










Automatic Detection of Building Displacements Through Unsupervised Learning From InSAR Data

Rıdvan Salih Kuzu , Leonardo Bagaglini , Yi Wang , *Student Member, IEEE*, Corneliu Octavian Dumitru , Nassim Ait Ali Braham , Giorgio Pasquali , Filippo Santarelli , Francesco Trillo, Sudipan Saha , *Member, IEEE*, and Xiao Xiang Zhu , *Fellow, IEEE*

Abstract—We introduce an unsupervised learning method that aims to identify building anomalies using Interferometric Synthetic Aperture Radar (InSAR) time-series data. Specifically, we leverage data obtained from the European Ground Motion Service to develop our proposed approach, which employs a long short-term memory autoencoder model and a reconstruction loss function based on a soft variant of the dynamic time warping, namely “soft-DTW”. We deliberately utilize this loss function for its ability to compare time-series that are not aligned in time, unlike the other conventional reconstruction losses that do not account for time shifts. Moreover, we enhance the performance of anomaly detection by smoothing inputs with a Hann window and defining the learning objective to reconstruct the time order of randomly permuted input series. Our experimental findings, based on persistent scatterer data from Rome, Italy, reveal that our method outperforms several unsupervised machine learning and deep learning methods in detecting various types of building displacement, such as trend, noise, and step anomalies. Additionally, quantitative and qualitative evaluations validate the efficacy of our approach in identifying potentially anomalous buildings. Thus, our method offers a promising solution for detecting anomalies in PS-InSAR time-series, which could have substantial implications in the fields of urban monitoring and infrastructure management.

Index Terms—Anomaly detection, autoencoders, building displacements, dynamic time warping (DTW), long short-term memory (LSTM) networks, persistent scatterer (PS), synthetic aperture radar interferometry (InSAR), unsupervised learning.

Manuscript received 4 April 2023; revised 29 May 2023; accepted 5 July 2023. Date of publication 20 July 2023; date of current version 2 August 2023. This work was supported in part by the European Space Agency with contract as part of the RePreSent project under Grant 4000137253/22/I-DT, and in part by the Helmholtz Association’s Initiative and Networking Fund through Helmholtz AI and on the HAICORE/FZJ partition. (*Corresponding author: Corneliu Octavian Dumitru.*)

Rıdvan Salih Kuzu, Yi Wang, Corneliu Octavian Dumitru, and Nassim Ait Ali Braham are with the Remote Sensing Technology Institute, German Aerospace Center, 82234 Wessling, Germany (e-mail: ridvan.kuzu@dlr.de; yi.wang@dlr.de; corneliu.dumitru@dlr.de; nassim.aitalibraham@dlr.de).

Leonardo Bagaglini is with the Space Technologies Lab, Leonardo S.p.A., 00156 Rome, Italy (e-mail: leonardo.bagaglini.ext@leonardo.com).

Giorgio Pasquali, Filippo Santarelli, and Francesco Trillo are with the e-GEOS, 00156 Rome, Italy (e-mail: giorgio.pasquali@e-geos.it; filippo.santarelli@e-geos.it; francesco.trillo@e-geos.it).

Sudipan Saha is with the Yardi School of Artificial Intelligence, Indian Institute of Technology Delhi, New Delhi 110016, India (e-mail: sudipan.saha@scai.iitd.ac.in).

Xiao Xiang Zhu is with the Chair of Data Science in Earth Observation, Technical University of Munich, 80333 Munich, Germany (e-mail: xiaoxiang.zhu@tum.de).

The source code of the study: <https://github.com/ridvansalihkuzu/representlib>
Digital Object Identifier 10.1109/JSTARS.2023.3297267

I. INTRODUCTION

BUILDING displacement anomalies refer to abnormal movements or shifts of a building or structure from its original position. The following are some common types of building displacement anomalies.

- 1) *Subsidence*: Sinking of the ground beneath a building, causing the structure to settle and shift downward [1].
- 2) *Heave*: Swelling or rising of the ground, causing the foundation of a building to lift and shift upward [2].
- 3) *Lateral movement*: Horizontal movement of a building caused by earthquakes, landslides, or soil erosion [3].
- 4) *Slope instability*: Instability of the ground beneath a building, causing the structure to tilt or lean [4].
- 5) *Differential settlement*: Uneven settling of different parts of a building, leading to structural damage [5].
- 6) *Creep*: Slow deformation of a building over time due to constant stress [6].
- 7) *Vibration*: Constant shaking or vibration of a building, such as from nearby construction or heavy traffic [7].
- 8) *Expansion and contraction*: Movement of a building’s materials due to changes in temperature, causing the structure to shift slightly [8].

Interferometric Synthetic Aperture Radar (InSAR) satellite measurements are an effective tool for monitoring ground motion, position of buildings [9], and the aforementioned displacements with millimetric resolution over long periods of time. The persistent scatterer (PS) pair method [10], is particularly useful for detecting differential displacements of buildings at multiple positions with few assumptions about the background environment. As a result, anomalous behaviors in building motion can be detected through PS time-series, which are commonly used to perform risk assessments in hazardous areas and diagnostic analyses after damage or collapse events. However, most of the current autonomous early warning systems based on PS-InSAR data are limited to detecting changes in linear trends and rely on simple sinusoidal and polynomial models [11]. This can be problematic if background signals exhibit more complex behaviors, as anomalous displacements may be difficult to identify. To address this issue, we rely on an anomaly detection method using artificial intelligence (AI) algorithms to identify potentially anomalous building motions based on PS long time-series data.

