model is used. The performance of this model is also excellent for ionospheric conditions that are not included in the validation dataset. Therefore, this model is robust and can be used to remotely sense, in real time, the D-region ionosphere in large areas, as well as various scientific and engineering needs.

Keywords: VLF electromagnetic wave; regional longwave propagation; transfer learning; convolutional neural network

check for updates

Received: 13 January 2025
Revised: 21 February 2025
Accepted: 27 February 2025
Published: 28 February 2025

Citation: Shi, H.; Xu, W.; Ni, B.; Gu, X.; Wang, S.; Feng, J.; Cheng, W.; Ma, W.; Xu, H.; Pan, Y.; et al. Predicting the Spatial Distribution of VLF Transmitter Signals Using Transfer Learning Models. *Remote Sens.* **2025**, *17*, 871. https://doi.org/10.3390/rs17050871

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The D-region ionosphere, located at altitudes between 60 and 100 km, is a partially ionized layer, with the electron density being typically between 1 and 1000 cm^{-3} under

daytime conditions [1]. This region is constantly influenced by various eruptive events with solar and galactic origins, particularly solar flares, solar eclipses, particle precipitation from the radiation belts, and gamma-ray bursts [2–5]. The D-region ionosphere is critical for wireless communication, navigation, and positioning [6–8]. Due to the low electron density in this region, direct measurements are extremely challenging. High-power incoherent scatter radars and riometers can detect the electron density of the D-region ionosphere, but their spatial resolution is limited, and the distance is short [9]. Although sounding rockets can provide direct measurements and accurate information about electron density, they are restricted to the limited temporal and spatial coverage and the operational cost is high [10]. One of most effective methods to remotely sense the D-region ionosphere is via measurements of VLF waves emitted by man-made transmitters [11]. These waves are well trapped within the Earth–ionosphere waveguide (EIWG) and can propagate for long distances with low attenuation (~2–3 dB/Mm) [12]. The VLF signals measured at a given location are mostly controlled by the electron density of the D-region ionosphere along the propagation path, as well as ground conductivity. Therefore, VLF measurements have been widely used to remotely sense the D-region ionosphere, with the aid of VLF propagation models. Lehtinen and Inan have developed a full-wave model [13], which has been used to estimate the attenuation of VLF transmitter signals while propagating through the Earth's atmosphere. The modeling results turn out to be highly consistent with satellite measurements [14]. On the other hand, Marshall et al. [15] have developed a Finite Difference Time Domain (FDTD) model to simulate the propagation of VLF pulses emitted by lightning discharge [16]. The most widely used propagation model is the LWPC, which is a mode solver developed by the Space and Naval Warfare Systems Center (now the Naval Information Warfare Center Pacific) [17]. With LWPC, VLF measurements have been utilized to estimate the average conditions of the D-region ionosphere along the path between the transmitter and the receiver [18]. More recently, various methods have been proposed to infer the D-region ionosphere in large areas using network observation and VLF propagation models [19]. In these methods, the computation time of the VLF propagation model needs to be sufficiently short in order to fulfill the need of D-region remote sensing in real time. Although the computation time of LWPC is significantly shorter compared to other propagation models, it is still too long for real-time remote sensing of the D-region ionosphere.

With the advent of Graphics Processing Units (GPUs), the deep learning method has exhibited remarkable capabilities in uncovering underlying patterns and automatically extracting features with extremely high accuracy and rapid computation [20]. Given their proficiency in handling nonlinear problems, deep learning-based methods have been employed in VLF-related work, which typically requires substantial computation time on Central Processing Units (CPUs). Xu et al. [21] utilized the YOLOv5 to automatically detect the VLF tweek signals. Gross and Cohen [22] developed the Artificial Neural Network (ANN) for remote sensing the D-region ionosphere. Liu et al. [23] employed the Elman Neural Network (ENN) to inverse the electron density in the lower ionosphere. Wang et al. [24] and Xiao et al. [25] harnessed the ability of Convolutional Neural Networks (CNNs) to classify the VLF/LF lightning signals and radiation electric field waveforms, respectively. They have all achieved outstanding results in their respective research.

However, the traditional deep learning framework functions as a black-box. Large sets of labeled data are needed to train such a model [26], and these are frequently unavailable in various research topics, including studies related to the D-region ionosphere and VLF propagation. This limitation largely hampers the usage of deep learning methods in these research topics. Nonetheless, transfer learning models can mitigate this issue, since they can be trained with a limited number of samples, a process that, together with physics-informed

neural networks, has been successfully utilized to predict nuclear reactor transients [27], as well as material properties.

The goal of this study is to shorten the computation time of the LWPC model using transfer learning models, thereby fulfilling the need to remote sense the D-region ionosphere in real time. Previous studies mostly treat the VLF data as-is when training the prediction model, while, in this study, we use the spatial distribution of VLF amplitude and treat it as an image in order to train the CNN model. In this regard, the models that are well trained in the community of image processing have been employed via the transfer learning method.

The ANN model developed by Gross was developed mostly under the assumption that the ionospheric parameters (h' and β) along the transmitter–receiver propagation path is homogeneous. This assumption is oversimplified and largely different from the real scenario. In contrast, in our method, each transmitter–receiver path has been divided into 200 segments, with the length of each segment being 40 km. Our assumption is more realistic, while also more challenging for the training of deep learning models. Specifically, we employ three well-established, pre-trained models—VGG16, GoogLeNet, and ResNet—by replacing and retraining the last three layers of these models, including the fully connected layer, the SoftMax layer, and the output classification layer. Moreover, we compare the performance of these models under different ionospheric conditions to assess their robustness and applicability to large-area D-region ionosphere estimations.

2. Models and Data Set

2.1. LWPC

The LWPC model is a theoretical framework designed to estimate the amplitude and phase of long-wave signals as they propagate within the Earth–ionosphere waveguide, with particular emphasis on low-frequency (LF) and VLF electromagnetic waves. This model can calculate the attenuation, reflection, and refraction at different boundaries, and the ionospheric electron density, geomagnetic field, and terrain have been taken into account; thus, the model is widely used in various research and engineering studies. Its ability to simulate complex wave interactions in diverse environments makes it an invaluable tool for optimizing communication networks, investigating natural electromagnetic phenomena, and improving signal reliability under challenging conditions.

The primary inputs to the LWPC model are the electron density profile along the VLF propagation path, the geomagnetic field, and the ground conductivity and dielectric constants. Default lookup tables are set as default for the ground conductivity and dielectric parameters, while the geomagnetic field is obtained from the World Magnetic Model (WMM) [28]. The transmitting power and direction of different transmitters are well defined in LWPC. The parameter that mostly controls the amplitude and phase of VLF signals near the transmitter is the D-region electron density, which is parameterized using the reflection height h' and the sharpness parameter β, as defined in the Wait and Spies formula [29]:

$$N_e(z) = 1.43 \times 10^{13} \exp(-0.15z) \exp(\beta(z - h\prime)) \quad (1)$$

where z is the height, h' is the reflection height, and β is the sharpness parameter.

