A Deep Convolutional Neural Network With Multiscale Feature Dynamic Fusion for InSAR Phase Filtering

Wang Yang, Yi He, Lifeng Zhang, Sheng Yao, Student Member, IEEE, Zhiqing Wen, Shengpeng Cao, Zhanao Zhao, Yi Chen, and Yali Zhang

Abstract—Interferometric phase filtering is a crucial step in the interferometric synthetic aperture radar (InSAR) data processing, which is also important for improving the accuracy of topography mapping and deformation monitoring. Most of the commonly used phase filtering methods perform windowing computations based on the statistical characteristics of a single interferogram in the spatial or frequency domain. However, the difficulty in taking into account the diversity and complexity of the phase image results in filtering methods with weak denoising, limited detail preservation, and poor generalization ability. At the same time, regardless of the spatial or frequency domain, improved phase filtering performance inevitably leads to the problem of declining effectiveness. This article proposes a phase filtering method based on the deep convolution neural network with multiscale feature dynamic fusion (MSFF). Unlike the traditional feedforward neural networks, the proposed method adopts a strategy of multiscale feature dynamic fusion that accounts for the deep and shallow features of the interferometric phase while also taking into account image detail preservation and noise suppression during phase filtering. Based on both subjective and objective evaluations, the experimental results using the simulated data prove that the proposed method has better noise suppression and detail preservation than the commonly used methods and that the filtering performance is less dependent on noise level. Experiments using the real data confirm that the proposed method has better generalization ability and can meet the precision requirements of practical applications. The method presented in this article can provide a new approach for research in high-precision InSAR data processing technology while also offering technical support for practical InSAR applications.

Index Terms—Deep convolutional neural network (DCNN), feature learning, interferometric phase filtering, interferometric synthetic aperture radar (InSAR).

I. INTRODUCTION

INTERFEROMETRIC synthetic aperture radar (InSAR) is a high-precision microwave interferometry method that can perform large-scale terrain reconstruction and deformation monitoring. During the InSAR processing, two SAR complex images are multiplied by conjugation to obtain an InSAR interferometric phase image [4]. However, the inherent characteristics of SAR imaging systems inevitably introduce phase noise due to spatial and temporal decorrelations, atmospheric delay effect, thermal noise of the system, and others [5], [6], [7], resulting in inaccurate terrain or deformation inversion. InSAR interferometric phase filtering can suppress this phase noise and preserve phase details [8], thereby improving the accuracy of terrain reconstruction and deformation monitoring. Consequently, establishing an efficient and accurate InSAR interferometric phase filtering method is very important.

The traditional phase filtering methods can be divided into spatial domain filter and frequency domain filter [9]. In spatial domain, defined (linear or nonlinear) operations, such as Pivoting median filter [10], Boxcar filter [11], Lee filter [12], NL-InSAR filter [13], and InSAR-BM3D filter are performed on the neighborhood pixels of the phase images to complete image smoothing and denoising [14]. Pivoting median filtering and Boxcar filtering are local processing methods that only take neighborhood pixels into account, resulting in reduced image resolution and blurred phase edges after filtering. By contrast, NL-InSAR and InSAR-BM3D filtering methods carry out image smoothing and detail optimization based on similar patches in the interferometric phase images [15]. These methods improve performance, but they require huge amounts of calculation and reduce the efficiency of filtering processing. In addition, NL-InSAR and InSAR-BM3D filtering methods may introduce a staircasing effect [16], resulting in insufficient detail preservation. As for frequency domain filtering, such as Goldstein filter and local frequency compensation filter [17], [18], they transform the phase into the frequency domain by Fourier transform and perform thresholding or weighting according to the frequency distribution of the noise and phase
information to achieve noise suppression and detail preservation [19], [20], [21]. These methods have improved the ability to preserve details due to the ease of distinguishing frequency information. However, the lack of a reliable reference in frequency threshold processing often leads to an inability to suppress noise and preserve details, thereby resulting in poor generalizability [22]. For example, Goldstein filter endeavors to improve the ability to preserve details by restraining noise suppression, but this results in under smoothing in low-coherence areas and over smoothing in high-coherence areas.

According to the above, the problems currently faced by interferometric phase filtering can be roughly summarized as follows: 1) noise suppression is insufficient and detail preservation is limited, especially in areas with low-coherence and dense fringes; 2) model generalization ability is poor, and the filtering performance depends strongly on the noise level; and 3) efficiency is sacrificed while improving filtering performance.

In recent years, the application of deep convolutional neural networks (DCNNs) for interferometric phase filtering has become a hot topic in the research of InSAR data processing technology. One advantage of these networks is that they can utilize a large number of simulated phase images for supervised autonomous learning without a need for strict selection of features; rather, guided learning alone is required to achieve the desired goal [23], [24]. Among the existing DCNN-based filtering models, interferometric phase denoising convolutional neural network (IPDcNN) and Φ-net has a significant improvement in performance [25], [26], [27]. IPDcNN adopts the traditional feedforward sequential convolutional neural network, through residual learning, estimates the noise in the interferometric phase, and then removes the noise from the interferometric phase to obtain the filtered phase, this method can reduce the phase noise while protecting fringe edges and avoid the use of filter windows, and far outperforms traditional filtering methods in performance. Φ-net is based on U-net, and replaces the single U-net layers with residual blocks, the cascade of encoder and decoder stages used in combination with skip connections and residual shortcuts enables an effective representation of the interferometric signal and the superimposed noise. Φ-net framework takes into account the ability of U-net to restore information at different scales and the generalization ability of the model applied to the real phase. The IPDcNN and Φ-net show the great potential of DCNN in phase filtering, however, they cannot take into account the different feature expression capabilities of feature maps at different scales, which also represent the importance of feature maps at different scales, this importance affects the performance and generalization ability of the model, therefore, in order to continue to optimize the performance of filtering methods, the importance of feature maps at different scales should be considered.

The published literature shows that extracting the interrelated information between channels can effectively take into account the different scales of semantic information in the feature extraction process [28], thereby avoiding the loss of information and the lack of model generalization ability caused by the change of the spatial scale of the feature map with pooling [29]. By contrast, the Squeeze and Excitation block (SE-block) can provide an attention mechanism for dynamic adjustment of feature weights after multiscale feature fusion (MSFF) to improve network training efficiency and network performance [30]. Inspired by this observation, we adjusted the traditional feedforward sequential neural network into an MSFF DCNN to account for deep and shallow features during model training. We constructed a multidepth fusion feature map and then used SE-block to dynamically adjust the weight distribution of the feature maps in the channel dimension to form a channel-dimensional attention mechanism. The resulting multiscale feature dynamic fusion model considers multilevel semantic information and is therefore helpful for both preservation of details and noise removal in interferometric phase filtering.

Obtaining a large number of real clean images as supervision for network model training is impossible in practice [31]; therefore, simulation methods are needed to generate datasets. Interferometric phase image simulation based on the real digital elevation model (DEM) is a previously used dataset simulation method [32], but using this method to simulate the interferometric phase is not economical. On the one hand, the huge amount of data in a real terrain imposes very high hardware requirements for phase image simulation. On the other hand, training of the network model for interferometric phase image filtering does not require simulation of the real terrain data of a specific area; instead, the hope is that terrain with different characteristics can be flexibly generated to meet different requirements. In addition, when using the real DEM to synthesize the simulated phase to ensure the reliability of the sample and the rationality of the sample distribution, the DEM must be calibrated and screened, which also increases a lot of labor costs. The application of fractal technology can generate terrain and solve this problem, as this technology can generate simulation data that approximate the real terrain with a small amount of calculation. One mathematical fractal method for simulating terrain is the diamond square algorithm [33], which can simulate terrain of any type and roughness by adjusting random parameters. When combined with InSAR geometry parameters, the diamond square algorithm can generate a large number of multitype simulated phase images for model training and testing. Therefore, the present article uses the diamond square algorithm based method to randomly simulate phase images that closely approximates the real situation.

The main objectives of this article are 1) to simulate and generate large amounts of simulated interferometric phase images for model training and testing by combining the diamond square method and the InSAR phase computing method; 2) to build a DCNN model with an MSFF network (named MSFF-DCNN) by combining it with SE-block; and 3) to compare the proposed method and commonly used methods based on both simulated and real data. The visual performance of the filtering method is assessed with subjective evaluation methods, and the subjective evaluation is verified and supplemented by objective evaluation methods.

II. MATERIALS AND METHODS

In this article, the InSAR interferometric phase simulation method based on the diamond square algorithm is first described to create the dataset required for model training and testing. The
The proposed MSFF-DCNN model is then built to predict simulated data and real data. Finally, filtering performance evaluation methods, including subjective evaluation and objective evaluation of interferometric phase filtering are used to evaluate the performance of the MSFF-DCNN model [9]. Fig. 1 shows the flow chart of the research idea in this article, and the methods in this article are also elaborated in this section.

