

METHODS

Deep Learning Improves Point Density in PS-InSAR Data Toward Finer-Scale Land Surface Displacement Detection

ANASTASIIA SAFONOVA¹ AND **MASAHIRO RYO^{1,2}**¹Leibniz Centre for Agricultural Landscape Research (ZALF), 15374 Müncheberg, Germany²Environment and Natural Sciences, Brandenburg University of Technology Cottbus-Senftenberg, 03046 Cottbus, Germany

Corresponding author: Anastasiia Safonova (Safonova.nastya1@gmail.com)

This work was supported by the Federal Ministry of Education and Research (BMBF–Bundesministerium für Bildung und Forschung) through the project “Multi-Modale Datenintegration, Domänenspezifische Methoden und KI zur Stärkung der Datenkompetenz in der Agrarforschung” under Grant 16DKWN089.

ABSTRACT The permanent scatterer interferometric aperture radar (PS-InSAR) technique is used to measure and monitor displacements of the Earth’s surface over time. While the approach is promising for large-scale deformation, the density of the received PS points is insufficient for localized deformation analysis. In this first work, we aim to improve the technique by increasing the point density of high-precision deformation monitoring in PS-InSAR data by developing a convolutional long short-term memory (ConvLSTM) model that predicts PS points on different land covers, such as forest, urban, natural, water, and combinations among them. The proposed architecture, PS-ConvLSTM, was trained on a temporary dataset with interferograms to classify stable and unstable PS pixels from over 200,000 site images obtained from the city of Barcelona, Spain. The result showed that the trained PS-ConvLSTM model is highly compatible with the method currently used, which requires a large manual effort by an expert (accuracy: 99%). In addition, the proposed approach increased the point density by 15%, indicating that ConvLSTM is a promising approach for increasing the point density in PS-InSAR data and thus improving localized deformation analysis.

INDEX TERMS Deep learning, recurrent neural network, long short-term memory, point density, persistent scatter, InSAR process.

I. INTRODUCTION

Insufficient point density of high-precision deformation monitoring in Interferometric Synthetic Aperture Radar (InSAR) data is one of the current research problems in Earth Science [1]. Solving this problem can be done using one of the most powerful Remote Sensing (RS) methods called Persistent Scatterer Interferometry (PSI). Thanks to this method, it is possible to measure millimeter displacements of the earth’s surface. Looking at the chronology of how PSI was developed [2], the method was first introduced by Ferretti et al., in 2000 and [3], [4]. Furthermore, the Small Basic Subset Method (SBAS) has been introduced in several papers: Berardino et al. [5], Mora et al. [6], Schmidt and

Bürgmann [7], Duro et al. [8], Crosetto et al. [9], [10], López-Quiroz et al. [11], Goel and Adam [12], and Devanthéry et al. [13]. Similar studies can also be attributed to the following series of works with a basic configuration - one primary: Werner et al. [14], Kampes [15], Hooper et al. [16], Costantini et al. [17], Ferretti et al. [18], Van Leijen [19], and Lv et al. [20].

There are hundreds of studies in the literature that present this method, but it still has several drawbacks. We noticed that since the first papers by Ferretti et al. in 2000 [3], researchers in InSAR have not used deep learning (DL) until 2019. One of the main reasons for this was unfamiliarity with these fields. In particular, based on the presented literature, the previously method is not fully automated and still requires the participation of an expert [21]. Another feature is that it does not allow obtaining the maximum point density map,

The associate editor coordinating the review of this manuscript and approving it for publication was Angel F. García-Fernández ^{ID}.

especially on scenes with natural objects. In addition, the method is computationally expensive and requires a significant amount of time to obtain the result, depending on the field of study. However, we highlight the following studies where DL has been used in individual steps of the InSAR process: scatterer acceleration approach [22], homogeneous pixel selection [23], and coherent pixel selection [24]. Further research is needed as the overall accuracy remains above 90% despite the increased data processing speed. In addition, the algorithms are not universal and require additional training on a diverse set of real-world data.

In the last decade, DL methods have gained immense popularity to solve various types of problems in different fields. Convolutional Neural Network (CNN) [25], [26], Recurrent Neural Network (RNN) [27], and Long Short-Term Memory (LSTM) [28] are widely used. CNNs are most commonly used for making predictions but not always suitable for time-series prediction [29]. It depends on the time series data to which CNN is applied. In real-time series datasets, many features are temporally correlated, i.e. they occur one after another in a certain order, and this information can be lost depending on the size and type of pooling operations used. RNNs are designed to handle complex sequence dependencies in time series analysis [30], [31]. However, a traditional RNN is not capable of predicting a long time series due to the problem of lack of long-term memory, that is, the vanishing gradient problem [32]. As the gradients disappear, the weights of the earlier layers in the network may not be updated significantly, causing these layers to learn very slowly or not at all. This can result in the network failing to capture long-term dependencies in the sequential data as it struggles to propagate relevant information over many time steps. To solve this problem, the LSTM model was developed [33]. Although, for the task of predicting any object on the data sequence (in our case, PS point and non-PS point), it is necessary to capture the spatial dependency including composition and configuration. For this purpose, a convolution layer was built into the LSTM (ConvLSTM) [34]. To perform individual steps of the InSAR process, the LSTM algorithm was used to detect ground displacements from InSAR products, which are deformation rate (velocity map) and deformation time series [35], [36]. Tiwari et al. [37] demonstrated the promising potential of ConvLSTM for selecting measurement pixels in multi-temporal InSAR. However, the performance of ConvLSTM for increasing the point density of InSAR has not been investigated.

The main contributions of this paper can be listed as follows:

- This is the first paper to present the potential of ConvLSTM for increasing point density in PS-InSAR data (hereafter, namely, PS-ConvLSTM). We demonstrate that PS-ConvLSTM increases the point density by 15%, enhancing the detectability of finer-scale land displacements with InSAR.
- We built a new labeled dataset of temporal interferometric with two classes as PS point and non-PS point.

- We provide a complete description of the network used with the open-source code for wider applications. Our source code is available at https://github.com/ansafo/AI_InSAR.

The rest of the paper is organized as follows. The materials are provided in Section II, which includes the study area and the InSAR Imagery and dataset construction. The methods are presented in Section III, which includes description of InSAR processing, the proposed PS-ConvLSTM architecture, experimental setup, and metrics for performance evaluation of the PS-ConvLSTM architecture. Section IV presents the experimental results of increasing point density by PS-ConvLSTM architecture, including training, validation, and testing results. The discussion and the conclusions are presented in Section V.