2.2. Data Preparation

In this work, we mainly focus on the NWC transmitter, which is located at the northwestern corner of Australia and has a transmitting frequency of 19.8 kHz. The key parameters that govern the spatial distribution of transmitter signals in LWPC simulations are h' and β in the nearby region. To calculate the signal distribution, we use NWC as the central point. With the north direction corresponding to the azimuth angle of 0°, the

area surrounding NWC can be divided into 360 azimuthal directions. For each azimuth angle, we calculate the signal attenuation along a propagation path of 8000 km from the NWC, which is divided into 200 segments, with the length of each segment being 40 km, when setting the input data. We can use 200 pairs of h' and β to describe the ionospheric condition for one transmitter–receiver path, assuming the D-region electron density within each segment is uniform. For the output data, the length of each segment is chosen to be 20 km in order to provide higher-resolution spatial distribution of transmitter signals, which gives 400 output data computed by LWPC. An example of this path is shown in Figure 1.

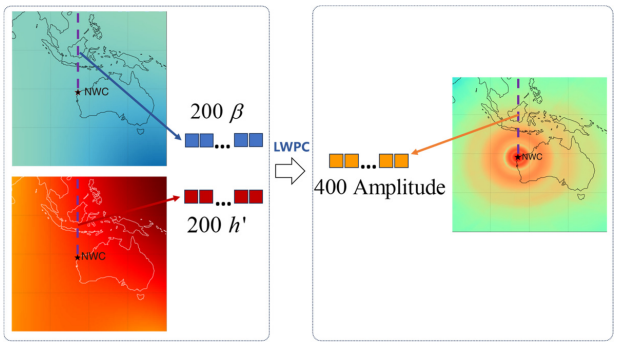


Figure 1. The h and β values used for the calculation of VLF amplitude along a single propagation path, corresponding to the azimuth angle of 0° near NWC. The figures on the left refer to h' and β in the area surrounding NWC, and the one on the right refers to the amplitude distribution.

Each path, denotated as path n, is characterized with 200 pairs of h' and β parameters: $X_{path_n} = [h'_1, h'_2, \ldots h'_{200}, \beta_1, \beta_2, \ldots \beta_{200}]_n^T \in \mathbb{R}$, corresponding to $h'_n = [h'_1, h'_2, \ldots h'_{200}]_n^T \in \mathbb{R}$ and $\beta_n = [\beta_1, \beta_2, \ldots \beta_{200}]_n^T \in \mathbb{R}$. The corresponding amplitude vector $A_n = [A_1, A_2, \ldots A_{400}]_n^T \in \mathbb{R}$ can be derived from each path, given that the output resolution is 20 km. The region, notated as region m, encompasses the 360 azimuthal directions and can be described by a two-dimensional matrix $X_{region_m} = [X_{path_1}, X_{path_2}, \ldots X_{path_360}]_m^T \in \mathbb{R}$ of waveguide parameters, while the associated output amplitude matrix is denoted as $Y_{region_m} = [A_1, A_2 \ldots A_{360}]_n^T \in \mathbb{R}$.

The data used to train the deep learning model are obtained using the LWPC code. Specifically, a total of 4000 spatial distributions have been utilized, namely 4000 images that are similar to Figure 1. It takes approximately 400 h to calculate the training dataset. The electron density of the D-region ionosphere near the NWC transmitter is calculated using the Ferguson model. A total of 2000 ionospheric conditions have been calculated using the Ferguson formula [30].

$$h'_{Ferguson} = 74.37 - 8.097\cos\theta - 1.213\cos\left(\frac{2\pi(m-0.5)}{12}\right) \quad (2)$$

$$\beta_{Ferguson} = 0.3849 - 0.08584\cos\left(\frac{2\pi(m-0.5)}{12}\right) + 0.1296\cos\left(\frac{2\pi(m-0.5)}{12}\right) - 0.1658\cos\chi + 0.15 \quad (3)$$

where χ is the solar zenith angle, θ is the geographical latitude, and m is the month number.

To enhance the model's applicability, we have also considered a Gaussian distribution for the D-region ionosphere near NWC. In this approach, 2000 random values between 65 and 75 km are generated for h', and 2000 random values between 0.25 and 0.45 km^{-1} are generated for β. These values are utilized as mean values (μ) in a two-dimensional Gaussian function. Subsequently, these matrices are then paired by selecting one matrix from the h' group and one from the β group, forming unique input pairs for a computational program. This process yielded 2000 input combinations [31], and the two-dimensional Gaussian

function is shown below. To simplify the calculation, we set the ρ to 0, σ_1 equals σ_2 to 1, and μ_1 equals μ_2 equals μ mentioned before.

$$f_{Gauss}(x,y) = \frac{1}{(2\pi\sigma_1\sigma_2\sqrt{1-\rho^2})}\exp[-\frac{1}{2(1-\rho^2)}(\frac{(x-\mu_1)^2}{\sigma_1^2} - \frac{2\rho(x-\mu_1)(y-\mu_2)}{\sigma_1\sigma_2} + \frac{(y-\mu_2)^2}{\sigma_2^2})] \quad (4)$$

where x, y are the random variables, μ_1, μ_2 are the mean values, σ_1, σ_2 are the standard deviations, and ρ is the correlation coefficient between x and y.

The Ferguson model, by integrating the physical properties of the ionosphere with empirical observational data, provides a more precise characterization of the reflection and propagation behaviors of VLF signals in the ionosphere. In contrast to traditional uniform ionospheric models, the Ferguson model more effectively accounts for the spatial inhomogeneity and temporal variability inherent in the ionosphere. Furthermore, the use of a Gaussian distribution to model the ionospheric state is aimed at enhancing the network's predictive performance under stochastic ionospheric conditions. The Gaussian distribution also exhibits inherent correlations and regularities within the ionosphere, thereby facilitating the training process and improving the convergence of deep learning networks.

2.3. Input and Output Image Data Set for Deep Learning

The ionospheric parameters (h' and β) are directly used as the input of the neural network model and don't need to be renormalized. In this study, three methods were used to process the input data to be used in the deep learning network. These methods were designed for different spatial and dimensional representations. The first method specifically used data from a single propagation path, namely a 1×400 one-dimensional matrix. The number 400 corresponds to the 200 pairs of h' and β used for each segment. This matrix was then reshaped into a 20×20 two-dimensional matrix to fulfill the input requirements of neural network (see Figure 2). The second method utilized data from the 360 azimuth angles, namely a 360×400 two-dimensional matrix. This approach preserved the full set of propagation paths along different azimuth angles. The third method mapped the data from the 360 paths to their corresponding geographic latitude and longitude coordinates, namely the h' and β at different latitudes and longitudes near the transmitter. This map was subsequently converted into an image, which was directly used as an input for the neural network.