A. Dataset

In practice, we cannot obtain a real InSAR interferometric phase with corresponding clean phase images, but the network requires a large amount of interferometric phase images with label images for training [34]. Therefore, simulated data are used as the input for network training. In this article, the simulation method for the interferometric phase images consists mainly of two steps: the first step is terrain data simulation based on the diamond square algorithm to provide a simulated terrain for interferometric phase calculation, and the second step is calculation of the interferometric phase and transformation according to the simulated terrain and InSAR interferometric geometry. The end result is the generation of the dataset needed for model training and testing.

1) Terrain Data Simulation: The diamond square algorithm starts with a two-dimensional (2-D) array of squares with a width and height of $2n + 1$ ($n$ starts at 0 in this article). It first sets the four corners of the array to random initial values within the elevation range (set to 0–1500 in this article), and then performs the diamond and square steps recursively, until a random matrix of a certain size (set to $256 \times 256$ in this article) is generated as simulated terrain data. Each recursion consists of two steps: the diamond step and the square step.

Fig. 2 shows the first two recursions of the diamond square method, and we briefly explain the principle of the diamond square algorithm.

1) Diamond step: finding the midpoints of the square

For each square in the 2-D array, the midpoint of the square is set as the average value of the four corners plus a random value. A larger range of random values will generate a rougher simulated terrain. In this article, the random value is set at $[-100, 100]$.

2) Interferometric Phase Calculation and Transformation:

2) Square step: Finding the value of the midpoints on each side of the square

For each diamond in a 2-D array, the midpoint of the diamond is set to the average of the four corners plus a finite random value.
Fig. 2. Schematic of diamond square algorithm, showing two recursions.

Fig. 3. As the number of recursive increases, the generation process of terrain data.

We recursively use these two steps to generate simulated terrain data with various distribution characteristics. Fig. 3 shows the process used for generation of the terrain data with increasing recursion times.

The calculation of the simulated interferometric phase first requires radar-coding of the terrain model; that is, the geometric parameters of the radar satellite interferometry are used to convert terrain data from the terrain coordinate system to the radar coordinate system [35]. The relative position of the radar satellite and terrain in the radar coordinate system is then used to calculate the interferometric phase, and the phase noise of the zero mean additive Gaussian distribution with a different level is added to obtain the interferometric phase with noise [36]. The input and output data of the network are then obtained by transforming the interferometric phase from a real to a complex field. The process for interferometric phase calculation and transformation is shown in Fig. 4.

1) Radar-coding of terrain data and interferometric phase calculation

Fig. 5 shows two images of the same target $P$ along slightly deviated orbits. At $S_1$ and $S_2$, the radar sensor transmits electromagnetic waves to the ground target and receives its echo signal, respectively. Here, $h$ is the distance from the target point $P$ to the reference ellipsoid along the normal direction, and $R_1$ and $R_2$ are the slant distances from point $P$ to $S_1$ (corresponding to the main image) and $S_2$ (corresponding to the secondary image), respectively. $B$ is the baseline during the two satellite transits, $B_{\perp}$ is the vertical baseline, $\theta$ is the incident angle, and $\alpha$ is the angle of the baseline relative to the reference horizontal plane. In this article, the radar-coding parameters are set to random values that conform to the satellite attitude and position parameters of sentinel-1A, as shown in the box in Fig. 5.

According to the parameter settings of the Radar-coding and the relative position between the ground objects and the radar satellites, the clean wrapped interferometric phase at point $P$ can
be expressed as [37]

\[ \varphi_{\text{clean}} = \operatorname{wrap} \left( \frac{4\pi}{\lambda} \times \frac{B_1 h}{P_1 \sin(\theta)} \right) \]

(1)

where \( \varphi_{\text{clean}} \) is the noise-free phase image and \( \operatorname{wrap}(\cdot) \) denotes a wrapping function that returns the wrapped phase [38].

To obtain the noisy phase (\( \varphi_{\text{noisy}} \)) corresponding to the clean phase (\( \varphi_{\text{clean}} \)), random additive Gaussian noise (\( v_G \)) is added to the clean phase in phase simulation [39]

\[ \varphi_{\text{noisy}} = \varphi_{\text{clean}} + v_G \]

(2)

where \( \varphi_{\text{clean}} \) and \( v_G \) are two independent variables, the estimated \( \varphi_{\text{clean}} \) will be obtained by filtering the \( \varphi_{\text{noisy}} \).

2) Phase transformation and dataset generation

The interferometric phase has a wrapping property and changes periodically between \((-\pi, \pi]\). A correct calculation of the phase gradient for phase unwrapping requires preservation of the phase jump between \(-\pi \) and \( \pi \) [40]. If the interferometric phase value is directly used as the input and output of the network, the phase edge can easily be mistaken for noise, leading to instability of the network training and a poor denoising effect. Therefore, (3) is used to convert the simulated interferometric phase from the real to the complex field, and the real and imaginary parts are trained as the input and output of the network, respectively. The prediction results are returned to the real field phase by (4), thereby avoiding the network training instability caused by the phase jump and the poor filtering effect of the model. Fig. 6 shows the cross-section of the interferometric phase fringes in the dataset transformation process, and the figure shows that the phase jump disappeared after the transformation of the real into the complex field.

The interferometric phase in complex number field can be expressed as

\[ \varphi_{\text{cpx}} = j \cdot \sin(\phi) + \cos(\phi) \]

(3)

where \( \varphi_{\text{cpx}} \) is the complex phase and \( j \) is the imaginary unit.

After phase filtering, the interferometric phase in the real number field can be calculated by

\[ \varphi = \arg(\varphi_{\text{cpx}}) \]

(4)

where \( \varphi \) is the phase in real number field and \( \arg(.) \) is the function to find the argument of \( \varphi_{\text{cpx}} \).

The cross-section of the phase is shown in Fig. 6.

Steps (1) and (2) were recycled to generate 5000 pairs of 256x256 real and complex interferometric phases. Ultimately, 4000 pairs of real and imaginary training sets and 1000 pairs of testing sets were generated. Fig. 7 shows representative sample data before and after phase transformation.

B. Squeeze and Excitation Block (SE-Block)

In feature learning, the SE-block can provide an attention mechanism that is responsible for adjusting the weight of the channel dimension of the feature maps, fitting the complex correlation of different types or different scales of features, and improving the performance of the network, while reducing the time complexity and spatial complexity [41]. Therefore, we embedded an SE-block in the proposed network model. Next, we introduce details of the structure and principle of the SE-block. The SE-block can efficiently utilize different scales or different types of feature maps. After fusion of different scales or different types of features, the SE-block is added to explore the weight relation of the different features. As shown in Fig. 8, the SE-block can assign different weights to different channel dimensions of the feature map. This block is used to obtain feature vectors from the fused feature map through global average pooling, and then to add nonlinear processing through two dense layers to fit the complex channel dimension correlation between features. The number of output units of the first dense layer is 1/3 of the channel dimension of the fusion feature map, while the number of output units of the second dense layer is the same as that of the channel dimension of the fusion feature map. The sigmoid function is used in the second dense layer to return a vector of weights between 0 and 1. Ultimately, the combined weight vector and fused feature map are obtained by Hadamard product operation, and the different channels are given different weights to obtain the weighted feature map that is used as the basis for the final prediction.

In general, the SE-block can be divided into two steps: spatial squeeze and channel excitation. The spatial squeeze step produced a compressed feature vector \( \hat{D} \in \mathbb{R}^{1 \times 1 \times C} \) by a global average pooling layer, and the \( C \)th element of the vector \( D \) can be obtained by [42]

\[ d_c = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} V_C(x, y). \]

(5)

The step of channel excitation step uses the vector \( D \) to obtain the importance of the feature map (\( V \)) in the channel dimension. The vector \( D \) is adjusted to \( \hat{D} = P_1(\delta(P_2 V)) \), \( P_1 \in \mathbb{R}^{C \times C} \), \( P_2 \in \mathbb{R}^{C \times C} \), and \( P_1 \) and \( P_2 \) are the parameter used to adjust the weight distribution between each channel. Through the activation function [sigmoid function \( \sigma(\cdot) \)], the range of \( \hat{D} \) is adjusted to be between 0 and 1. Finally, the weight-adjusted feature map \( \hat{V}_{\text{SE}} \) is the Hadamard product of \( \sigma(\hat{d}_c) \) and \( V \) [Formula (6)]

\[ \hat{V}_{\text{SE}} = F_{\text{SE}}(V) = [\sigma(\hat{d}_1)V_1, \sigma(\hat{d}_2)V_2, \ldots, \sigma(\hat{d}_C)V_C]. \]

(6)
The scalar $\sigma(\hat{d}_i)$ is the weight of each channel dimension, which means the importance of the $i$th channel. During the feature learning, these weights are adaptively adjusted to weaken the unimportant channels and emphasize the important ones.