II. MATERIALS

A. STUDY AREA AND InSAR IMAGES

The study area is located in Barcelona, Catalonia, Spain with the Global Positioning System coordinates of $41^{\circ}23'24.7380''$ N and $2^{\circ}9'14.4252''$ E (Figure 1).



FIGURE 1. Test area in Barcelona, Catalonia, Spain.

The climate is Mediterranean with mild winters, hot summers, and rainy fall and spring [38]. The average annual temperature in the city is 21.2°C during the day and 15.1°C at night. The average annual relative humidity is 72%, and the average annual rainfall is 614 mm. This part of Spain is located on the northeastern coast of the Iberian Peninsula, on a plain about 5 km wide, bounded by the Collserola mountain range, the Llobregat rivers in the southwest and the Besòs in the north. The total area of the plain is 170 km², of which 101 km² is urban. The territory of Barcelona is varied and rich and includes such terrains as mountains, forests, urban environments, water, natural and their combinations. Therefore, the whole territory of Barcelona was chosen as a test area. For the experiment we used a set of InSAR images in

the form of a rectangle dimension 5600 pixels by 1600 pixels with a resolution of 10 meters per pixel.

B. DATASET CONSTRUCTION

To use the PS-InSAR technique in theory, 10 temporary images may be sufficient to highlight stable points and build a PS mask. In practice, this may not be enough, especially in natural landscapes such as forests and water, due to the lack of stable points on them. However, in our study, the InSAR dataset consists of 240 time series of satellite interferometry images obtained from Sentinel-1 radar images with 4×14 m spatial resolution as Single Look Complex products, which were used as inputs for the NN. The Single Look Complex images are in the oblique range along the azimuth image plane, in the satellite data acquisition image plane. Each pixel in the image is represented by a complex (I and Q) amplitude value with image resolution 4×14 m. It contains both amplitude and phase information. Each I and Q value is 16 bits per pixel. The images are georeferenced using satellite orbit and position data. The image set was downloaded from the Alaska Satellite Facility website [39] with dates ranging from January 1, 2019 to August 2022. Interferograms have short temporal baselines, less than 18 days.

Since each interferometric image was large (5600 by 1600 pixels), it was therefore decided to divide the entire dataset into patches of 100 by 100 pixel resolution. The patches are binary raster with a byte value between 0 and 255. This step made it possible to train the NN more efficiently and to speed up the computations depending on the available computing power. Also, because of this step, all patches were classified

and divided into 9 areas, including forest, urban, natural, water, forest and urban, urban and natural, natural and water, urban and water, and forest and natural. This allowed us to see the percentage of content represented in the dataset. Since the test area is not artificially prepared for the experiment, there is an imbalance of classes. The percentages of the classes are shown in Table 1.

TABLE 1. Proportions of dataset areas.

No.	Class	Patch	%
1	Forest	227	16.27
2	Urban	124	14.35
3	Natural	13	1.5
4	Water	14	1.62
5	Forest & Urban	233	26.97
6	Urban & Natural	235	27.2
7	Natural & Water	8	0.93
8	Urban & Water	1	0.12
9	Forest & Natural	9	1.04
Total		864	100

Thus, a dataset representing the diversity of land cover was created. Therefore, 864 patches per interferogram were prepared, resulting in 207,360 patches in the dataset. Based on the documentation provided for the TensorFlow library for the following function ConvLSTM3D, in our case the input

layer was prepared as a 5D tensor with shape: (samples, time, channels, rows, cols, depth).

As a ground truth (output) for the NN, a PS mask was used as a binary raster with dimensions 1600 by 5400 pixels and with values 1 (PS point) and 0 (non-PS point). PS point is a stable behavior that doesn't have significant movements and has an average deformation of less than ± 2 mm per year. Like the entire set of images, the mask was divided into patches of 100 by 100 pixels. The dataset and mask with one patch example are shown in Figure 2.

III. METHODS

A. InSAR PREPROCESSING

PSI is a radar-based technique within the field of differential interferometric SAR and is proving to be a powerful RS method for the measuring and monitoring of surface displacements over time [2]. Typically, pixels are selected by considering the perceived noise in the spatially uncorrelated phase component along with the angle of view error in the temporal interferometric stack. Various PS-InSAR approaches have been proposed in the last thirty years, where four of the most important approaches in terms of impact on the PSI field are the PSInSARTM technique proposed by Ferretti et al., in 2000 and 2001 [3], [4], the Small Baseline Subset (SBAS) technique [5], the PSI contribution was given by Hooper et al. in 2004 [16] with their new InSAR PS method, and the SqueeSARTM algorithm proposed by Ferretti et al. in 2011 [18]. The main flowchart of the algorithm is shown in Figure 3 [13].

The method includes 7 steps: interferograms generation, linear velocity estimation, linear velocity residuals, time series without linear component, atmosphere estimation and removal from the time series, reintroduction of the linear component into the time series, and geocoding.

The first step includes image loading, burst and swath extraction, co-registration, and different interferograms and coherences generation. The second step performs dispersion of amplitude selection, PS linear velocity estimation and Gamma-based PS selection. The third step is the removal of the linear velocity component from all the interferograms and the construction of residual interferograms. The fourth step includes 2D phase unwrapping and 1D phase unwrapping (time series estimation). The fifth and sixth steps include atmosphere estimation and removal from the time series and reintroduction of the linear component into the time series, respectively. A map of the deformation activity is created at the final stage - geocoding. The most detailed description of the method developed at the Centre Tecnologic de Telecomunicacions de Catalunya (CTTC) was presented by Devanthery et al. [13], [21].