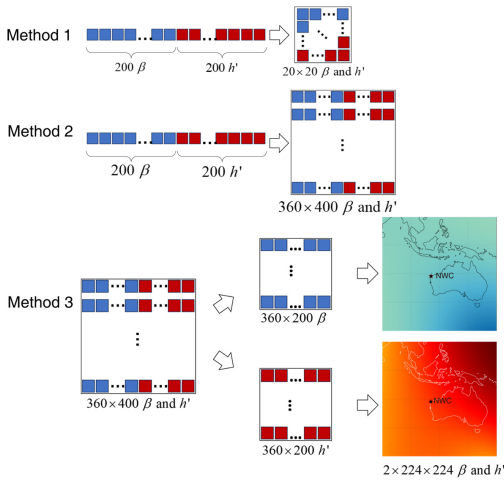


Figure 2. Three methods to convert the ionospheric parameters near the transmitter into the input of neural network models. Input data in method 1 is the 1 × 400 matrix from one propagation path, which is reshaped into 20 × 20. The one in method 2 is the 360 × 400 matrix from 360 propagation paths. The input in method 3 is the figure mapping the data from 360 propagation paths to the corresponding geographic latitude and longitude coordinates.

By employing these methods, the neural network model can analyze the data from different perspectives, allowing it to extract relevant features from spatial, structural, and visual dimensions. This multifaceted approach enhanced the network's ability to learn complex patterns and improved its performance across diverse input formats [32].

3. Method

3.1. Convolutional Neural Network

The CNN typically consists of convolutional layers, pooling layers, and a fully connected layer [33]. The primary goal of the convolutional layer is to extract features from the input image using kernels or filters, allowing the network to learn spatial hierarchies of patterns. Pooling layers can resolve the overfitting issue by down-sampling the data, effectively compressing the feature maps while retaining important information [34]. Finally, the fully connected layer connects the nodes from the previous layers to the output dataset, enabling the network to provide prediction results based on the features learned throughout the architecture [35]. This structure facilitates effective image processing and prediction tasks in various applications. The brief structure of CNN is shown in Figure 3.

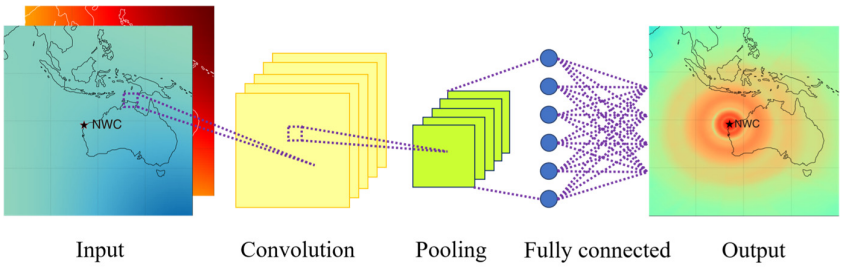


Figure 3. The brief structure of a convolutional neural network. The convolution layer is utilized to capture the feature of the data input. The pooling layer is employed to reduce the complexity of the feature. And the fully connected layer serves as the role to connect the output data to the network.

In this work, three of the most commonly used deep learning models have been tested, including VGG16, GoogLeNet, and ResNet. VGG16 is a deep CNN that made significant advancements in image classification tasks by focusing on simplicity and depth, as proposed by Simonyan and Zisserman [36]. It is different from other models in that

small 3 × 3 filters are uniformly applied across all convolutional layers, allowing for deeper feature extraction while maintaining a manageable number of parameters. The small convolutional filters enable the network to capture fine-grained spatial hierarchies and patterns. Additionally, VGG16 employs max-pooling layers to progressively reduce the spatial dimensions of the feature maps, thereby enhancing feature invariance. Despite its relatively large number of parameters, VGG16 is easy to implement and has been highly effective in transfer learning due to its straightforward and modular design [37].

GoogLeNet won the championship at ISVRC-2014 [38] and has brought vital innovations to deep convolutional neural networks by introducing the Inception module. The primary feature of this module is its ability to simultaneously perform convolutions of varying sizes (1 × 1, 3 × 3 and 5 × 5) and pooling operations. This architecture reduces computational complexity compared to traditional deep networks by using 1 × 1 convolutions to increase the nonlinearity of the network and reduce the dimensionality of feature maps before performing more computationally expensive operations [39]. Although GoogLeNet consists of 22 layers, it is much more efficient in training parameters compared to previous models, thanks to its modular design and careful use of pooling layers. This efficiency allows GoogLeNet to achieve state-of-the-art performance while being less prone to overfitting, marking a major step forward in the design of deep neural networks [40].

ResNet was proposed by He et al. [41] to address the degradation problem in deep neural networks. The key innovation in ResNet is the introduction of residual connections, which bypass one or more layers by using shortcut connections to allow the model to learn residual functions with respect to the input, making it easier to optimize deeper networks. The network's architecture is built on stacking multiple residual blocks, each consisting of two or three convolutional layers, followed by batch normalization and a Rectified Linear Unit (ReLU). The primary advantage of ResNet is its ability to train extremely deep networks without the vanishing gradient problem [42].

In this work, we have utilized the TensorFlow library in Python to save the pre-trained model structures, as well as the training weights of VGG16, GoogLeNet, and ResNet.

3.2. Transfer Learning

Training a deep CNN model requires a large dataset to improve the model accuracy. As in this study, we needed to perform significant amount of LWPC simulations, which are extremely computationally expensive. However, the above-mentioned models were well trained using the ImageNet dataset, which contains more than 14 million images [43]. Transfer learning, which is designed for scenarios when large datasets of labeled images are not available, is thus utilized in this study. The core concept of transfer learning is to leverage a previously well-trained model and adapt it to fulfill new goals using a smaller training dataset [44]. One of the key advantages of transfer learning is feature reuse, since the lower layers of deep neural networks typically learn general features (e.g., edges, textures) that are common in different tasks [45]. A major innovation in transfer learning is the ability to only fine-tune the latter layers of a pre-trained model, while keeping the earlier layers frozen. This approach significantly reduces training time and mitigates the risk of overfitting when working with smaller datasets [46].

In this work, we utilized the pre-trained weights for the initial layers of VGG16, GoogLeNet, and ResNet, which were imported from the TensorFlow library in Python. The latter layers of the networks were modified based on the LWPC dataset. For example, using Method 2, the input data—organized as a four-dimensional tensor—was directly fed into the frozen convolutional and pooling layers. The last three layers of the network, including the fully connected layer, the linear activation layer, and the output layer, were replaced to provide the output format required for the LWPC dataset. Specifically, the

neural elements in the fully connected layer were adjusted to output the data relevant to the LWPC task, instead of the original 1000-class classification used in the pre-trained models. The flowchart of the transfer learning model are shown in Figure 4.