### C. MSFF-DCNN Network Structure

This study proposes an MSFF-DCNN for InSAR interferometric phase filtering. The model adopts a dual-input, dual-output structure, with the main structure mainly divided into three processes: downsampling and feature encoding; upsampling and MSFF; and image reconstruction and generation. The structure of MSFF-DCNN is shown in Fig. 9.

1) **Downsampling and Feature Encoding**: The downsampling and feature encoding process occurs from input to dropout-1. First, features are extracted using 2-D convolution for real and imaginary inputs, and then the concatenate layer and SE-block are used to integrate the real and imaginary features and weight the channel dimensions. Finally, features are encoded by stepwise downsampling using the layers of 2-D-convolution, max-pooling, batch normalization (BN), and dropout. In this process, the size of convolutional kernels is $3 \times 3$, and the filling method of the feature map is set to “same” to ensure a consistent size of the output feature maps and the current one. The output channel of the convolutional layer increases as the size of feature maps decreases, with a gradual increase from a minimum of 32 to 128 256. The size of the feature map is reduced from the original $256 \times 256$ to $64 \times 64$ after two downsampling operations with the same scale. During the pooling process, the window size for pooling under the maximum-size feature map is $2 \times 2$, the step size is 2, and the feature map filling method of all pooling layers is “valid”; therefore, no repetition occurs under the premise of completely traversing all pixels. The pixel values are calculated,
and the width and height of the feature map are reduced to the desired size.

In the process of deep network training, due to the internal covariate shift, the network fitting speed will be slower, so the required learning rate will be smaller. This not only reduces the generalization ability of the model, but it also increases time consumption in training. BN can effectively solve this problem [43]; therefore, we use BN after the two convolutions in CCBD-i (i = 1, 2, 3, 4) (the CCBD block contains two convolutional layers, a BN layer, and a dropout layer) to speed up the network training and improve the ability of generalization. According to published literature and a large number of subsequent experimental demonstrations [44], dropout can effectively prevent the occurrence of overfitting, and achieve the effect of regularization to a certain extent. Therefore, adding a dropout layer after BN drops neurons with a probability of 0.25 to prevent the network from overfitting.

2) Upsampling and Multiscale Feature Fusion: The upsampling and MSFF process uses multilayer and multipath upsampling and convolution to generate multiscale feature maps, which are then dynamically fused using concatenate layer and the SE-block. Path 1 starts from dropout-3, and goes through upsampling-1, Conv-9, upsampling-2, and Conv-10, to enlarge the feature map size from 64 × 64 to 256 × 256, and generate the first-scale feature map. Path 2 starts from dropout-1 and uses one convolution (Conv-5-2) directly as the second-scale feature map without resizing. Path 3 starts from dropout-2, and goes through an upsampling (upsampling-1-2) and convolution (Conv-7-2), thereby expanding the feature map size from 128 × 128 to 256 × 256 to generate the third-scale features. We then use concatenate-2 to splice the three-scale feature maps in the channel dimension to generate a feature matrix that takes into account the deep and shallow features. The use of SE-2 then dynamically adjusts the feature weight of the channel dimension to generate a feature map with a channel attention mechanism. In the three upsamplings in this process, the sampling factor is 2 in both rows and columns, the interpolation method is “nearest,” and the convolution kernel size of the convolutional layer connected after each upsampling layer is the same as that of the upsampling factor (2 × 2) to reduce the mosaic effect caused by the nearest neighbor algorithm in the upsampling layer [45].

3) Image Reconstruction and Generation: The image reconstruction and generation process includes gradient control and image detail reconstruction. After the network is trained through SE-2, the weight change of the channel dimension may cause a gradient disappearance problem. Therefore, we use CCBD-4 to avoid this gradient problem. After CCBD-4, the feature map is split into two paths, and Conv-13-1/2 and Conv-14-1/2 are used to gradually restore the imaginary and real parts of the interferometric phase. In this process, the number of output channels of the convolution layer continues to decrease. As the convolution proceeds, the number of channels gradually decreases from 128 to 32. The size of the convolution kernel in CCBD-4 is 3 × 3, after the feature is split, the sizes of the Conv-13-1/2 and Conv14-1/2 convolution kernels become 2 × 2 and 1 × 1, respectively, and their function is to gradually reduce the feature dimension and restore the real and imaginary image details.

Fig. 9. Structure of MSFF-DCNN.
TABLE I
CONFIGURATIONS OF EXPERIMENTAL ENVIRONMENT

| Item     | CPU              | GPU              | Python | Tensorflow | Keras |
|----------|------------------|------------------|--------|------------|-------|
| Version  | Intel Core i7-10700F | NVIDIA GeForce RTX 3090 | 3.9.8  | 2.7.1       | 2.6.0 |

Fig. 10. Loss and metrics performance of MSFF-DCNN.

D. Model Training

In this article, the hardware is at the current mainstream level, and the model design and training are based on the Keras deep learning framework. The environment configurations of experiments are shown in Table I.

After many experiments, and considering the three aspects of model calculation efficiency, result accuracy, and hardware conditions, in the final training parameters, epoch is set to 128, batch size is set to 12, adaptive motion estimation is used as the optimizer [46], and the initial learning rate is set to $10^{-4}$. The shape of the feature map temporarily stored in the calculation process is large; therefore, a half-precision (16-bit) floating-point format model is used for training. In the gradient descent algorithm, the index for evaluating the difference of pixel values or the index for evaluating the similarity of image structure alone cannot effectively preserve the details and maintain the resolution of the interferometric phase at the same time; therefore, the custom loss function $L(\Theta)$ is a weighted combination of mean absolute error (MAE) and structural similarity (SSIM) [47], [48], the formula is as follows:

$$L(\Theta) = W_{MAE} \times MAE + W_{SSIM} \times SSIM$$

(7)

where $\Theta$ is the trainable parameters of MSFF-DCNN model, $W_{MAE}$ and $W_{SSIM}$ are the weights of MAE and SSIM, respectively, $W_{MAE} = 0.5$, $W_{SSIM} = 0.5$ and is the setting of this experiment.

Where expression of MAE is

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |\hat{\varphi}_i - \varphi_i|$$

(8)

where $n$ is the number of samples, and $\hat{\varphi}_i$ and $\varphi_i$ are the filtered and clean image, respectively.

Where expression of SSIM is

$$SSIM(\varphi, \hat{\varphi}) = \frac{(2\mu_\varphi \mu_{\hat{\varphi}} + (k_1L)^2)(2\sigma_{\varphi,\hat{\varphi}} + (k_2L)^2)}{(\mu_\varphi^2 + \mu_{\hat{\varphi}}^2 + (k_1L)^2)(\sigma_{\varphi}^2 + \sigma_{\hat{\varphi}}^2 + (k_2L)^2)}$$

(9)

the $\mu_{\varphi}$ and $\mu_{\hat{\varphi}}$ are the mean of the clean image and the filtered image, respectively, $\sigma_{\varphi}$ and $\sigma_{\hat{\varphi}}$ are the standard deviations of them, $\sigma_{\varphi,\hat{\varphi}}$ is the covariance of the phase matrix of the clean and the filtered image, $L$ is the dynamic range of the phase images, $k_1$ and $k_2$ are two constants, where $k_1 = 0.01$ and $k_2 = 0.03$ by default. and its function is to avoid the instability of the calculation result due to the denominator being too small.

As can be seen from Fig. 10, the network quickly reaches convergence. The SSIM metrics of the network models in real and imaginary parts of training set and that in testing set are similar, in the end, they all converge approximately to 0.92. The loss finally converges to approximately 0.12. It shows that our model has faster fitting effect and better prediction accuracy.

III. FILTERING PERFORMANCE EVALUATION

This article provides a detailed evaluation of the performance and efficiency of the proposed PSFF-DCNN. The performance indicators of the interferometric phase filtering method are divided into noise-suppression ability and detail-preservation ability; therefore, this article uses subjective evaluation methods to evaluate the performance of the phase filtering methods through visual observation, and objective evaluation methods to verify and supplement the judgment of the objective evaluation through quantitative evaluation.

A. Subjective Evaluation Methods

Subjective evaluation is the most direct evaluation method. It allows the observer to judge a filtering result by visual
observation, and the results are based on predefined evaluation content or personal experience. The detail-preservation ability of the phase image is evaluated by the preservation of image resolution, the clarity of the phase edge, the phase fringe periodicity, and the completeness of the phase jump.