InSAR techniques include several processing steps to obtain the final outputs, including SAR data quality assessment, image co-registration, phase unwrapping errors, topographic and atmospheric corrections, etc. Although almost all these steps are performed automatically, they still need to be checked step by step. For instance, experts should assess the quality of the raw SAR data. They need to identify potential

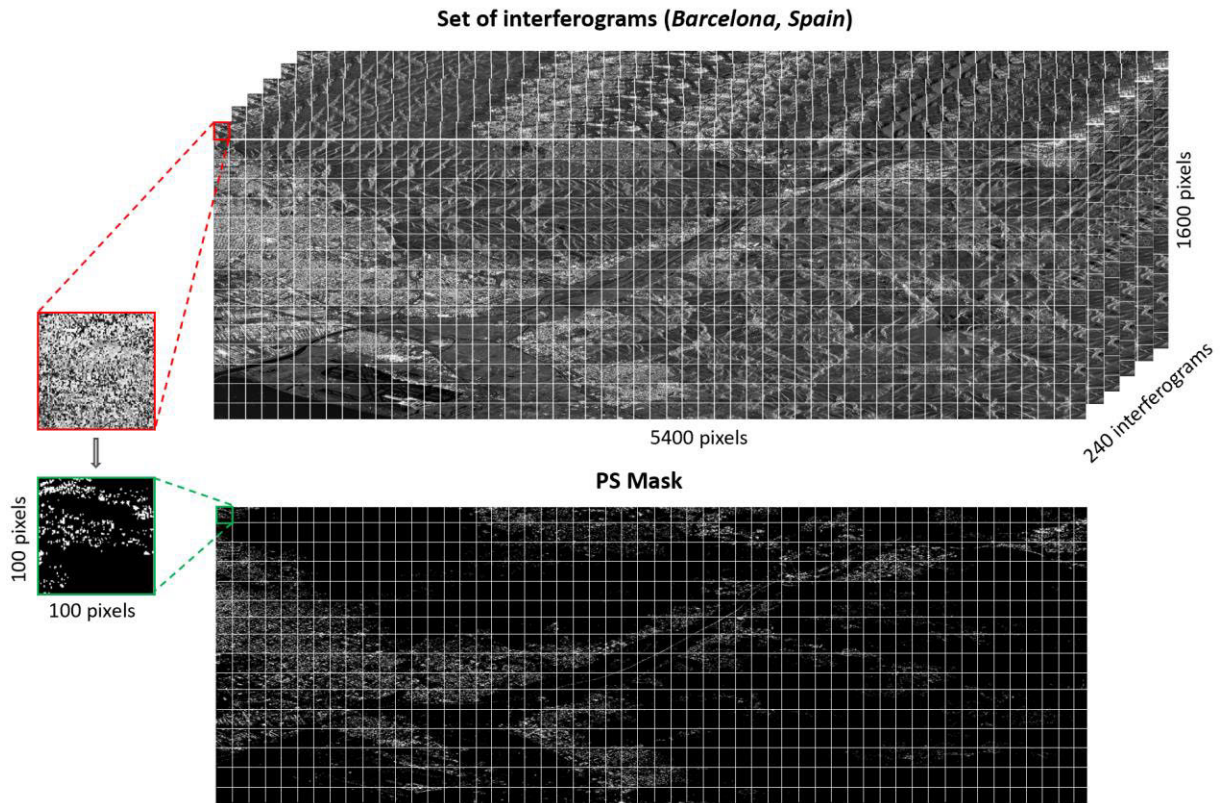


FIGURE 2. Test area in Barcelona, Catalonia, Spain.

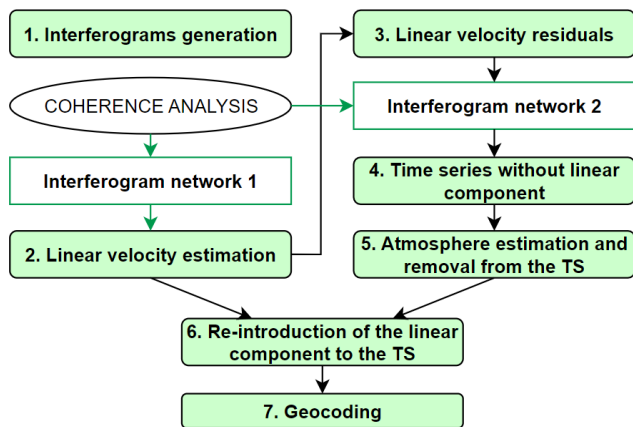


FIGURE 3. Flowchart of conventional InSAR processing.

issues such as atmospheric artifacts, orbital errors, and other sources of noise that could affect the accuracy of the deformation measurements. Also, the co-registration of primary interferogram must be visually monitored by an expert to check whether the corresponding pixels of the images and interferograms are correctly aligned to the same geographical location [2].

After the acquisition of SAR images, the next processing steps are performed by processing tools. The final output

is then reviewed by InSAR experts to assess the validity of the results. For example, an active construction site cannot be without movement during a time series of InSAR measurements. Therefore, if InSAR measurements show that an active construction site did not move during the acquired SAR images, the InSAR outputs have not been processed correctly. Consequently, InSAR experts start to check the processing step by considering the errors of the final outputs. However, almost all InSAR processing techniques can be performed at a lower level (checking the output of each processing step). Thus, an expert can check the correctness of each step. In the maximum point density step, the strong backscattered signals are mostly received from natural and artificial objects that are proper reflectors, such as rocks and urban areas. Thus, in study areas without proper reflectors, the number of measurement points will decrease, which may affect the quality of the outputs.

B. PS-ConvLSTM ARCHITECTURE

In order to improve the PSI method (step 2 in Figure 3) developed by CTTC [13], [21], it was decided to introduce DL technologies as the most promising technology [25], [40] to partially increase the point density of high-precision deformation monitoring. The PS-ConvLSTM architecture was developed and trained on the prepared images (see Section II-B) to increase the point density on stable and unstable PS pixels. A number of experiments were conducted to

achieve a high performance. To create the NN and improve the quality of its training, we manually tuned the hyperparameters in the network. To evaluate the quality of the developed model in each experiment, a different number of layers, network hyperparameters, training time, percentage split of the dataset into test and validation sets, optimization, loss function and evaluation metric were chosen.

The architecture of a NN has sequential layers. Since the input data are vectors and can be grouped into a 5D array in time, it was decided to use the ConvLSTM2D layer (1). This layer can inject a tensor with a shape: (samples, time, rows, cols, and channels) and the output shape: (samples, new rows, new cols, and filters) [34].

$$\begin{aligned} i_t &= \sigma(W_{xi} * x_t + W_{hi} * h_{t-1} + W_{ci}^\circ c_{t-1} + b_i) \\ f_t &= \left(W_{xf} * x_t + W_{hf} * h_{t-1} + W_{cf}^\circ c_{t-1} + b_f\right) \\ c_t &= f_t^\circ c_{t-1} + i_t^\circ \tanh(W_{xc} * x_t + W_{hc} * h_{t-1} + b_c) \\ o_t &= \sigma(W_{xo} * x_t + W_{ho} * h_{t-1} + W_{co}^\circ c_t + b_o) \\ h_t &= o_t^\circ \tanh(c_t), \end{aligned} \quad (1)$$

where $(x_1, \dots, x_t)$ are inputs, $(c_1, \dots, c_t)$ are outputs, $(h_1, \dots, h_t)$ are hidden states, i_t is input gate, f_t is forgot gate, o_t is output gate, σ is the Sigmoid activation function, W_x , W_h , and b are the input data weight, $*$ denotes the convolution operator, and $^\circ$ denotes the Hadamard product.