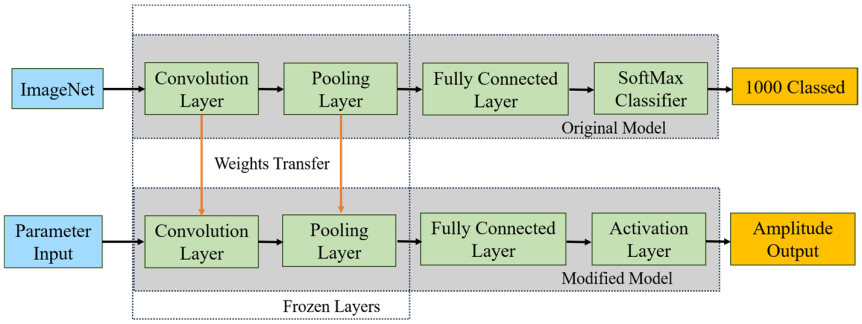


Figure 4. The flowchart of the transfer learning model. The weight of former layers, such as convolution layers, which are used to capture the feature are frozen. The last several layers are retrained to modify the pre-trained model.

3.3. Training Set

The training parameters of the present model are set as follows: the samples are randomly shuffled and divided into the training and validation set with a ratio of 8:2, which is commonly used in model training, since this ratio balances the training and validation of the model. The input batch is set to 32, which is large enough to provide stable gradient estimates during backpropagation, leading to more reliable and smoother model updates [46]. Moreover, this value is well within the memory constraints of modern GPUs, allowing for efficient parallel computation. Adam is chosen as the optimizer due to its ability to combine the strengths of two widely used optimization algorithms: AdaGrad and RMSProp. This optimizer can adaptively adjust the learning rate of each parameter while maintaining per-parameter learning rates based on the first moment (mean) and second moment (variance) of the gradient [47].

The learning rate (denoted as 'lr' hereafter) is set as a function of epochs (5), in order to improve model performance by adjusting how quickly the model learns over time [48]. The rationale behind this approach is that during the early stages of training, a relatively high learning rate helps the model explore the parameter space more broadly, allowing for faster convergence. As the model converges in later epochs, a smaller learning rate enables finer adjustments, preventing the model from overshooting minima in the loss function [49]. The epoch set is 100, which allows for sufficient training of the network. We use the Root Mean Squared Error (RMSE) as the training loss function since it assigns more weight to larger errors and penalizes significant deviations more than smaller ones, forcing the model to focus on reducing relatively large errors [50]. The training parameters set in shown in Table 1.

$$lr = \begin{cases} 0.001 & ,epoch < 10 \\ 0.001 \times \exp(-0.1 \times epoch), & epoch \geq 10 \end{cases} \quad (5)$$

where lr is the learning rate and $epoch$ is the training epochs.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(Y_t - Y_p)^2} \quad (6)$$

where n is the number of the training result, Y_t is the true value, and Y_p is the predicted outputs of the network model.

Table 1. Training Parameters Set.

Ratio	Batch Size	Optimizer	Epoch	Lr
8:2	32	Adam	100	f (epoch)

The software platforms used in this study were PyCharm Community Edition 2023.3.2 (JetBrains, CZE) and Python 3.9.19 with TensorFlow 2.10 for the machine learning algorithms and the neural network models. MATLAB R2021a is used for plotting figures.

The hardware in this study was an Intel® Core i7-14700KF CPU @ 3.40GHz, 64 GB memory, and a NVIDIA GeForce RTX 4070 Ti graphics card, 12 GB video memory.

4. Results and Discussion

4.1. Training Processing and Results

During the training process of deep learning, the loss typically decreases as the model gradually minimizes the discrepancy between its predictions and the target values. The initial loss may be high, and, as the training progresses, the optimization algorithm updates the model's parameters, improves the predictions, and the loss decreases. This decrease is largely nonlinear, with a rapid decrease followed by plateaus or fluctuations, particularly when the model encounters more complex patterns in the data. Monitoring the loss on both the training and validation (denoted as 'val loss' hereafter) sets is crucial. This helps identify the overfitting issue, namely the training loss continues to decrease while the validation loss starts to increase [51]. The loss curves obtained with the three methods and three networks are presented in Figure 5.

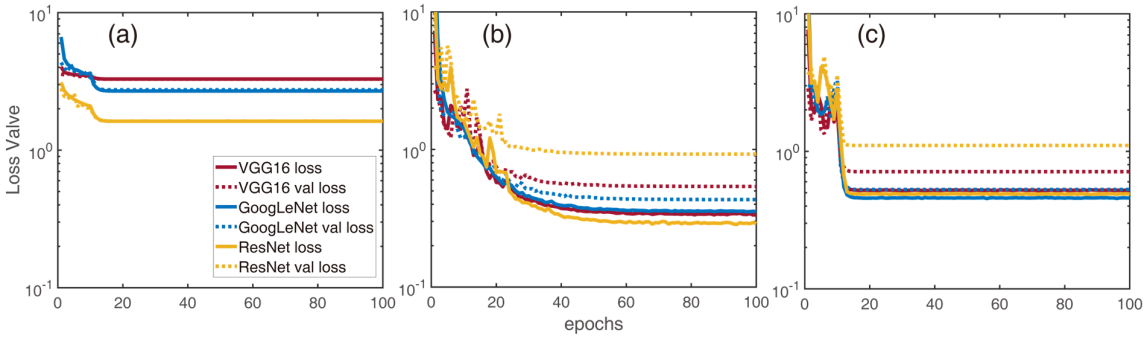


Figure 5. The training loss and validation loss of (**a**) input method 1; (**b**) input method 2; and (**c**) input method 3.

Figure 5a shows the loss for method 1 and different neural networks. For all three networks, the loss is above 1, indicating that the single path provides fewer features than the other methods, which hampers efficient fitting of the neural network. In contrast, as shown in Figure 5b, the loss and validation loss for method 2 exhibit a stable downward trend and converge after 40 epochs. The loss value for ResNet is the lowest among the three networks, while its validation loss approaches 1, indicating better performance on the training dataset compared to the validation dataset. This discrepancy implies potential overfitting, where the model captures specific patterns and noise in the training data, but not for new data, as evidenced by the higher validation loss. The performance of GoogLeNet is balanced between the training and validation dataset. Figure 5c shows the loss curves for the third input method. These curves sharply decrease during the first 18 epochs and remain at low values thereafter. This is attributed to input method 3 being more similar to the original training data of ImageNet, facilitating smoother adaptation of the neural network to the new dataset. However, the loss value after convergence is slightly higher than that of method 2.

RMSE is commonly used as a loss function in regression tasks due to its effectiveness in quantifying the difference between predicted and actual values [52]. RMSE has the same units as the target variable, and the model's performance can be better estimated [53]. In addition, RMSE is a smooth and differentiable function and a standard choice for evaluating model performance in regression tasks, such as gradient descent, allowing for effective computation of gradients [54]. Thus, we assess the model performance using RMSE. The average amplitude value is 63.86, and the RMSE values for the three methods are 0.354, 0.334, and 0.289.