B. Objective Evaluation Methods

The aims of the objective evaluation index are to evaluate the filtering results by defining some quantitative evaluation indexes, and then to express the results automatically, quickly, and quantitatively. Here, the objective evaluation indicators we use are MSE, SSIM, and a phase error frequency distribution graph for simulated phase filtering. The MSE evaluates the difference between the filtered image and the corresponding pixels of the clean phase. A smaller MSE yields better model performance. SSIM evaluates the correlation between pixel points after model filtering. This correlation contains structural information of ground objects and is suitable for a highly structured and terrain-dependent InSAR interferometric phase.

For real data filtering, since the ground truth is unknown, we can only choose some no-reference evaluation methods, such as the number of residues (NOR) [17], the percentage of the reduced residues (PRR), the no-reference matrix $Q$, and the residual phase standard deviation (RPSD) [49], [50].

In phase unwrapping, the residues make the unwrapping results based on the phase gradient integration relative to the path of the integration, while the region unwrapping results that do not contain residues are path independent. Therefore, the NOR is one of the most important evaluation indicators for the quality of interferometric phase filtering. In general, the residues in the area with low real data coherence cannot be completely removed; therefore, the PRR can also be used as an evaluation method. As for RPSD, the smaller the RPSD, the smoother the phase after filtering, and the stronger the noise-suppression ability of the filtering method. The no-reference metric $Q$ can comprehensively reflect the noise-suppression ability and detail-preservation ability of the filtering methods. A higher $Q$ means that the accuracy of the filtering method is better.

IV. RESULTS AND ANALYSIS

We apply simulated and real data with uniform and nonuniform noise level to the proposed model, and compare it with five commonly used interferometric phase filtering methods: Pivoting median filter, Lee filter, Goldstein filter, NL-InSAR filter, and $\Phi$-Net, which is a new DCNN-based filtering model. We evaluate their abilities for noise suppression, detail preservation, and model generalization using subjective and objective evaluation methods. The parameter settings of the four filtering methods are shown in Table II.

A. Simulated-Data Experiments

In practical application, the level of phase noise is not consistent in spatial distribution, but is instead generally manifested as uniform and nonuniform distributions of noise level [34]. Therefore, we use simulated interferometric phase images with uniform and nonuniform noise level to evaluate the ability of the proposed MSFF-DCNN model for noise suppression and detail preservation in different scenarios, as well as to assess the dependence of the model performance on noise level.

1) Data With Uniform Noise: The selected image from testing set is a simulated interferometric phase with uniform noise distribution, and it is filtered using the proposed MSFF-DCNN method and compared with the interferometric phase filtering methods shown in Table II. Furthermore, two typical areas of patch A with sparse fringes and patch B with dense fringes are cropped from the whole interferometric phase image to evaluate the performance of the proposed MSFF-DCNN model in more detail. Fig. 11 shows the noisy phase and the clean reference.

Subjective evaluation: Fig. 12 shows that the phase loses detailed information after filtering with the Pivoting median filter and that the resolution is significantly reduced. This is especially the case in patch B, where the fringe edges of the interferometric phase are blurred, the fringe structure is severely damaged, and a large amount of fringe edge structure remains in the phase error image. Overall, these observations indicate a very limited filtering performance with this method. The fixed size and orientation windows of the pivoting median filter counteracts the noise suppression ability, thereby affecting the filtering effect and resulting in loss of detailed information and a reduced resolution of the filtered image [50].

As for Lee filter, because the Lee filter cannot accurately determine the fringe direction in the dense interferometric fringe area [51], Lee filtering results in insufficient detail preservation where fringes are dense. This leads to a loss of resolution of the whole image, as the fringe structure in patch B is damaged and the local denoising ability is insufficient. The Goldstein filter is a commonly used frequency domain filtering method. Unlike the case for the Pivoting median filter and Lee filter, the fringe
structure in patch A and B is complete and continuous with the Goldstein filter. However, a large number of phase residues occur at the phase edges in the whole and patch phase error images, and bluish and yellowish phase residues appear in the phase error image of patch B. This indicates that the method carries useful phase information while filtering out noise, resulting in the loss of phase details. In addition, this method limits the algorithm performance due to the lack of a reliable reference for the values of its filtering parameters [52]. By contrast, the NL-InSAR algorithm shows a significant improvement in noise suppression and detail preservation compared to aforementioned three methods; however, the phase error map still shows retention of the bluish and yellowish phase errors at the phase continuity (this is more obvious in patch B). The phase detail information is lost, although the phase edge remains clear and sharp.

The above four traditional interferometric phase filtering methods still retain the defects of weak noise suppression and limited detail preservation in subjective evaluation. By contrast, Φ-net and proposed filtering methods have superior performance, the whole image and the patch images after filtering are smooth and clear. In further comparison, there are fewer phase errors in the phase error images after filtering with the proposed method than with the Φ-net, and the overall error image appears green, with no apparent bluish and yellowish residual
TABLE III
OBJECTIVE EVALUATION UNDER THE CONDITION OF UNIFORM NOISE DISTRIBUTION

| Methods     | Image   | MSE     | SSIM   | Time(s) |
|-------------|---------|---------|--------|---------|
| Pivoting median | Whole   | 1.44    | 0.48   | 3.5     |
|             | Patch A | 1.63    | 0.51   |         |
|             | Patch B | 1.78    | 0.44   |         |
| Lee         | Whole   | 1.31    | 0.53   | 11.8    |
|             | Patch A | 1.52    | 0.60   |         |
|             | Patch B | 1.49    | 0.49   |         |
| Goldstein   | Whole   | 0.83    | 0.66   | 11.2    |
|             | Patch A | 0.78    | 0.66   |         |
|             | Patch B | 0.81    | 0.62   |         |
| NL-InSAR    | Whole   | 0.66    | 0.74   | 165.6   |
|             | Patch A | 0.59    | 0.88   |         |
|             | Patch B | 0.47    | 0.78   |         |
| Φ-Net       | Whole   | 0.29    | 0.89   | 0.13    |
|             | Patch A | 0.20    | 0.91   |         |
|             | Patch B | 0.21    | 0.94   |         |
| Proposed method | Whole   | 0.27    | 0.91   | 0.19    |
|             | Patch A | 0.19    | 0.94   |         |
|             | Patch B | 0.22    | 0.95   |         |

Fig. 13. Phase error frequency distribution graph, the part of the graph is enlarged in the box. The sharper the curve is at 0, the better the noise suppression of the filtering method is. The closer the curve is to the x-axis at 2π or −2π, the better the detail preservation of the filtering methods.

Information. In short, the proposed MSFF-DCNN model in this article has strong visual noise suppression ability and can maintain good detail information and image resolution under the conditions of sparse or dense fringes.

Objective evaluation: The judgment from subjective evaluation was further verified and supplemented using SSIM and MSE in Section III (Table III), combined with the phase error frequency distribution graph (Fig. 13) to evaluate the ability
of the proposed model to suppress noise and preserve details. The efficiency of the filtering method was evaluated using the time $T$ spent in filtering each image. In both the whole and patch images, the proposed MSFF-DCNN model significantly outperforms Pivoting median filter, Lee filter, and Goldstein filter on MSE and SSIM. Compared with NL-InSAR, the MSE of the proposed method is 59.1% lower, and the SSIM is 23.0% higher in the whole image, while the MSE of the MSFF-DCNN model is 67.8% lower and the SSIM is 6.8% higher in patch A, and the MSE of the MSFF-DCNN model is 53.2% lower and the SSIM is 21.8% higher in patch B. Compared with $\Phi$-net, the proposed MSFF-DCNN model also has a slight advantage in the performance of MSE and SSIM. In terms of efficiency, both the proposed model and $\Phi$-net show efficiency advantages that traditional methods cannot achieve, especially the NL-InSAR filter.

The phase error frequency distribution graph (Fig. 13) and its enlarged graphs and show that the frequency distributions of the proposed model for the whole image and for patch A and B are more concentrated toward 0, when the phase error is around $2\pi/2\pi$, the frequency distribution curve is closer to the $x$-axis, showing that the proposed model has better edge preserving ability.