To normalize the input data in the network, the batch normalization layer was used (2) [41].

$$\bar{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}, \quad (2)$$

where x_i is the i -th element of the input, μ_B is the mean within a batch, σ is the variance within a batch, and ϵ is a very minimal number, can be 0 or 1e-12.

The NN architecture consists of three blocks. Each block includes one ConvLSTM layer and the batch normalization layer. After the first and last blocks, there are Dropout regularization layers (3) with values of 0.1 and 0.2, respectively, to prevent network retraining [42].

$$\begin{aligned} r_i^{(l)} &\sim \text{Bernoulli}(p) \\ \tilde{y}^{(l)} &= r^{(l)} * y^{(l)} \\ z_i^{(l+1)} &= w_i^{(l+1)} \tilde{y}^{(l)} + b_i^{(l+1)} \\ y_i^{(l+1)} &= f\left(z_i^{(l+1)}\right), \end{aligned} \quad (3)$$

where r is a vector of independent Bernoulli random variables each of which has probability p of being 1, z^l is the vector of inputs into layer l , y is the vector of outputs from layer l ($y^{(0)} = x$ is the input), $\tilde{y}$ is the thinned outputs, w and b are the weights and biases at layer l , f is the input of the activation function, and $*$ denotes an element-wise product.

At the end of the NN, two regulars densely connected layers at 100 and 1 (4), respectively, were used over the time-distributed layer.

$$x_j^l = f\left(\sum_i x_i^{l-1} w_{i,j}^{l-1} + b_j^{l-1}\right), \quad (4)$$

where x is the input of the layer l , $w_{i,j}$ is the weight vector, and b_j is the bias.

Throughout the latent PS-ConvLSTM, the Rectified Linear Unit function (5) was used to activate the neuron on the layer [43]. However, on the last layer of the densely connected NN layer, the sigmoid activation function of the neuron was used because it is a smooth function and varies in the range from -1 to 1 (6).

$$g(x) = \max\{x, 0\}, \quad (5)$$

$$g(x) = \frac{1}{1 + e^{-x}}. \quad (6)$$

The loss function to be minimized in our network was given by the binary cross entropy loss (7), which is commonly used in binary classification problems like ours [44].

$$\text{Log loss} = -\frac{1}{N} \sum_i^N \sum_j^M y_{ij} \log(p_{ij}), \quad (7)$$

where N is the number of rows, M is the number of classes, and p_{ij} is the probability of the class.

As an adaptive learning rate method or optimizer, the Root Mean Square Propagation algorithm (8) was used [45], which is a root mean square propagation and is a modification of the stochastic gradient descent algorithm, the impulse method, and the basis of the Adam algorithm [46]. The propagation value was set up 0.001. The performance of the NN was evaluated using the Accuracy metric. To evaluate the impact of the number of training epochs on the accuracy of the CNN, training was performed and compared in the range of 1 to 50 epochs.

$$\begin{aligned} v_t &= \rho v_{t-1} + (1 - \rho) * g_t^2 \\ \Delta \omega_t &= -\frac{\eta}{\sqrt{v_t + \epsilon}} * g_t \\ \omega_{t+1} &= \omega_t + \Delta \omega_t, \end{aligned} \quad (8)$$

where ρ is a hyperparameter, η – initial learning rate, v_t – exponential average of squares of gradients, and g_t – gradient at time t along ω_j .

The output dimensions and network parameters of the PS-ConvLSTM architecture are shown in Table 2 and Figure 4.

TABLE 2. The PS-ConvLSTM with output dimensions and network parameters.

Layer	Output dimension	Network parameters
Input	(None, 240, 100, 100, 1)	-
ConvLSTM2D	(None, 240, 100, 100, 16)	$3 \times 3 \times 3, 16$
Batch Normalization	(None, 240, 100, 100, 16)	0.001
Dropout	(None, 240, 100, 100, 16)	0.1
ConvLSTM2D	(None, 240, 100, 100, 32)	$3 \times 3 \times 3, 32$
Batch Normalization	(None, 240, 100, 100, 32)	0.001
ConvLSTM2D	(None, 240, 100, 100, 64)	$3 \times 3 \times 3, 64$
Batch Normalization	(None, 240, 100, 100, 64)	0.001
Dropout	(None, 240, 100, 100, 64)	0.2
Dense	(None, 240, 100, 100, 100)	ReLU

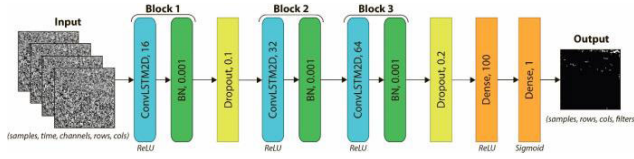


FIGURE 4. PS-ConvLSTM architecture.

C. METRICS FOR PS-ConvLSTM ARCHITECTURE PERFORMANCE EVALUATION

The performance of the PS-ConvLSTM architecture was evaluated using the Accuracy metric (9) based on confusion matrix, that shown in Figure 5 [47].

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP}, \quad (9)$$

where TP – true positive, TN – true negative, FP – false positive, and FN – false negative. The highest and best value of all this metric is 1.0 and the worst is 0.0.

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

FIGURE 5. Confusion matrix.

D. EXPERIMENTAL SETUP

The developed architecture was trained and tested using the Python programming language version 3.5.2 and the TensorFlow Object Detection API [48], an open-source software library for high-performance DL models. The calculations were performed on a computer with an Intel(R) Core(TM) i7-9750H central processing unit (CPU) accelerated by an NVIDIA GeForce GTX 1650 graphics processor unit (GPU) as a platform for studying and testing the proposed PS-ConvLSTM architecture.