The RMSE between different combinations of input methods and networks is summarized in Table 2. The RMSE for method 1 is, in general, larger than the other two methods, while the RMSE for the other two methods is satisfactory.

Figure 6 shows an example of the LWPC results used for estimating the model performance. Figure 6a,b show the h' and β values at different latitudes and longitudes. Figure 6c shows the LWPC results obtained using the h' and β values shown in Figure 6a,b. The average signal amplitude value is 63.86, and the amplitude near the transmitter exhibits typical ring structures, which are caused by model interference [55]. We use the results in Figure 6c to verify the model performance. Of note, it takes approximately 1184 s to calculate the results shown in Figure 6c.

Table 2. RMSE of the Three Methods with Three Modified Networks.

	VGG16	GoogLeNet	ResNet
Method 1	3.28	2.69	1.62
Method 2	0.354	0.334	0.289
Method 3	0.516	0.457	0.488

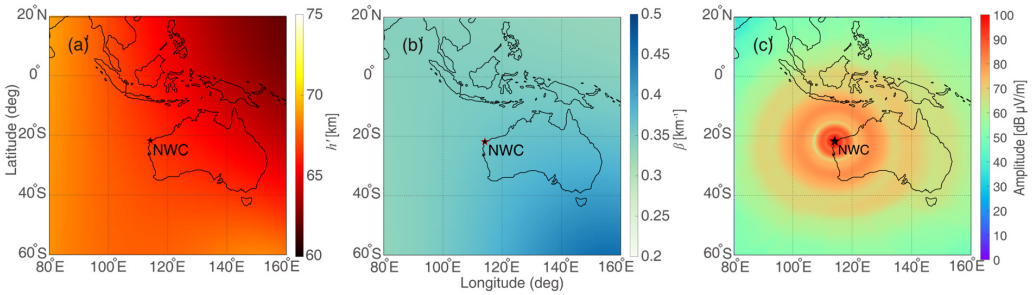


Figure 6. The (**a**) h' and (**b**) β values used for the calculation of signal amplitude near NWC. (**c**) The spatial distribution of signal amplitude near NWC, used for model verification.

Though ResNet has the lowest RMSE value (see Table 2), the large difference obtained between training loss and validation loss (see Figure 5b) implies the applicability of this network is limited. In this regard, we consider GoogLeNet as the best choice. The time used for the neural network to calculate the spatial distribution of transmitter signals (for example, Figure 7a,c,e) is only 0.87 s, while the computation time used by LWPC is 1184 s (Figure 6c). Moreover, considering the structural advantages, the unique inception module in GoogLeNet, which employs parallel convolutional layers with varying filter sizes and pooling operations within the same layer, enables the network to capture both fine and large-scale patterns efficiently, similar to the spatial distribution of transmitter signals. The multi-level feature extraction capacity allows the network to capture features at different scales and recognize complex patterns in VLF propagation images.

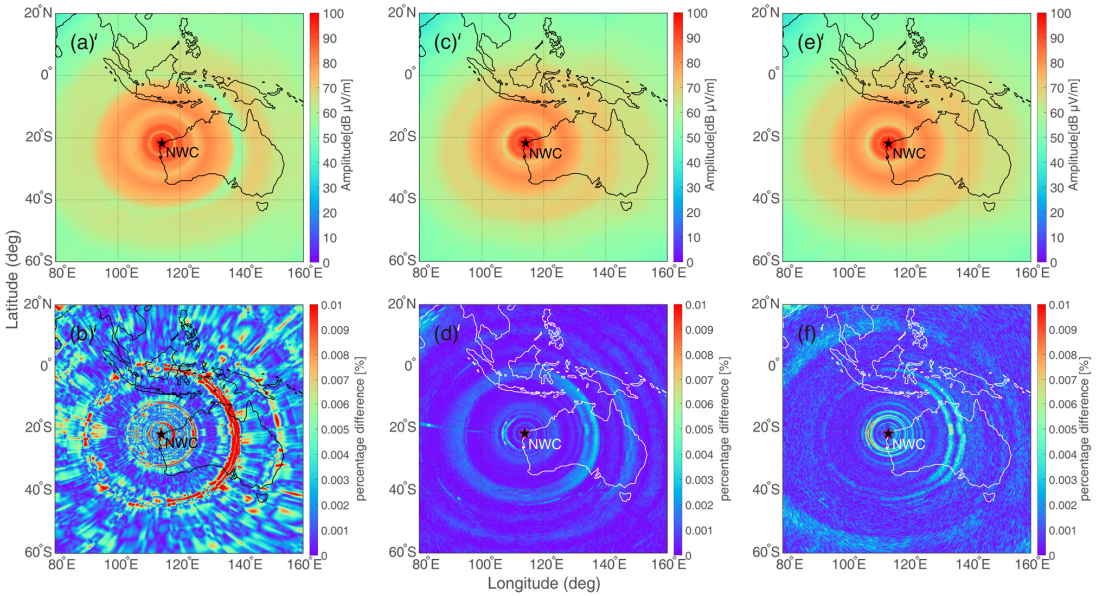


Figure 7. Upper row: the VLF amplitude obtained using (**a**) input method 1; (**c**) input method 2; and (**e**) input method 3. Bottom row: (**b**,**d**,**f**) the difference between the upper row and the results shown in Figure 6c.

Figure 7 shows the signal amplitude estimated using the three networks utilized in this work. The first row of Figure 7 shows the signal amplitude, while the second row shows the percentage difference between the results predicted using the three networks and the LWPC results shown in Figure 6c. The percentage error for method 1 is between 0.2% and 0.3% for most regions near the transmitter. In contrast, the percentage error obtained using methods 2 and 3 is much lower. Input method 2 outperforms method 3, since the percentage error is below 0.1% in most regions.

4.2. Verification Using Real VLF Measurements

To further verify the applicability of the neural networks, we have utilized the h' and β values obtained using real VLF measurements and an ionospheric inversion method [56]. Specifically, we use the real measurements of VLF signals in Suizhou, Wuhan, and Leshan, China, collected by VLF receivers independently developed by the Wuhan University [57], as shown in Figure 8a. The VLF signals from the NWC, JJI, and VTX transmitters are used to infer the h' and β values of the D-region ionosphere. The h' and β values obtained from this method are notably independent and different from the Ferguson or Gaussian ionospheres used for the training and validation datasets. Therefore, these values are used to examine the applicability/robustness of the present model.