2) Data With Nonuniform Noise: In Section I, the coherence of the simulated interferometric phase is approximately equal in different areas; however, in practical applications, different noise levels may occur in different positions in the image to be filtered [53]. At the same time, the same filtering method may show different performance in areas with different noise levels. Many experiments have now revealed that the filtering method has no obvious performance difference under extreme conditions where the coherence of the interferometric phase image is lower than 0.2 or higher than 0.8 [14]. On this basis, we designed four InSAR interferometric images with spatially varying coherences (Fig. 14) to evaluate the dependence of the proposed method on noise level and the generalization performance of the method. The four simulated interferometric phase images are: a cone with a constant gradient along the radius and with a fringe separation of 16 pixels; a straight ramp with a varying gradient and a fringe separation that changes from 32 pixels to 8 pixels; a pyramid with a constant gradient and some fringe corners with a fringe separation of 12 pixels; and a peak produced by superposition of some normally distributed surfaces to simulate a scene with mountains and plains. To further investigate the dependence of the performance of the filtering method on the noise level, in the phase simulating, we set the coherence to be 0.2–0.8 from top to bottom.

Subjective evaluation: Fig. 15 shows that the five filtering methods exhibit different performances in the filtering of interferometric phase images with varying coherence. The Pivoting median filter has poor filtering performance in the low-coherence area, the resolution is reduced, the fringe structure is seriously damaged, and a large amount of fringe information remains in the phase error images. The pyramid and peak render fringes at the tops of the images that are almost indistinguishable.

The Lee filter provides significant improvements over Pivoting median filter, but a large amount of phase errors remain at the corners of the pyramid phase error image, while the phase error image of the peak contains a large area of bluish and yellowish phase errors. This is because the direction window selection mechanism cannot accurately determine the fringe direction in areas with dense fringes or rapid changes in fringe direction; therefore, the filtering results are deviated [54].

In the Goldstein filtered image, the interferometric phase fringe structure is destroyed in the areas with low-coherence, and more phase errors remain on the top edge in the phase error images. In the areas with high-coherence, the interferometric phase fringe structure is well maintained, and fewer phase errors are retained on the bottom edge in the phase error image. However, the Goldstein filtering method is very sensitive to the noise level largely because the filtering performance relies on the estimation of the frequency components [55]. When the phase is not sufficiently reliable, the reliability of the frequency estimation also drops rapidly, resulting in poor noise suppression in areas of low-coherence. However, the characteristics of Goldstein filter processing in the frequency domain prevent the retention of large amounts of phase errors in the phase error images. NL-InSAR shows a small area of phase structure damage at the top of the pyramid and peak, and the image resolution is well maintained and few phase errors remain in the phase error images, indicating that the method has strong abilities for noise suppression and image resolution preservation. However, corner phase errors remain in the phase error image of the pyramid, and bluish and yellowish phase errors appear in the dense fringes of the peak, indicating that the method has a limited ability to preserve details and still has a strong dependence on noise level.

Compared with the traditional four methods, the filtering performance of $\Phi$-net has been significantly improved, and the fringe structure after filtering can be almost completely preserved. But by observing the phase error images, we find that the obvious edge structure is still preserved in the low-coherence area, which shows that the filtering performance of $\Phi$-net has a moderate dependence on the noise intensity. Therefore, the filtering performance of $\Phi$-net in low-coherence regions still has room for optimization.

The proposed MSFF-DCNN model shows strong filtering performance in all four simulated images. The interferometric phase image is smooth after filtering, the resolution is well maintained, the edges are clear, and the phase errors are fewer. Compared
Fig. 15. Filtered image and its corresponding phase error image.
TABLE IV

| Objective Evaluation Under the Condition of Nonuniform Noise Level |
|----------------------|----------------|--------------|
|                      | MSE       | SSIM       | Time(s)     |
| Noisy phase          | 3.71      | 0.44        |             |
| Pivoting median      | 1.62      | 0.69        | 3.7         |
| Lee                  | 1.02      | 0.72        | 12.1        |
| Goldstein            | 0.94      | 0.76        | 11.5        |
| NL-InSAR             | 0.57      | 0.87        | 155.7       |
| Ф-net                | 0.39      | 0.91        | 0.13        |
| Proposed method      | 0.35      | 0.93        | 0.21        |

![Fig. 16. Variation of MSE and SSIM with coherence.](image)

Fig. 16. Variation of MSE and SSIM with coherence.

with the other five filtering methods, only small amounts of phase errors remain at the top and bottom in the phase error images, indicating that the proposed MSFF-DCNN model has superior abilities for noise suppression and detail preservation and that the filtering performance has the least dependence on noise level. However, the proposed MSFF-DCNN method also has some (albeit tolerable) problems: the proposed method has a small amount of corner phase errors retained in the phase error image of the pyramid, and the filtered image of the peak still shows fringe structure damage in areas with very dense fringes. We will endeavor to overcome these issues in follow-up research.

**Objective evaluation:** Table IV shows the average MSE, SSIM, and Time of the six methods applied to the four images, and the performance is basically the same as presented in Section I. We verified the judgment of the subjective evaluation of the filter methods on the dependence of noise level by taking a sliding window (32 × 32) for the five filtered images, and calculated the MSE of the four simulated images. The change in the SSIM curve from top to bottom is shown in Fig. 16. The findings indicate that, except for the Pivoting median filter in the peak fringe dense areas where the MSE is greatly increased, and the SSIM is greatly decreased, the other four filtering methods can all obtain good filtering effects in the high-coherence areas (bottom). By contrast, in the low-coherence area (top), a clear difference is evident between the MSE and SSIM curves of the five filtering methods. The MSE curve of the MSFF-DCNN model always remains at the bottom with a change in coherence, especially for the cone and ramp, and the MSE changing curve always remains at a low level, whereas the curves of SSIM and MSFF-DCNN always maintain a high level, thereby confirming that the MSFF-DCNN model in the subjective evaluation has little dependence on noise level.

**Comprehensive analysis of Section I and II:** Compared to Pivoting median filtering, Lee filter, Goldstein filter, NL-InSAR filter, and the Ф-net, the proposed MSFF-DCNN model has the best ability for noise suppression and detail preservation in simulated images, and the least dependence on noise level.

**B. Real-Data Experiments**

Although the simulated interferometric phase and the real interferometric phase have similar noise and fringe distribution
characteristics, the full use of the simulated interferometric phase in place of the real interferometric phase is not sufficient for evaluating the performance of the proposed MSFF-DCNN model [34]. Therefore, in this section, we conduct filtering experiments using real interferometric phase images to evaluate the performance and generalization of the proposed method.

For this article, nine pieces of 256 × 256 interferometric phase with different terrain features and coherence were cropped from the interferometric processing results of sentinel-1A images (the processing software uses SARscape 5.3), located in Xinjiang and Gansu province. The selected items contain densely fringed interferometric phases generated from regions with larger slopes, sparsely fringed interferometric phases generated from regions with smaller slopes, and regions with nonuniform fringe density. The distribution of coherence is between 0.36 and 3.71, and the details are shown in Fig. 17. The interferometric phase fringes have been completely lost in the lower right urban area in $h$ and in the very densely vegetated area in $j$ (marked with red boxes in Fig. 17); therefore, those two areas are excluded from the evaluation.

**Subjective evaluation:** Similar to the performance in the simulated data, the Pivoting median filter in the interferometric phase images with higher average coherence shows a reduced resolution, a blurred fringe edge, and destruction of the phase structure. Increasing noise level [Fig. 18(e)–(i)] severely degrades the filtering performance. The Lee filter and Goldstein filter show good noise suppression and detail preservation in the high-coherence and sparse fringe areas, such as real data a, b, c, and d, but the noise suppression is weak and the detail preservation is insufficient in the low-coherence areas, such as real data e, f, g, h, and i. Compared with these three methods, NL-InSAR filtering generates a very smooth interferometric phase image, and the fringe edges are very sharp and clear. This is visually appealing but is, in fact, an overfiltering phenomenon termed the staircasing effect, which is very common in nonlocal filtering [14].

Both proposed model and $\Phi$-net show considerable visual effects. Compared with $\Phi$-net, the proposed model has better detail preservation ability during phase filtering. For example, in real data (e and f), the phase after $\Phi$-net filtering appears obvious and isolated residues, causing it to have a weaker fringe restoration. In real data (b and h), there is an unexpected yellow dot, we speculate that it is caused by the abnormal structure of the samples during the training process of the model.

**Objective evaluation:** Table V shows the average values for FRR, $Q$, and RPSD for the Lee filter, the Pivoting median filter, and the Goldstein filter in the nine real phase images. These three methods show significantly lower results than those obtained for NL-InSAR and MSFF-DCNN, we mainly compare the two well-performing methods: NL-InSAR and MSFF-DCNN. The NL-InSAR has 1407 fewer residues than the proposed MSFF-DCNN method, and the PRR is 11.3% higher, indicating that NL-InSAR has a very strong ability to suppress noise. However, $Q$ is higher for MSFF-DCNN than for NL-InSAR, and the RPSD is also higher for MSFF-DCNN than for NL-InSAR, indicating that the higher noise suppression of NL-InSAR comes at the cost of loss of detail in phase images. This noise preservation is not worth the loss in interferometric phase filtering. In addition, although the NL-InSAR algorithm is better than the Pivoting median filter and the Lee filter in terms of PRR and $Q$, this method is extremely time consuming, whereas the NL-InSAR takes 170.4 s and MSFF-DCNN model takes only 0.17 s.