The NN training process on our dataset takes about 60 hours on the GPU. Testing the PS-ConvLSTM architecture on a new dataset takes a few seconds, depending on the number of temporary images provided and the size of the test region. To train the PS-ConvLSTM architecture during its development, a new dataset consisting of 207360 patches and a shape as (864, 240, 100, 100, 1) was used. Where 864 is the number of interferogram patches, 240 is the time, 100 by 100 is the patch dimension in pixels, and 1 is the binary gray channel, respectively. The entire dataset was divided into 60% for NN training and 40% for NN validation.

The PS-ConvLSTM architecture was tested on randomly selected patches for each class in the dataset, except for the 8th class - Urban & Water, due to the fact that only one sample was present in the dataset (see Table 1). To externally test the PS-ConvLSTM architecture, a small dataset was used for each class with a shape as (1, 240, 100, 100, 1). Where 1 is the class type (Forest, Urban, Natural, Water, Forest & Urban,

Urban & Natural, Natural & Water, and Forest & Natural), 240 is the number of images in time, 100×100 is the patch dimension in pixels, and 1 is the binary gray channel, respectively. The average time spent on each test set was 1 millisecond.

IV. EXPERIMENTAL RESULTS OF INCREASING POINT DENSITY USING THE PS-ConvLSTM ARCHITECTURE

A. PS-ConvLSTM ARCHITECTURE TRAINING AND VALIDATION RESULTS

To evaluate the impact of the number of training epochs on the CNN accuracy, training was performed and compared in the range of 1 to 50 epochs. The maximum performance of the NN was achieved at the 16th training epoch, providing an internal test accuracy up of 92% and a minimum training loss of less than 0.2. After the 16th epoch, the validation loss was stabilized and the difference between the training and validation loss increased. In this context, and to avoid overfitting, the model was trained on 20 epochs (Figure 6).

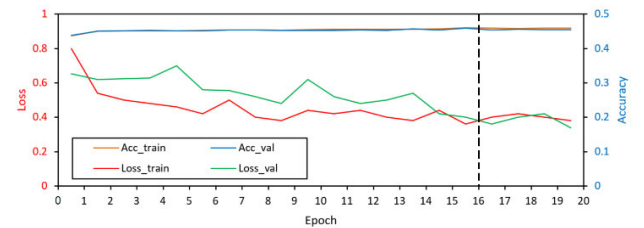


FIGURE 6. Accuracy and loss for each epoch of the training and validation PS-ConvLSTM architecture.

B. TEST RESULTS OF THE PS-ConvLSTM ARCHITECTURE

The results of point density prediction in PS-InSAR data using the newly developed PS-ConvLSTM architecture are shown in Figure 7. Note that the original interferogram images are shown in the first column of Figure 7. The second column shows the results of the PS mask computed using the PS-InSAR techniques developed at the CTTC (Section III-A), which proved to be the ground truth in the PS-ConvLSTM training. The third column contains the results of testing the trained PS-ConvLSTM, where each pixel represents a percentage probability of a stable pixel in the color map, where green is a predicted label, red is a missed label, and blue is an extra label. A visual examination of the final color representation reveals a clear increase in the density of points on the predicted map.

As can be seen from the results obtained in Figure 7, the average test accuracy was 94.5% and the average loss was 0.14. However, the best classes were Nature, Water, and Nature & Water with an average accuracy of 99.94% and a loss of 0.03. This suggests that the developed PS-ConvLSTM architecture can distinguish the tested classes in the best possible way. However, the worst result from testing the model with a relatively low accuracy of 81.21% is on the Forest & Nature class.

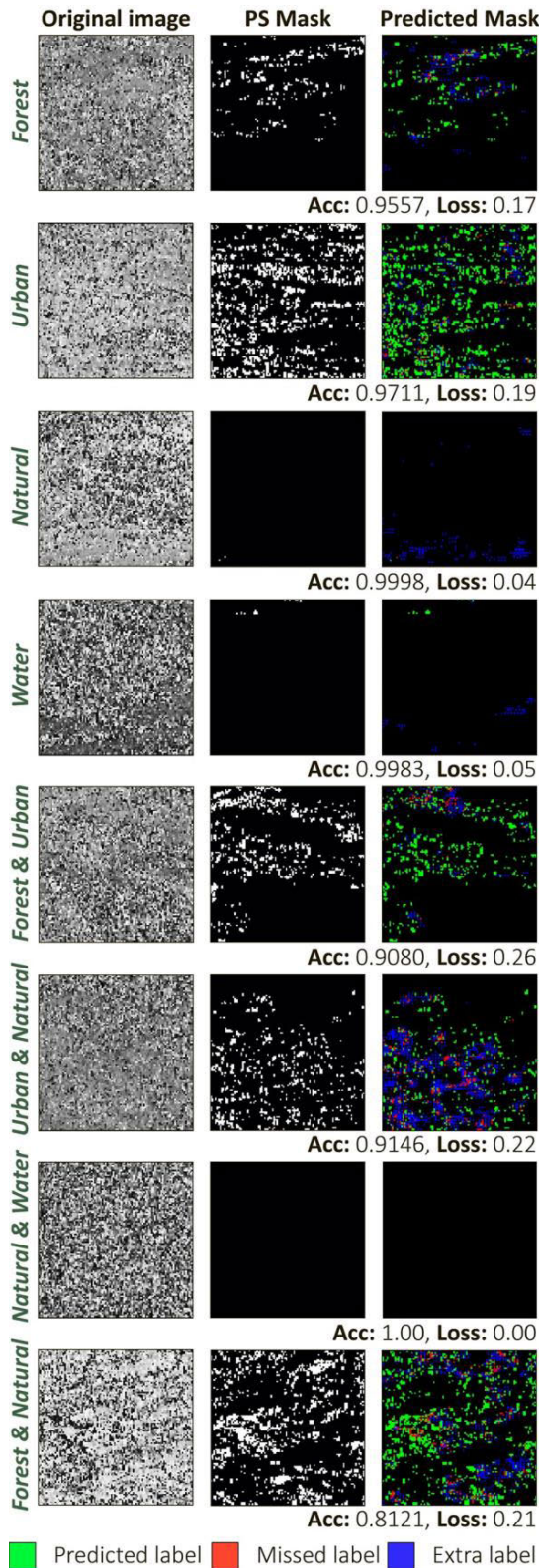


FIGURE 7. A results of increasing point density in PS-InSAR data using the newly developed PS-ConvLSTM architecture, where each path has dimensions of 100×100 pixels.