The β and h' values, as inferred from real VLF measurements, are presented in Figure 8b,c. We have further estimated the signal amplitude near NWC using the LWPC code, and the results are shown in Figure 8c. The results obtained using input method 2 and GoogLeNet are shown in Figure 8d. The difference between the LWPC results (Figure 8c) and the results obtained using neural networks (Figure 8d) is shown in Figure 8e. It is clear that the results obtained using the neural network are reasonably close to those of LWPC. The average percentage difference is 0.0564%, while the last difference is 0.13%. The good agreement obtained between LWPC and the neural networks demonstrates that the proposed model is sufficient for estimating the spatial distribution of transmitter signals while being significantly faster than LWPC.

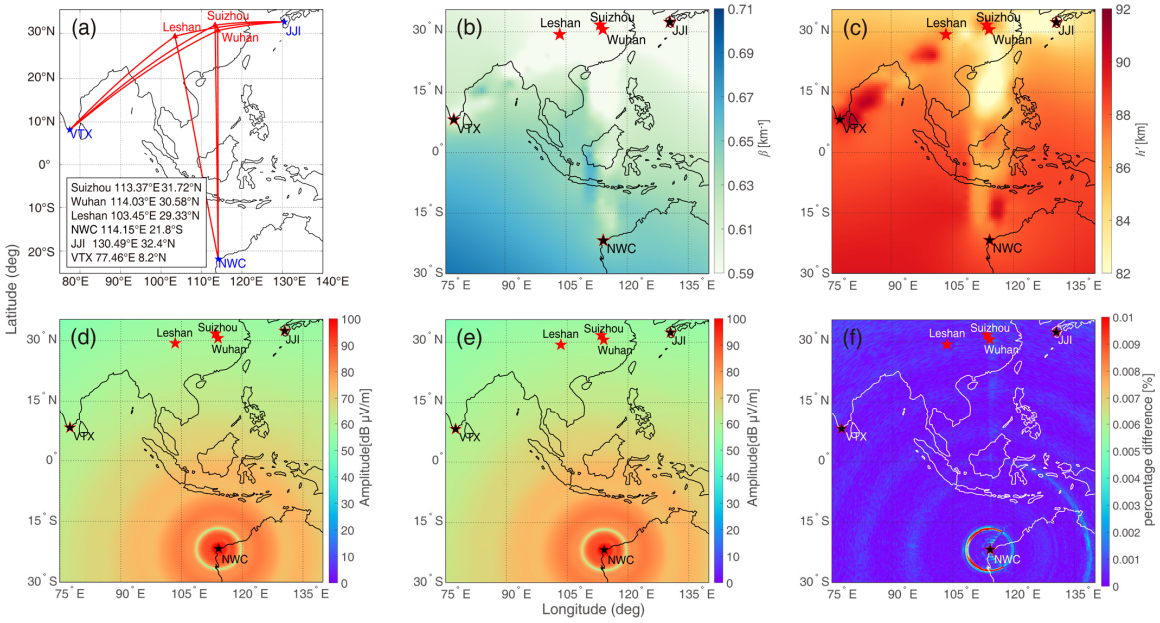


Figure 8. (**a**) The great circle paths between NWC, JJI, and VTX transmitters and the receivers in Suizhou, Wuhan, and Leshan, China. (**b**) β (**c**) h' derived from the VLF measurements in Suizhou, Wuhan, and Leshan. The signal amplitude near NWC calculated using (**d**) LWPC and (**e**) the GoogLeNet model (input method 2). (**f**) The difference between LWPC results and the results obtained using the GoogLeNet model.

5. Conclusions

The VLF technique represents one of the most reliable methods for remote sensing the D-region ionosphere. VLF propagation models are essential in this process, but their high computational expense highly limits their scientific and engineering usages. Previous studies employed deep learning networks to replace traditional VLF propagation models. However, most studies mainly focused on mapping the properties of the D-region ionosphere and the amplitude of VLF waves along single paths, and how to estimate the spatial distribution of transmitter signals has not been investigated. In this study, we have adopted deep learning techniques to train a neural network model, which can replace the LWPC model using a much shorter computation time, while calculating the spatial distribution of transmitter signals.

To train a deep learning model typically requires a large number of samples. To address this challenge, we have employed transfer learning by reusing the pre-trained VGG16, GoogLeNet, and ResNet models. By fine-tuning the last three layers of these well-trained models, one can largely reduce the training time while maintaining the accuracy. These models have undergone extensive refinement in different research areas, and their weight structures have been improved via training on vast datasets. Therefore, these models are highly adaptive and can be easily tuned for the present study.

When the form of the input data aligns with the original data, the transfer learning model converges relatively fast. Therefore, we employ three methods to convert the ionospheric parameters to match the input format of the neural network. These methods ensure compatibility with the neural network architecture and facilitate better feature extraction.

By examining the training loss and directly comparing the prediction results with LWPC results, we have found the best data input method and neural network model. Among different data input methods and neural networks, input method 2 with GoogLeNet exhibits the best performance with notably lower values for both loss and validation loss. The average predicted amplitude is 63.86 dB, with an RMSE of 0.334. Furthermore, the

percentage difference between the LWPC results and the neural network results is less than 0.1%. The model proposed in this study is thus sufficiently accurate for estimating the spatial distribution of transmitter signals. More importantly, the computation time is significantly shortened using the present neural network model. To calculate the same spatial distribution, it takes approximately 1184 s if one uses the LWPC model, compared to 0.87 s if one uses the present neural network model. Thus, the computation speed fulfills the engineering need to estimate the spatial distribution of transmitter signals in real time.

We emphasize that traditional neural network models can estimate the VLF signals along single paths, but they are inefficient in estimating the spatial distribution of VLF signals near transmitters. To address this limitation, we have directly used maps of spatial distribution of transmitter signals as the training dataset and developed a neural network model. A key advantage of this model is its ability to significantly reduce the computation time while maintaining accuracy, even under complex ionospheric conditions Therefore, this model is better suited for atmospheric and space physics research, as well as various engineering needs.

Author Contributions: Data curation, X.G., S.W., Y.P. and D.Z.; methodology, H.S., X.G., W.X. and S.W.; validation, H.S., X.G., W.X., H.X. and W.C.; investigation, H.S., W.X., X.G., W.M. and J.F.; writing—original draft preparation, H.S., X.G. and W.X.; writing—review and editing, X.G., W.X., H.S. and B.N.; project administration, X.G., W.X., S.W. and B.N.; funding acquisition, X.G., B.N. and W.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key R&D Program of China (Grant No. 2022YFF0503700), the National Natural Science Foundation of China (Grant Nos. 42188101, 42274205, 42461160256, 42025404, 41874195), and the open fund of Hubei Luojia Laboratory (Grant No. 220100051).