Compared with the $\Phi$-net, among the objective evaluation indicators, the performance of the proposed model and $\Phi$-net are too close to make a comparison; therefore, we take the subjective visual evaluation as the main judgment basis, and at the same
time, we will also use the phase unwrapping experiments in the discussion section to further discuss the advantages of the proposed method over the $\Phi$-net.

In summary, in real-data filtering, the proposed MSFF-DCNN model has the best ability for noise suppression and detail preservation compared with Pivoting median filter, Lee filter, Goldstein filter, and NL-InSAR filter. The MSFF-DCNN model also has better generalization. The method proposed in this article meets the accuracy requirements of practical engineering applications.
TABLE V
OBJECTIVE EVALUATION OF REAL DATA FILTERING

| Methods       | NOR  | FRR (%) | Metric Q | RPSD  | T (s) |
|---------------|------|---------|----------|-------|-------|
| No filter     | 12451| 0       | 0.04     | 1.81  |       |
| Pivoting median| 7906 | 36.3    | 21       | 1.44  | 3.2   |
| Lee filter    | 4955 | 60.2    | 38       | 1.09  | 11.7  |
| Goldstein     | 4270 | 65.7    | 64       | 0.93  | 10.9  |
| NL-InSAR      | 1543 | 87.6    | 68       | 0.64  | 170.4 |
| Φ-net         | 2471 | 80.2    | 75       | 0.47  | 0.12  |
| Proposed method | 2950 | 76.3    | 78       | 0.45  | 0.17  |

As mentioned above, using the real and imaginary parts of the complex phase as the input and output achieves better results and the main reasons are as follows:

1) Reduce the difficulty of detecting noise information
In real field, phase noise and phase edges are moderately homogenous in both the spatial and frequency domains. In the spatial domain, both noise and phase edge information are isolated structures and have jump characteristics. In the frequency domain, both noise and phase edge information belong to high-frequency information. This homogeneity makes it easy for all filtering methods to “misjudgment” the phase edge as phase noise, bringing difficulty to phase filtering.

In contrast, in complex field, the phase noise of the real and imaginary parts still appears as jumping (spatial domain), high-frequency (frequency domain) information, but the phase edge appears as a continuous (spatial domain), low-frequency (frequency domain) information, and the difficulty of distinguishing phase edge and phase noise is significantly reduced. Therefore, phase filtering can have better edge-preserving ability in complex domain.

2) Requirements for the model loss function
For the current loss function, one component is to calculate MAE of each pixel, at the phase edge, due to the spatial value discontinuity in real field, the loss value of phase edges will be relatively large, which will affect the convergence of the model. Conversely, in a complex field, the real and imaginary parts of the complex phase are continuous, so the convergence and performance of the model are not affected.

V. DISCUSSION

A. Necessity of Transforming Phase
Converting the real phase with jumps into complex phase without jumps can enhance discrimination between noise and phase jumps, so that the model can better preserve edge information. In this section, the necessity and applicability of this transformation will be illustrated according to the results of comparative experiments.

Fig. 19 shows the filtering results of the model in different input and output. In Fig. 19(a), the phase structure is severely damaged, and there are also obvious abnormal structures at the top right of the filtered image, besides, using the phase directly as input and output significantly reduces the edge-preserving ability of the model, as shown in the enlarged image in the box, a large number of isolated residues appear on the phase edge, which will seriously affect the quality of phase unwrapping. On the contrary, in Fig. 19(b), the above problem is significantly improved.

![Fig. 19. Filtered results of different input and output ways: (a) real phase as input and output, and (b) complex phase as input and output.](image)

B. Gain Effect of SE-Block
To prove the gain effect of SE-block in the model, a simulated phase image and a real phase image are selected to conduct a control experiment with or without SE-block in the model. The filtering results are shown in the Fig. 20.

As shown in Fig. 20, in the regions of high-coherence, the performance difference of the models with SE-block and without SE-block is not obvious. But in the low-coherence regions, in box A and box B of the Fig. 20, the difference in the detail preservation ability of the two cases is shown, in box A, although the correct fringe structure is not completely restored in both cases, the fringes filtered by the model with SE-block are relatively less damaged. In Box B, the model with SE-block restored the ground truth almost completely. However, the phase
Fig. 20. Filtered simulated phase images and phase error images using models. (a) With SE-block. (b) Without SE-block.

Fig. 21. Filtered real data using models with and without SE-block. (a) With SE-block. (b) Without SE-block.

Table VI

| Objective Evaluation of Ablation Experiments (Phase Simulated from Peak) |
|-----------------------------|-----------------------------|
|                            | Without SE-block | With SE-block |
| MSE                        | 0.43            | 0.37          |
| SSIM                       | 0.88            | 0.92          |

filtered by the model without SE-block shows abnormal isolated structures and damage of fringe structures. In addition, the model with SE-block has better resolution preservation in regions of low-coherence.

In Fig. 21, the model with SE-block has a stronger ability to maintain the fringe structure, as in the black circle in the Fig. 21, the model with SE is trying its best to restore a complete fringe, while without SE-block, these regions show significant structural damage that is very prone to serious errors in phase unwrapping.

Furthermore, in Table VI, the model with SE-block gained lower MSE and higher SSIM, and in Table VII, the model with SE-block gained lower RPSD and higher $Q$. In summary, SE-block has obvious gain effect on the performance of the model.

C. Effectiveness of Synthesizing Phase Samples Using Simulated Terrain

In the deep learning phase filtering research, phase samples synthesized from simulation and real terrain are the two main sample synthesis methods [25], [34]. In the previous experiments, we have tried these two methods. To demonstrate the effectiveness of synthesizing samples using simulated terrain in this article, we design three sample making schemes to evaluate the influence of sample settings on model performance.

Scheme (1): All samples are synthesized by SRTM DEM (30 m), the mosaic image of SRTM is shown in Fig. 22.
Fig. 23. Filtered simulated phase under the condition of different sample simulation schemes.

Fig. 24. Filtered real data under the condition of different sample simulation schemes.

Fig. 25. Clean phase and its unwrapped results of simulated phase (peak).

Scheme ①: The samples use simulated terrain and SRTM DEM at a 1:1 scale.
Scheme ③: All samples synthesized from simulated terrain.

Fig. 23 shows the experimental results of the three schemes on simulated data. Phase fringes in low-coherence regions are severely damaged, a large number of abnormal structures appear in the upper right corner of the image, and the resolution of the image is also reduced, as the number of simulated terrain data in the sample increases, from scheme ② to ③, these errors gradually diminish.

Fig. 24 shows the experimental results of the three schemes on real data (e). The most significant features of the scheme ① and scheme ② are that the residues suppression is insufficient, and the preservation of phase fringes is relatively weak, the obvious residues in scheme ① and scheme ② will bring great challenges to the phase unwrapping. In Table VIII, the result from scheme ③ gained lower MSE and higher SSIM, in Table IX, the result from Scheme ③ gained lower RPSD and higher Q.

Overall, training samples synthesized from simulated terrain are more applicable in this article, and we speculate that the main reasons are as follows:

1) Simulated terrain makes it easier to control the diversity and tendency of samples

From the literature [56], [57], [58], we know that the terrain simulated by the diamond-square method is too realistic to synthesize interferometric phase. In addition, in the sample production, a large number of samples of conventional terrain are used in the sample, such as mountains with small slopes, plains, etc., and a small number of samples of extreme conditions, such as high-slope terrain, cliffs, etc., are more conducive to the fitting and performance of the model. The setting of the tendency and diversity of such samples is easy to achieve by using the random parameter setting of the diamond-square method. However, when using SRTM DEM for sample simulation, due to the spatial correlation, the sample characteristics and distributions tend to have high similarity, which is not conducive to the generalization of the model, although this situation can be changed, but needless to say, it required a lot of manual intervention. Therefore, the phase simulation based on the diamond-square method can control the tendency and only requires little manual intervention.

2) The error of SRTM DEM affects the sample quality

The SRTM DEM is also derived from the InSAR results. Therefore, the quality of the DEM is also affected by factors, such as radar shadowing, phase unwrapping, echo lag, etc., resulting in data holes and abnormal structures in the SRTM DEM, which affect the fitting and performance of the model unless manually screened and corrected.