The presented PS mask approach can overlook stable points, although it was used as a ground truth. Therefore,

we consider that the developed model completed the task of increasing the point density on PS-InSAR data by an average of 15%.

V. DISCUSSION AND CONCLUSION

Insufficient point density on stable and unstable PS pixels in SAR data is one of the current research problems in Earth Observation because the insufficiency prevents from local land surface displacement detection. Despite the fact that there is a huge number of methods to solve this problem (Ferretti et al. [3], [4], Berardino et al. [5], Mora et al. [6], Schmidt and Bürgmann [7], Duro et al. [8], Crosetto et al. [9], [10], López-Quiroz et al. [11], Goel and Adam [12], and Devanthéry et al. [13], Werner et al. [14], Kampes [15], Hooper et al. [16], Costantini et al. [17], Ferretti et al. [18], Van Leijen [19], and Lv et al. [20] and others), all of them are carried out with the participation of an expert or work in a semi-automatic mode. Another point is that the methods proposed in the literature do not allow to obtain a map of the maximum of the point density. In addition, the method is computationally expensive. Depending on the field of study, it can take a considerable amount of time to obtain a result. From the above, it can be concluded that in order to achieve this goal, it is necessary to develop some automatization methods.

In this article, we demonstrated that the novel approach, PS-ConvLSTM, can increase the point density on stable and unstable PS pixels according to PS-InSAR data. To construct the NN, each experiment consisted of a different number of layers, network hyperparameters, number and position of Dropout layers, training time, percentage split of the dataset into test and validation sets, optimization, loss function, and evaluation metric. For example, we found that a more complex model with a large number of layers and kernels could not be trained correctly to solve the problem, and in most cases led to overfitting. In addition, the long time it took to train the model on the limited data also failed to effectively capture the underlying patterns in the data.

Also, the quality of training is influenced by such hyperparameters as dilation rate, loss and activation functions, optimizer, and metrics. Thus, the fine-tuning of the NN was done manually by refining the pre-trained model to better match the PS pixel selection. The proposed network consists of 3 blocks. The PS-ConvLSTM architecture was trained on a new labeled InSAR dataset with two classes: PS points and non-PS point. In addition, the entire data set was divided into 9 areas, including forest, urban, natural, water, forest and urban, urban and natural, natural and water, urban and water, and forest and natural. The results obtained with the PS-ConvLSTM model showed an average test accuracy of 94.5% and an average loss of 0.14. The developed model increased the point density on PS-InSAR data by an average of 15%. As illustrated in Figure 7, the majority of scenes, including Forest, Urban, Forest&Urban, Urban&Natural, Forest&Natural, are capable of containing stable points that have been accurately identified with a 94.5% success rate by the developed NN. However, an examination of scenes

such as Natural, Water and Natural&Water reveals that in the majority of cases, there is a paucity of PSs, or indeed, no points at all. This is exemplified by the Water class. This indicates that even the ground truth employed in the experiment may not be entirely reliable, as evidenced by the accuracy percentage on the predicted mask. For instance, in the case of the Water instance, the ground truth mask displays stable points, which do not always align with the model's actual prediction.

The proposed development can be implemented in practice or as a separate step in InSAR pre-processing. The presented results are of great interest both for scientific and practical purposes, since the proposed approach based on DL allows to increase the density of map points in a fully automatic mode.

Compared to classical methods (Fig. 3), the proposed PS-ConvLSTM network is fully automated and does not require the participation of an expert. It also improves point density, even in scenes with natural objects. In addition, the method is not computationally expensive and does not require a significant amount of time to obtain a result, depending on the field of study. However, despite the high accuracy results and the advantages of using the developed algorithm, we encountered some limitations. The most important issue is the inability to evaluate the performance of the PS-ConvLSTM based on the available ground truth data because the ground truth itself tends to underestimate point density.

In future research, we therefore plan to apply the algorithm in real-world scenarios on different test areas and analyze whether these proposed points by ConvLSTM network are reliable. Furthermore, a comparison will be made between the developed network and other existing PS-InSAR methods, including interferometric point target analysis, stable point network, Stanford method for PS, spatio-temporal unwrapping network, Delft PSI, coherence pixel technique, small baseline subset, and others [49]. It is of particular interest to note the expansion of the test areas to include phenomena such as building collapses, earthquakes and volcanic eruptions.

ABBREVIATIONS

PS	Permanent Scatterer.
InSAR	Interferometric Synthetic Aperture Radar.
CNN	Convolutional Neural Network.
RNN	Recurrent Neural Network.
LSTM	Long Short-Term Memory.
ConvLSTM	Convolutional Long Short-Term Memory.
CPU	Central Processing Unit.
GPU	Graphics Processing Unit.

ACKNOWLEDGMENT

The authors are grateful to Oriol Monserrat from the Centre Tecnologic de Telecomunicacions de Catalunya (Barcelona, Spain) for providing the image dataset. They are very grateful to the reviewers for their valuable comments, which helped to improve the article.

DATA AVAILABILITY

The datasets generated during and analyzed during the current study are publicly available at the following link https://github.com/ansafo/AI_InSAR.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