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: We are grateful to the editor and anonymous reviewers for their assistance in evaluating this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Richardson, D.K.; Cohen, M.B. Seasonal variation of the D-region ionosphere: Very low frequency (VLF) and machine learning models. *J. Geophys. Res. Space Phys.* **2021**, *126*, e2021JA029689. [CrossRef]
2. Cheng, W.; Xu, W.; Gu, X.; Wang, S.; Wang, Q.; Ni, B.; Lu, Z.; Xiao, B.; Meng, X. A comparative study of VLF transmitter signal measurements and simulations during two solar eclipse events. *Remote Sens.* **2023**, *15*, 3025. [CrossRef]
3. Richardson, D.K.; McCormick, J.C.; Cohen, M.B. A D-Region Ionospheric Imaging Method Using Sferic-Based Tomography. *J. Geophys. Res. Space Phys.* **2023**, *128*, e2023JA031573. [CrossRef]
4. Marshall, R.A. *Very Low Frequency Radio Signatures of Transient Luminous Events Above Thunderstorms*; Stanford University: Stanford, CA, USA, 2009.
5. McCormick, J.C.; Cohen, M.B.; Gross, N.C.; Said, R.K. Spatial and temporal ionospheric monitoring using broadband sferic measurements. *J. Geophys. Res. Space Phys.* **2018**, *123*, 3111–3130. [CrossRef]
6. Cheng, W.; Xu, W.; Xiong, S.; Gu, X.; Ni, B.; Wang, C.; Zhang, Y.; Wang, S.; Feng, J.; Pan, Y.; et al. Spectra of GRB 221009A at Low Energies Derived from Ground-based Very Low-frequency Measurements. *Astrophys. J.* **2024**, *971*, 55. [CrossRef]
7. Richardson, D.K.; Cohen, M.B. Unifying VLF transmitter and sferic modeling efforts via tomography. *J. Geophys. Res. Space Phys.* **2023**, *128*, e2023JA031989. [CrossRef]
8. Xu, W.; Gu, X.; Ni, B.; Wang, S.; Yang, Z.; Cheng, W.; Hu, Z.; He, F.; Li, B.; Chen, X.; et al. Measurements and modeling of the responses of VLF transmitter signals to X-class solar flares at the Great Wall Station in Antarctica. *Space Weather* **2023**, *21*, e2022SW003249. [CrossRef]
9. Xu, W.; Marshall, R.A.; Kero, A.; Turunen, E.; Drob, D.; Sojka, J.; Rice, D. VLF measurements and modeling of the D-region response to the 2017 total solar eclipse. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7613–7622. [CrossRef]

10. Gu, X.; Wang, Q.; Ni, B.; Xu, W.; Wang, S.; Yi, J.; Hu, J.; Li, B.; He, F.; Chen, X.; et al. First results of the wave measurements by the WHU VLF wave detection system at the Chinese Great Wall station in Antarctica. *J. Geophys. Res. Space Phys.* **2022**, *127*, e2022JA030784. [CrossRef]
11. Cummer, S.A.; Inan, U.S.; Bell, T.F. Ionospheric D region remote sensing using VLF radio atmospherics. *Radio Sci.* **1998**, *33*, 1781–1792. [CrossRef]
12. Thomson, N.R. Experimental daytime VLF ionospheric parameters. *J. Atmos. Sol. Terr. Phys.* **1993**, *55*, 173–184. [CrossRef]
13. Lehtinen, N.G.; Inan, U.S. Radiation of ELF/VLF waves by harmonically varying currents into a stratified ionosphere with application to radiation by a modulated electrojet. *J. Geophys. Res. Space Phys.* **2008**, *113*, A06301. [CrossRef]
14. Carlson, B.E.; Lehtinen, N.G.; Inan, U.S. Constraints on terrestrial gamma ray flash production from satellite observation. *Geophys. Res. Lett.* **2007**, *34*, L08809. [CrossRef]
15. Marshall, R.A.; Inan, U.S. Two-dimensional frequency domain modeling of lightning EMP-induced perturbations to VLF transmitter signals. *J. Geophys. Res. Space Phys.* **2010**, *115*, A00E29. [CrossRef]
16. Bérenger, J.P. Use of the FDTD method for computing the VLF-LF propagation in the Earth-Ionosphere waveguide. In Proceedings of the 2006 IEEE Antennas and Propagation Society International Symposium, Albuquerque, NM, USA, 9–14 July 2006; pp. 3785–3788.
17. Ferguson, J.A. Computer Programs for Assessment of Long-Wavelength Radio Communications, Version 2.0. In *Users Guide and Source Files*; Space and Naval Warfare Systems Center Technical Document 3030; Defense Technical Information Center: Fort Belvoir, VA, USA, 1998.
18. McCormick, J.C.; Cohen, M.B. A new four-parameter D-region ionospheric model: Inferences from lightning-emitted VLF signals. *J. Geophys. Res. Space Phys.* **2021**, *126*, e2021JA029849. [CrossRef]
19. Han, F.; Liang, B.; Li, J.; Liu, F.; Lu, G.; Liu, Q. Mid-and low-latitude ionospheric D region remote sensing by radio atmospherics—Part I: Forward modeling and field measurement validations. *IEEE Trans. Antennas Propag.* **2019**, *68*, 1044–1054. [CrossRef]
20. Lipkova, J.; Chen, R.J.; Chen, B.; Lu, M.Y.; Barbieri, M.; Shao, D.; Vaidya, A.J.; Chen, C.; Zhuang, L.; Williamson, D.K.; et al. Artificial intelligence for multimodal data integration in oncology. *Cancer Cell* **2022**, *40*, 1095–1110. [CrossRef]
21. Xu, W.; Ma, W.; Wang, S.; Gu, X.; Ni, B.; Cheng, W.; Feng, J.; Wang, Q.; Hu, W. Automatic Detection of VLF Tweek Signals Based on the YOLO Model. *Remote Sens.* **2023**, *15*, 5019. [CrossRef]
22. Gross, N.C.; Cohen, M.B. VLF remote sensing of the D region ionosphere using neural networks. *J. Geophys. Res. Space Phys.* **2020**, *125*, e2019JA027135. [CrossRef]
23. Liu, B.; Wang, M.; Zhang, S.; Wang, X.; Nemati, S.; Li, H.; Yu, M.; Xu, J. Inversion of the Electron Density in the Lower Ionosphere Using Artificial Intelligence. *IEEE Antennas Wirel. Propag. Lett.* **2024**, *23*, 1618–1622. [CrossRef]
24. Wang, J.; Huang, Q.; Ma, Q.; Chang, S.; He, J.; Wang, H.; Zhou, X.; Xiao, F.; Gao, C. Classification of VLF/LF lightning signals using sensors and deep learning methods. *Sensors* **2020**, *20*, 1030. [CrossRef] [PubMed]
25. Xiao, L.; Chen, W.; Wang, Y.; Bian, K.; Fu, Z.; Xiang, N. Data-driven classification model of lightning generated LF/VLF radiation electric field waveforms. In Proceedings of the ICLP 2024 37th International Conference on Lightning Protection, Dresden, Germany, 1–7 September 2024; VDE: San Jose, CA, USA, 2024; pp. 694–698.
26. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1–9. [CrossRef]
27. Barat, C.; Ducottet, C. String representations and distances in deep convolutional neural networks for image classification. *Pattern Recognit.* **2016**, *54*, 104–115. [CrossRef]
28. Chulliat, A.; Macmillan, S.; Alken, P.; Beggan, C.; Nair, M.; Hamilton, B.; Woods, A.; Ridley, V.; Maus, S.; Thomson, A. *The US/UK World Magnetic Model for 2015–2020*; Natural Environment Research Council: Swindon, UK, 2015.
29. Wait, J.R.; Spies, K.P. *Characteristics of the Earth-Ionosphere Waveguide for VLF Radio Waves*; US Department of Commerce, National Bureau of Standards: Washington, DA, USA, 1964.
30. Ferguson, J.A. *Ionospheric Profiles for Predicting Nighttime VLF/LF Propagation*; Naval Sea Systems Command: Washington, DC, USA, 1980; p. 530.
31. Shao, W.; Shi, H.; Liu, J.; Zuo, Y.; Sun, L.; Xia, T.; Chen, W.; Wan, P.; Sheng, J.; Zhu, Q.; et al. Multi-instance Multi-task Learning for Joint Clinical Outcome and Genomic Profile Predictions from the Histopathological Images. *IEEE Trans. Med. Imaging* **2024**, *43*, 2266–2278. [CrossRef]
32. Wang, Y.; Luo, J.; Liu, C.; Yuan, X.; Wang, K.; Yang, C. Layer-wise residual-guided feature learning with deep learning networks for industrial quality prediction. *Trans. Instrum. Meas.* **2022**, *71*, 1–11. [CrossRef]
33. Ren, L.; Jia, Z.; Laili, Y.; Huang, D. Deep learning for time-series prediction in IIoT: Progress, challenges, and prospects. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 15072–15091. [CrossRef]
34. Tan, Q.; Liu, Y.; Liu, J. Demystifying deep learning in predictive spatiotemporal analytics: An information-theoretic framework. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 3538–3552. [CrossRef]