D. Evaluating Filter Quality Using Phase Unwrapping Results

The quality of the filtering results directly affects the quality of phase unwrapping. In this article, the simulated data peak, real data (e) and (h) are unwrapped by Goldstein’s branch cut method, and the unwrapping results are used to evaluate the effectiveness of phase filtering. According to the visual effect of the unwrapped phase images, such as the continuity of the absolute phase and the degree of visual correlation with the real terrain, we can clearly judge the quality of phase unwrapping.
In this article, objective evaluation indicators, such as MSE and the proportion of unwrapped area (PUA), were used to evaluate the unwrapping results of the simulated data. The real data will not be evaluated using objective evaluation indicators because in the absence of reliable terrain data, it is more meaningful to evaluate the unwrapping results visually.

Fig. 25 is the unwrapping result of the clean simulated data, which is used as a reference for the quality evaluation of the phase unwrapping of the simulated data. Fig. 26 is the result of phase unwrapping of filtered images using different methods. Compared with the previous four traditional methods, the simulated phase gets better results after being filtered by Φ-net and the proposed method, the unwrapping results show only a small area of unwrapping errors in the regions where fringes are very dense. Further, compared with Φ-net, the unwrapping result of the proposed method at top of image is smoother, and is closer to the unwrapped result of the clean phase. At the same time, in the objective evaluation (Table X), the proposed model got smaller MSE and larger PUA.

![Fig. 26](image1)

**TABLE X**

| Methods         | MSE  | PUA (%) |
|-----------------|------|---------|
| Pivoting Median | 1.77 | 87.5    |
| Lee             | 0.37 | 95.2    |
| Goldstein       | 0.56 | 92.1    |
| NL-InSAR        | 0.51 | 93.8    |
| Φ-net           | 0.21 | 97.6    |
| Proposed model  | 0.16 | 98.2    |

Fig. 27 shows the terrain conditions of the real data (e) and (h), and Figs. 28 and 29 show the unwrapping results of the two terrain conditions. With real terrain conditions as reference, the unwrapped result of real data (e) should be smooth and continuous, and relative to the actual terrain in phase values. Compared with the proposed method, the previous five methods have a large area of unwrapping errors at the bottom of the
image in Fig. 28, which is represented by obvious faults in the unwrapped phase, and the result of \( \Phi \)-net is no exception. On the contrary, the unwrapped phase from the proposed model is more continuous and visually reliable. The unwrapping of the real data (h) is challenging, in Fig. 29, due to severe decoherence in the lower right corner of the image, the unwrapping results are no longer reliable; therefore, only the unwrapping results of the mining settlement funnel (in the circle in Fig. 29) are discussed; in this case, only filtered phase from NL-InSAR and the proposed model have a large unwrapped area at the pit. However, in combination with [2] and [59] and the optical image, the unwrapping result from the proposed method has a
more reasonable shape at the pit and is closer to a complete funnel-shaped deformation area.

In general, the proposed method achieves more reliable results in both simulated and real data unwrapping, which proves the effectiveness of the proposed method in InSAR applications.

E. Existing Problems and Future Research

The main goal of this article is to establish a multiscale feature dynamic fusion network based on DCNN to achieve efficient and highly accurate interferometric phase filtering. Compared with four traditional methods, the proposed MSFF-DCNN model has high performance and efficiency with both simulated phase image and real phase image. The proposed model can, without a doubt, provide support for InSAR high-precision data processing. We also avoided the problem of overfitting by reducing the simulation of strong noise and corner points of interferometric phases during the production of the dataset. Nevertheless, because of this preservation, the phase residual of corner-point and phase edge cannot be completely removed.

We speculate that the attention mechanisms targeting phase edges may address the above problems; therefore, in future article, we will embed an attention mechanism for interferometric phase edge information into the network, so that the performance of the phase filtering method could be improved again.

In addition, the problem of fake fringes and over smoothing under extreme conditions has always been a difficult problem for deep-learning-based phase filtering methods, and this article is no exception. We speculate that this is due to the fact that the training samples produced did not have sufficient diversity and reasonable tendency; therefore, in the follow-up work, the simulation study of the sample will be our research focus.

VI. Conclusion

In this article, an interferometric phase filtering method based on DCNNs is proposed and confirmed to have better filtering performance and higher computational efficiency than the currently widely used phase filtering methods. The article first uses the diamond-square algorithm to simulate the interferometric phase and transforms it into the complex domain to provide the dataset required for model training, and uses the real and imaginary parts as both the input and output of the network to avoid judging the phase edge as noise. The training is unstable, and the filtering effect is poor. The SE-block is then embedded in the model to construct a convolutional neural network model with dynamic fusion of multiscale features, so that the network can finally take into account the deep and shallow features of multiscale semantics when predicting the clean interferometric phase to enhance noise suppression and guarantee detail preservation. Finally, the performance of the proposed method is evaluated using subjective evaluation and objective evaluation.

In the simulation experiment, the interferometric phase filtering experiment with uniform noise intensity distribution shows that, compared with the commonly used interferometric phase filtering methods, the proposed MSFF-DCNN model shows better filtering performance in both dense and sparse fringe distribution. The experiments of high-density interferometric phase filtering with nonuniform intensity show that, compared with the commonly used interferometric phase filtering methods, the proposed MSFF-DCNN method has less dependence on the noise intensity while ensuring noise suppression and detail preservation. In the real data experiments, where real interferometric phase images with different distribution characteristics are selected for phase filtering, the results show that the proposed method has high filtering performance and strong generalization ability.

References

[1] W. Zhang et al., “Improved DEM reconstruction method based on multibaseline InSAR,” IEEE Geosci. Remote Sens. Lett., vol. 19, 2022, Art. no. 4011505, doi: 10.1109/LGRS.2021.3069239.
[2] W. Wang, Y. He, L. Zhang, Y. Chen, L. Qiu, and H. Pu, “Analysis of surface deformation and driving forces in Lanzhou,” Open Geosci., vol. 12, no. 7, pp. 1274–1145, 2020.
[3] W. Yang, Y. He, W. H. Wang, Y. D. Chen, and Y. Chen, “InSAR monitoring of 3-D surface deformation in Jinchuan mining area, Gansu province,” Remote Sens. Nat. Resour., vol. 34, no. 1, pp. 177–188, 2022.
[4] Y. He, Y. Chen, W. Wang, H. Yan, L. Zhang, and T. Liu, “TS-InSAR analysis for monitoring ground deformation in Lanzhou new district, the loess Plateau of China, from 2017 to 2019,” Adv. Space Res., vol. 67, pp. 1267–1283, 2021.
[5] M. Khaki, M. S. Filmer, W. E. Featherstone, M. Kuhn, L. K. Bui, and A. L. Parker, “A sequential Monte Carlo framework for noise filtering in InSAR time series,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 8, pp. 1904–1912, Mar. 2020.
[6] H. Zink, Zebker, and J. Villasenor, “Decorrelation in interferometric radar echoes,” IEEE Trans. Geosci. Remote Sens., vol. 30, no. 5, pp. 950–959, Sep. 1992.
[7] H. Bahr and R. F. Hansen, “Reliable estimation of orbit errors in spaceborne SAR interferometry,” J. Geodesy, vol. 86, pp. 1147–1164, 2012.
[8] Y. Gao, S. Zhang, T. Li, L. Guo, Q. Chen, and S. Li, “A novel two-step noise preservation approach for interferometric phase images,” Opt. Lasers Eng., vol. 121, pp. 1–10, 2020.
[9] Y. Wang, Study on High- Efficiency and High-Precision Filtering Methods for Synthetic Aperture Radar Interferometric Phase Images. Changsha, China: Nat. Univ. Defense Technol., 2016.
[10] R. Lanari et al., “Generation of digital elevation models by using SIR-C/JX-SAR multifrequency two-pass interferometry: The Etna case study,” IEEE Trans. Geosci. Remote Sens., vol. 34, no. 5, pp. 1097–1114, Sep. 1996.
[11] J. S. Lee, K. W. Hoppell, S. A. Mango, and A. R. Miller, “Intensity and phase statistics of multilook polarimetric and interferometric SAR imagery,” IEEE Trans. Geosci. Remote Sens., vol. 32, no. 5, pp. 1017–1028, Sep. 1994.
[12] J. S. Lee, K. P. Papathanassiou, T. L. Ainsworth, M. R. Grunes, and A. Reiger, “A new technique for noise filtering of SAR interferometric phase images,” IEEE Trans. Geosci. Remote Sens., vol. 36, no. 5, pp. 1456–1465, Sep. 1998.
[13] C. A. Deledalle, L. Denis, and F. Tupin, “NL-InSAR: Nonlocal interferogram estimation,” IEEE Trans. Geosci. Remote Sens., vol. 49, no. 4, pp. 1441–1452, Apr. 2011.
[14] F. Sica, T. Ciofini, X. Y. Zhu, L. Verdoliva, and G. Poggi, “InSAR-BM3D: A nonlocal filter for SAR interferometric phase restoration,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 6, pp. 3456–3467, Jun. 2018.
[15] G. Xu, Y. Gao, J. Li, and M. Xing, “InSAR phase denoising: A review of current technologies and future directions,” IEEE Geosci. Remote Sens. Mag., vol. 8, no. 2, pp. 64–82, Jun. 2020.
[16] G. Baier, X. X. Zhu, M. Lachaise, H. Breit, and R. Bamler, “Nonlocal InSAR filtering for DEM generation and addressing the staircasing effect,” in Proc. 11th Eur. Conf. Synthetic Aperture Radar, 2016, pp. 1–4.
[17] R. M. Goldstein and C. L. Werner, “Radar interferogram filtering for geophysical applications,” Geophys. Res. Lett., vol. 25, pp. 4035–4038, 1998.
[18] E. Trouve, J. M. Nicolas, and H. Maitre, “Improving phase unwrapping techniques by the use of local frequency estimates,” IEEE Trans. Geosci. Remote Sens., vol. 36, no. 6, pp. 1963–1972, Nov. 1998.
[19] Y. Rao, W. Zhao, Z. Zhu, J. Lu, and J. Zhou, “Global filter networks for image classification,” Adv. Neural Inf. Process. Syst., vol. 34, pp. 980–993, 2021.
M. Jiang, X. Din, X. Tian, R. Malhotra, and W. Kong, “A hybrid method for optimization of the adaptive Goldstein filter,” ISPRS J. Photogramm. Remote Sens., vol. 98, pp. 29–43, 2014.