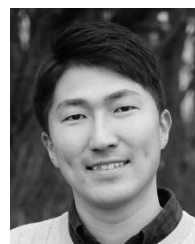
REFERENCES

- [1] Z. Lu, O.-I. Kwoun, and R. P. Rykhus, "Interferometric synthetic aperture radar (InSAR): Its past, present and future," *Photogramm. Eng. Remote Sens.*, vol. 73, no. 3, pp. 217–221, 2007.
- [2] M. Crosetto, O. Monserrat, M. Cuevas-González, N. Devanthéry, and B. Crippa, "Persistent scatterer interferometry: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 115, pp. 78–89, May 2016, doi: [10.1016/j.isprsjprs.2015.10.011](https://doi.org/10.1016/j.isprsjprs.2015.10.011).
- [3] A. Ferretti, C. Prati, and F. Rocca, "Nonlinear subsidence rate estimation using permanent scatterers in differential SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 5, pp. 2202–2212, Sep. 2000, doi: [10.1109/36.868878](https://doi.org/10.1109/36.868878).
- [4] A. Ferretti, C. Prati, and F. Rocca, "Permanent scatterers in SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 1, pp. 8–20, Jan. 2001, doi: [10.1109/36.898661](https://doi.org/10.1109/36.898661).
- [5] P. Berardino, G. Fornaro, R. Lanari, and E. Sansosti, "A new algorithm for surface deformation monitoring based on small baseline differential SAR interferograms," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 11, pp. 2375–2383, Nov. 2002, doi: [10.1109/TGRS.2002.803792](https://doi.org/10.1109/TGRS.2002.803792).
- [6] O. Mora, J. J. Mallorqui, and A. Broquetas, "Linear and nonlinear terrain deformation maps from a reduced set of interferometric SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 10, pp. 2243–2253, Oct. 2003, doi: [10.1109/TGRS.2003.814657](https://doi.org/10.1109/TGRS.2003.814657).
- [7] D. A. Schmidt and R. Bürgmann, "Time-dependent land uplift and subsidence in the Santa Clara valley, California, from a large interferometric synthetic aperture radar data set," *J. Geophys. Res., Solid Earth*, vol. 108, no. B9, pp. 1–13, 2003, doi: [10.1029/2002JB002267](https://doi.org/10.1029/2002JB002267).
- [8] J. Duro, J. Inglada, J. Closa, N. Adam, and A. Arnaud, "High resolution differential interferometry using time series of ERS and ENVISAT SAR data," in *Proc. FRINGE*, vol. 550, Jun. 2004, p. 72.
- [9] M. Crosetto, E. Biescas, J. Duro, J. Closa, and A. Arnaud, "Generation of advanced ERS and Envisat interferometric SAR products using the stable point network technique," *Photogramm. Eng. Remote Sens.*, vol. 74, no. 4, pp. 443–450, Apr. 2008, doi: [10.14358/pers.74.4.443](https://doi.org/10.14358/pers.74.4.443).
- [10] M. Crosetto, B. Crippa, and E. Biescas, "Early detection and in-depth analysis of deformation phenomena by radar interferometry," *Eng. Geol.*, vol. 79, nos. 1–2, pp. 81–91, Jun. 2005, doi: [10.1016/j.enggeo.2004.10.016](https://doi.org/10.1016/j.enggeo.2004.10.016).
- [11] P. López-Quiroz, M.-P. Doin, F. Tupin, P. Briole, and J.-M. Nicolas, "Time series analysis of Mexico City subsidence constrained by radar interferometry," *J. Appl. Geophys.*, vol. 69, no. 1, pp. 1–15, Sep. 2009, doi: [10.1016/j.jappgeo.2009.02.006](https://doi.org/10.1016/j.jappgeo.2009.02.006).
- [12] K. Goel and N. Adam, "A distributed scatterer interferometry approach for precision monitoring of known surface deformation phenomena," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 9, pp. 5454–5468, Sep. 2014, doi: [10.1109/TGRS.2013.2289370](https://doi.org/10.1109/TGRS.2013.2289370).
- [13] N. Devanthéry, M. Crosetto, O. Monserrat, M. Cuevas-González, and B. Crippa, "An approach to persistent scatterer interferometry," *Remote Sens.*, vol. 6, no. 7, pp. 6662–6679, Jul. 2014, doi: [10.3390/rs6076662](https://doi.org/10.3390/rs6076662).
- [14] C. Werner, U. Wegmüller, T. Strozzi, and A. Wiesmann, "Interferometric point target analysis for deformation mapping," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, vol. 7, Jul. 2003, pp. 4362–4364, doi: [10.1109/IGARSS.2003.1295516](https://doi.org/10.1109/IGARSS.2003.1295516).
- [15] B. M. Kampes Ed., "The permanent scatterer technique," in *Radar Interferometry: Persistent Scatterer Technique*. Dordrecht, The Netherlands: Springer, 2006, pp. 5–30, doi: [10.1007/978-1-4020-4723-7_2](https://doi.org/10.1007/978-1-4020-4723-7_2).
- [16] A. Hooper, H. Zebker, P. Segall, and B. Kampes, "A new method for measuring deformation on volcanoes and other natural terrains using InSAR persistent scatterers," *Geophys. Res. Lett.*, vol. 31, no. 23, pp. 1–5, 2004, doi: [10.1029/2004GL021737](https://doi.org/10.1029/2004GL021737).