35. Byun, J.; Jun, C.; Kim, J.; Cha, J.; Narimani, R. Deep learning-based rainfall prediction using cloud image analysis. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–11. [CrossRef]
36. Karen, S.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
37. Ahad, M.T.; Li, Y.; Song, B.; Bhuiyan, T. Comparison of CNN-based deep learning architectures for rice diseases classification. *Artif. Intell. Agric.* **2023**, *9*, 22–35. [CrossRef]
38. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
39. Prantikos, K.; Chatzidakis, S.; Tsoukalas, L.H.; Heifetz, A. Physics-informed neural network with transfer learning (TL-PINN) based on domain similarity measure for prediction of nuclear reactor transients. *Sci. Rep.* **2023**, *13*, 16840. [CrossRef] [PubMed]
40. Pu, Y.; Chen, Y.; Dong, Y.; Zhang, K.; Wang, F.; Xi, X. Prediction the Diurnal Variation of VLF Waves in Earth-Ionosphere Waveguide Based on BPNN-TL Method. *IEEE Antennas Wirel. Propag. Lett.* **2024**, *24*, 38–42. [CrossRef]
41. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
42. Zhang, X.; Zhang, B.; Yu, W.; Kang, X. Federated deep learning with prototype matching for object extraction from very-high-resolution remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–16. [CrossRef]
43. Xiong, Y.; Li, Y.; Wang, C.; Shi, H.; Wang, S.; Yong, C.; Gong, Y.; Zhang, W.; Zou, X. Non-destructive detection of chicken freshness based on electronic nose technology and transfer learning. *Agriculture* **2023**, *13*, 496. [CrossRef]
44. Liu, Y.; Li, S.; Yan, W.; Suo, Y.; Wang, L.; Zou, Z.; Wu, L. High-resolution Digital Bathymetric Model (DBM) with Uncertainty Reconstructed by Deep Learning with Dropout. *IEEE Trans Geosci. Remote Sens.* **2024**, *62*, 1–17. [CrossRef]
45. Hwang, E.; Park, Y.S.; Kim, J.Y.; Park, S.H.; Kim, J.; Kim, S.H. Intraoperative hypotension prediction based on features automatically generated within an interpretable deep learning model. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 13887–13901. [CrossRef]
46. Sharma, O.; Trivedi, D.; Pattnaik, S.; Hazra, V.; Puhan, N.B. Improvement in district scale heavy rainfall prediction over complex Terrain of North East India using deep learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *61*, 1–8. [CrossRef]
47. Liu, Y.; Liu, W.; Yan, X.; Guo, X.; Zhang, C. Adaptive transfer learning for PINN. *J. Comput. Phys.* **2023**, *490*, 112291. [CrossRef]
48. Li, Y.; Wang, Y.; Qi, S.; Ren, Q.; Kang, L.; Campbell, S.D.; Werner, P.L.; Werner, D.H. Predicting scattering from complex nano-structures via deep learning. *IEEE Access* **2020**, *8*, 139983–139993. [CrossRef]
49. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proc. IEEE* **2020**, *109*, 43–76. [CrossRef]
50. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2009**, *22*, 1345–1359. [CrossRef]
51. Kim, H.E.; Cosa-Linan, A.; Santhanam, N.; Jannesari, M.; Maros, M.S.; Ganslandt, T. Transfer learning for medical image classification: A literature review. *BMC Med. Imaging* **2022**, *22*, 69. [CrossRef] [PubMed]
52. Pinto, G.; Wang, Z.; Roy, A.; Hong, T.; Capozzoli, A. Transfer learning for smart buildings: A critical review of algorithms, applications, and future perspectives. *Appl. Energy* **2022**, *5*, 100084. [CrossRef]
53. Jha, D.; Choudhary, K.; Tavazza, F.; Liao, W.; Choudhary, A.; Campbell, C.; Agrawal, A. Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning. *Nat. Commun.* **2019**, *10*, 5316. [CrossRef]
54. Dong, A.; Dou, J.; Li, C.; Chen, Z.; Ji, J.; Xing, K.; Zhang, J.; Daud, H. Accelerating cross-scene co-seismic landslide detection through progressive transfer learning and lightweight deep learning strategies. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 1–13. [CrossRef]
55. Liu, Q.; Hang, R.; Song, H.; Li, Z. Learning multiscale deep features for high-resolution satellite image scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *56*, 117–126. [CrossRef]
56. Gasdia, F.; Marshall, R.A. Assimilating VLF Transmitter Observations With an LETKF for Spatial Estimates of the D-Region Ionosphere. *IEEE Trans. Geosci. Remote Sens.* **2019**, *58*, 3526–3543. [CrossRef]
57. Chen, Y.; Ni, B.; Gu, X.; Zhao, Z.; Yang, G.; Zhou, C.; Zhang, Y. First observations of low latitude whistlers using WHU ELF/VLF receiver system. *Sci. China* **2017**, *1*, 166–174. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.