Y. Bian and B. Mercier, “Interferometric SAR phase filtering in the wavelet domain using simultaneous detection and estimation,” IEEE Trans. Geosci. Remote Sens., vol. 49, no. 4, pp. 1396–1416, Apr. 2011.

M. Jiang et al., “The improvement for baran phase filter derived from unbiased InSAR coherence,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 7, no. 7, pp. 3002–3100, Jul. 2014.

J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, “Stacked convolutional auto-encoders for hierarchical feature extraction,” in Proc. Int. Conf. Artif. Neural. Netw., 2011, pp. 52–59.

Y. He et al., “A unified network of information considering superimposed landslide factors sequence and pixel spatial neighbourhood for landslide susceptibility mapping,” Int. J. Appl. Earth Observ. Geoinf., vol. 104, 2021, Art. no. 102508.

S. Li, H. Xu, S. Gao, W. Liu, C. Li, and A. Liu, “An interferometric phase noise reduction method based on modified denoising convolutional neural network,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 13, pp. 4947–4959, Aug. 2020, doi: 10.1109/JSTARS.2020.3017808.

K. Zhang, W. Zao, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” IEEE Trans. Image Process., vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

F. Sica, G. Gobbi, P. Rizzoli, and L. Bruzzone, “9-net: Deep residual learning for InSAR parameters estimation,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 5, pp. 3917–3941, May 2021.

A. G. Roy, N. Navab, and C. Wachinger, “Concurrent spatial and channel ‘squeezing excitation’ in fully convolutional networks,” in Proc. Int. Conf. Med. Image Comput. Assist. Intervention, 2018, pp. 421–429.

Y. He et al., “An extraction method for glacial lakes based on Landsat-8 imagery using an improved U-Net network,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 14, pp. 6544–6556, Jun. 2021, doi: 10.1109/JSTARS.2021.3085397.

J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 7132–7141.

S. Mukherjee, A. Zimmer, X. Sun, P. Ghanam, and L. Cheng, “An unsupervised generative neural approach for InSAR phase filtering and coherence estimation,” IEEE Geosci. Remote Sens. Lett., vol. 18, no. 11, pp. 1971–1975, Nov. 2021.

H. J. Yin and G. Y. Zhang, “InSAR interferogram and phase noise simulation,” Geotech. Earthq. Eng., vol. 45, no. 9, pp. 58–61, 2017.

J. M. Willemse and K. A. Hawick, “Generation and rendering of fractal terrains on approximated spherical surfaces,” in Proc. 17th Int. Conf. Comput. Graph. Virtual Reality, 2013, Art. no. CGV4061.

L. Pu, X. Zhang, Z. Zhou, J. Shi, S.Wei, and Y. Zhou, “A phase filtering method with scale recurrent networks for InSAR,” Remote Sens., vol. 12, no. 20, 2020, Art. no. 3453.

L. Pu, X. Zhang, Z. Zhou, L. Li, J. Shi, and S. Wei, “Nonlocal feature selection encoder-decoder network for accurate InSAR phase filtering,” Remote Sens., vol. 14, no. 5, 2022, Art. no. 1174.

F. Rocca, F. A. Ferretti, A. V. Monti-Guarnieri, C. M. Prati, and D. Massommet, InSAR Processing: A Mathematical Approach. Noordwijk, The Netherlands: European Space Agency, 2007, pp. 29–30.

K. Yan, Y. Yu, T. Sun, A. Asundi, and Q. Kemao, “Wrapped phase de-noising using convolutional neural networks,” Opt. Lasers Eng., vol. 128, 2020, Art. no. 105999.

A. B. Sukmonon and A. Hirose, “Adaptive noise preservation of InSAR images based on a complex-valued MRF model and its application to phase unwrapping problem,” IEEE Trans. Geosci. Remote Sens., vol. 40, no. 3, pp. 699–709, Mar. 2002.

C. Werner, U. Wegmüller, T. Strozzi, and A. Wiesmann, “Processing strategies for phase unwrapping for INSAR applications,” in Proc. Eur. Conf. Synthetic Aperture Radar, 2002, pp. 353–356.

L. Wang, J. Peng, and W. Sun, “Spatial–spectral squeeze-and-excitation residual network for hyperspectral image classification,” Remote Sens., vol. 11, no. 7, 2019, Art. no. 884.

Z. Lin, K. Ji, X. Leng, and G. Kuang, “Squeeze and excitation rank faster R-CNN for ship detection in SAR images,” IEEE Geosci. Remot. Sens. Lett., vol. 16, no. 5, pp. 751–755, May 2019.

S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in Proc. Int. Conf. Mach. Learn., 2015, pp. 448–456.
Lifeng Zhang received the M.S. degree in cartography and geographic information system from the Lanzhou Jiaotong University, Lanzhou, China, in 2010, and the Ph.D. degree in environmental science and engineering from the Lanzhou Jiaotong University, Lanzhou, China, in 2017. He is currently an Associate Professor with the Lanzhou Jiaotong University, Lanzhou, China. His research interests include ecological environment remote sensing monitoring, visualized analysis, and analysis of land use change.

Sheng Yao (Student Member, IEEE) received the B.E. degree in digital media technology from the North China University of Technology, Beijing, China, in 2020. He is currently working toward the M.S. degree in surveying and mapping with the Faculty of Geomatics, Lanzhou Jiaotong University, Lanzhou, China. His research interests include application of remote sensing in deep learning.

Zhiqing Wen received the B.E. degree in surveying and mapping engineering from the Chang’an University, Xian, China, in 2020. He is currently working toward the M.S. degree in resources and environment with the Faculty of Geomatics, Lanzhou Jiaotong University, Lanzhou, China. His research interests include remote sensing image processing and remote sensing application.

Shengpeng Cao received the B.E. degree in remote sensing science and technology from the Lanzhou Jiaotong University, Lanzhou, China, in 2018. He is currently working toward the M.S. degree in surveying and mapping with the Faculty of Geomatics, Lanzhou Jiaotong University, Lanzhou, China. His research interests include drought remote sensing monitoring and predicting.

Zhanao Zhao received the B.E. degree in surveying and mapping engineering from the Chengdu University of Technology, Chengdu, China, in 2020. He is currently working toward the M.S. degree in surveying and mapping with the Faculty of Geomatics, Lanzhou Jiaotong University, Lanzhou, China. His research interests include application of remote sensing in deep learning.

Yi Chen received the B.E. degree in remote sensing science and technology from the Lanzhou Jiaotong University, Lanzhou, China, in 2018, and the M.S. degree in surveying and mapping from the Lanzhou Jiaotong University, Lanzhou, China, in 2022. His research interests include land subsidence monitoring and prediction, and deep learning.

Yali Zhang received the B.E. degree in remote sensing science and technology from the Lanzhou Jiaotong University, Lanzhou, China, in 2020. She is currently working toward the M.S. degree in surveying and mapping with the Faculty of Geomatics, Lanzhou Jiaotong University, Lanzhou, China. Her research interests include InSAR data processing technology, extraction of glacier movement velocity, and remote sensing image fusion.