- [17] M. Costantini, S. Falco, F. Malvarosa, and F. Minati, "A new method for identification and analysis of persistent scatterers in series of SAR images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2008, pp. II-449–II-452, doi: [10.1109/IGARSS.2008.4779025](https://doi.org/10.1109/IGARSS.2008.4779025).
- [18] A. Ferretti, A. Fumagalli, F. Novali, C. Prati, F. Rocca, and A. Rucci, "A new algorithm for processing interferometric data-stacks: SqueeSAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 9, pp. 3460–3470, Sep. 2011, doi: [10.1109/TGRS.2011.2124465](https://doi.org/10.1109/TGRS.2011.2124465).
- [19] F. J. Van Leijen, "Persistent scatterer interferometry based on geodetic estimation theory," Delft Univ. Technol., Delft, The Netherlands, 2014, no. 1, pp. 1–220.
- [20] X. Lv, B. Yazici, M. Zeghal, V. Bennett, and T. Abdoun, "Joint-scatterer processing for time-series InSAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7205–7221, Nov. 2014, doi: [10.1109/TGRS.2014.2309346](https://doi.org/10.1109/TGRS.2014.2309346).
- [21] N. D. Arasa, "High-resolution deformation measurement using," Ph.D. thesis, Departament de Teoria del Senyal i Comunicacions, Universitat Politècnica de Catalunya, Barcelona, Spain, 2014. Accessed: Jan. 19, 2024. [Online]. Available: <https://www.tdx.cat/handle/10803/283579>
- [22] D. Wang, M. Even, and H. Kutterer, "Deep learning based distributed scatterers acceleration approach: Distributed scatterers prediction net," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 115, Dec. 2022, Art. no. 103112, doi: [10.1016/j.jag.2022.103112](https://doi.org/10.1016/j.jag.2022.103112).
- [23] J. Hu, W. Wu, R. Gui, Z. Li, and J. Zhu, "Deep learning-based homogeneous pixel selection for multitemporal SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5234518, doi: [10.1109/TGRS.2022.3203975](https://doi.org/10.1109/TGRS.2022.3203975).
- [24] Y. Zhang, J. Wei, M. Duan, Y. Kang, Q. He, H. Wu, and Z. Lu, "Coherent pixel selection using a dual-channel 1-D CNN for time series InSAR analysis," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 112, Aug. 2022, Art. no. 102927, doi: [10.1016/j.jag.2022.102927](https://doi.org/10.1016/j.jag.2022.102927).
- [25] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, p. 53, Mar. 2021, doi: [10.1186/s40537-021-00444-8](https://doi.org/10.1186/s40537-021-00444-8).
- [26] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network—A deep learning approach," *Proc. Comput. Sci.*, vol. 132, pp. 679–688, Jan. 2018, doi: [10.1016/j.procs.2018.05.069](https://doi.org/10.1016/j.procs.2018.05.069).
- [27] R. M. Schmidt, "Recurrent neural networks (RNNs): A gentle introduction and overview," 2019, *arXiv:1912.05911*.
- [28] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," 2014, *arXiv:1402.1128*.
- [29] N. Sharma, V. Jain, and A. Mishra, "An analysis of convolutional neural networks for image classification," *Proc. Comput. Sci.*, vol. 132, pp. 377–384, Jan. 2018, doi: [10.1016/j.procs.2018.05.198](https://doi.org/10.1016/j.procs.2018.05.198).
- [30] Y. Dupont, M. Dinarelli, and I. Tellier, "Label-dependencies aware recurrent neural networks," in *Computational Linguistics and Intelligent Text Processing* (Lecture Notes in Computer Science), A. Gelbukh, Ed. Cham, Switzerland: Springer, 2018, pp. 44–66, doi: [10.1007/978-3-319-77113-7_4](https://doi.org/10.1007/978-3-319-77113-7_4).
- [31] A. B. Dieng, C. Wang, J. Gao, and J. Paisley, "TopicRNN: A recurrent neural network with long-range semantic dependency," 2016, *arXiv:1611.01702*.
- [32] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 6, no. 2, pp. 107–116, Apr. 1998, doi: [10.1142/S0218488598000094](https://doi.org/10.1142/S0218488598000094).
- [33] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735).
- [34] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," 2015, *arXiv:1506.04214*.
- [35] S. M. Mirmazloumi, Y. Wassie, L. Nava, M. Cuevas-González, M. Crosetto, and O. Monserrat, "InSAR time series and LSTM model to support early warning detection tools of ground instabilities: Mining site case studies," *Bull. Eng. Geol. Environ.*, vol. 82, no. 10, p. 374, Sep. 2023, doi: [10.1007/s10064-023-03388-w](https://doi.org/10.1007/s10064-023-03388-w).
- [36] M. Peng, M. Motagh, Z. Lu, Z. Xia, Z. Guo, C. Zhao, and Q. Liu, "Characterization and prediction of InSAR-derived ground motion with ICA-assisted LSTM model," *Remote Sens. Environ.*, vol. 301, Feb. 2024, Art. no. 113923, doi: [10.1016/j.rse.2023.113923](https://doi.org/10.1016/j.rse.2023.113923).
- [37] A. Tiwari, A. B. Narayan, and O. Dikshit, "Deep learning networks for selection of measurement pixels in multi-temporal SAR interferometric processing," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 169–182, Aug. 2020, doi: [10.1016/j.isprsjprs.2020.06.005](https://doi.org/10.1016/j.isprsjprs.2020.06.005).
- [38] W. Köppen, E. Volken, and S. Brönnimann, "The thermal zones of the earth according to the duration of hot, moderate and cold periods and to the impact of heat on the organic world," *Meteorologische Zeitschrift*, vol. 20, no. 3, pp. 351–360, Jun. 2011, doi: [10.1127/0941-2948/2011/105](https://doi.org/10.1127/0941-2948/2011/105).
- [39] *ASF Data Search*. Accessed: Jan. 19, 2024. [Online]. Available: <https://search.asf.alaska.edu/#/>
- [40] X. X. Zhu, S. Montazeri, M. Ali, Y. Hua, Y. Wang, L. Mou, Y. Shi, F. Xu, and R. Bamler, "Deep learning meets SAR: Concepts, models, pitfalls, and perspectives," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 4, pp. 143–172, Dec. 2021, doi: [10.1109/MGRS.2020.3046356](https://doi.org/10.1109/MGRS.2020.3046356).
- [41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.
- [42] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [43] A. Fred Agarap, "Deep learning using rectified linear units (ReLU)," 2018, *arXiv:1803.08375*.
- [44] U. Ruby and V. Yendapalli, "Binary cross entropy with deep learning technique for image classification," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 4, pp. 5393–5397, Aug. 2020, doi: [10.30534/ijatcse/2020/175942020](https://doi.org/10.30534/ijatcse/2020/175942020).
- [45] T. Tieleman and G. Hinton, "Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude," *COURSERA, Neural Netw. Mach. Learn.*, vol. 7, no. 4, pp. 26–31, 2012.
- [46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [47] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," 2020, *arXiv:2010.16061*.
- [48] *TensorFlow 2 Object Detection API Tutorial—TensorFlow 2 Object Detection API Tutorial Documentation*. Accessed: Jan. 19, 2024. [Online]. Available: <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/>
- [49] S. M. Mirmazloumi, "Advanced modeling of InSAR time series for ground displacement hazard assessment," Ph.D. thesis, Departament de Teoria del Senyal i Comunicacions, Universitat Politècnica de Catalunya, Barcelona, Spain, 2024, doi: [10.5821/dissertation-2117-410341](https://doi.org/10.5821/dissertation-2117-410341).



ANASTASIIA SAFONOVA received the M.S. degree in computer science from Siberian Federal University, Krasnoyarsk, Russia, in 2015, and the double Ph.D. degrees in remote sensing and data mining from Siberian Federal University, and the University of Granada, Granada, Spain, in 2020 and 2021, respectively. She is currently a Postdoctoral Researcher with the Research Platform "Data Analysis and Simulation," Artificial Intelligence for Smart Agriculture Working Group, Leibniz-Centre for Agricultural Landscape Research (ZALF), Müncheberg, Germany. Her main research interests include remote sensing, data mining, deep learning, neural networks, and image processing.



MASAHIRO RYO received the Ph.D. degree in civil engineering from Tokyo Institute of Technology, in 2015. He is a Professor of environmental data science and a Leader with Brandenburg University of Technology Cottbus-Senftenberg, Germany. His research interest includes the application of artificial intelligence for agricultural and biodiversity problems.

...