On the other hand, the development of a machine learning (ML) or deep learning (DL) system for detecting anomalies

in building displacements typically requires ground-truth data from in situ measurements [12][13], such as displacement rate and type from reference buildings, which can be difficult to obtain due to the need for regular physical monitoring by experts. That is why the proposed solutions in this work rely on unsupervised representation learning based on the combination of autoencoders and long short-term memory (LSTM) networks to mitigate the lack of annotated data, similar to some recent efforts [14], [15]. However, what distinguishes our work from the recent efforts is that we propose 1) a data preprocessing approach based on windowing and permutation of time-series; and 2) a novel reconstruction loss function as compared to conventional loss objectives applied for the autoencoders, resulting in significantly improved anomaly detection accuracy. The proposed approach is also validated by comparing its performance with several ML and DL benchmarks. Additionally, it is worth mentioning that our study represents one of the pioneering endeavors in leveraging AI to develop value-added solutions for urban monitoring by utilizing the European Ground Motion Service [16], which was launched in 2022 by the Copernicus programme of the European Union.

In the light of those contributions, the objective of our study is to assess how effective our proposed method is in identifying anomalous building movements amidst nonanomalous ones, in the context of three displacement scenarios. These scenarios, namely: 1) *trend anomalies*; 2) *noise (Gaussian-like) anomalies*; and 3) *step anomalies*, were specifically chosen to serve as representatives of the diverse range of abnormalities that buildings may encounter, as discussed earlier.

The rest of this article is organized as follows. Section II presents a literature review on PS-InSAR based displacement monitoring methods. Section III provides a detailed account of the anomaly scenarios and their characteristics as well as the dataset used in our experiments. Section IV explains the unsupervised learning methodologies proposed in this study, Section V describes the experimental protocols for different learning strategies, and Section VI provides the performance results for detecting anomalies. Finally, Section VII concludes the article and discusses future work.

II. LITERATURE BACKGROUND

Building anomaly and displacement detection are critical for ensuring the safety and longevity of structures. PS-InSAR is a remote sensing technique that uses SAR images to monitor and estimate the displacement of buildings over time. However, traditional linear regression models exploiting PS-InSAR may not be sufficient to capture the complex deformation patterns of buildings. That is why, recent studies have investigated the use of ML, DL, and statistical methods to improve the accuracy of building anomaly and displacement detection with PS-InSAR data analysis.

Deep neural networks, including convolutional neural networks (CNN) and LSTM networks, have demonstrated potential in accurately detecting ground anomalies and displacement. For instance, Anantrasirichai et al. [17] proposed a CNN that utilized sparse satellite InSAR data, where “sparse” refers to the limited

availability of measurement points in the InSAR images, and they addressed challenges such as noise, lack of data, and slow deformation signals, showing potential for automated ground motion analysis systems. Chen et al. [14] developed an LSTM neural network to predict land subsidence using time-series InSAR data, outperforming multilayer perceptron and recurrent neural network models, and enabling early warning and hazard relief. Lattari et al. [13] also utilized LSTM cells and time-gated LSTM cells to monitor seismic faults, subsidence, landslides, and urban structure from nonuniformly sampled time-series.

Clustering techniques have also been applied to identify anomalous areas in PS-InSAR time-series data. One such approach was proposed by Milone et al. [18] used a combination of hierarchical clustering and k -medoids, a variant of k -means clustering, to identify deformation in an urban area with complex ground structure. Similarly, Zhu et al. [11] proposed a hierarchical clustering algorithm and a signal processing method to automatically detect potential instability risks affecting buildings and infrastructures by grouping neighboring PS and analyzing anomalous velocity trends and accelerations using trend variation analysis. On the other hand, Martin et al. [19] utilized a deep temporal clustering method that involved the use of convolutional autoencoders and bidirectional LSTM to detect ground displacements. Thereafter, Shakeel et al. [15] presented a method, utilizing an LSTM autoencoder-based anomaly detector with semivariogram analysis and density-based spatial clustering, for automatic detection of transient deformation in InSAR datasets achieving an accuracy of 91.25%, demonstrated by detecting a synthetic deformation test case and a real earthquake in Turkey.

Apart from ML- and DL-based techniques, statistical methods have been explored for building anomaly and displacement detection with PS-InSAR data analysis. For instance, Bernardino et al. [20] proposed the small baseline subset (SBAS) technique, which involves selecting a subset of interferograms with small temporal baselines and using a least squares approach to estimate surface deformation. Similarly, Ferretti et al. [21] introduced an extension of the PS technique using a Wiener filter for spatio-temporal estimation of nonlinear target motion, taking into account different phase contributions including motion, atmospheric effects, and decorrelation noise. Subsequently, Yu et al. [22] conducted a study on Shanghai Pudong International airport, which was constructed on ocean-reclaimed lands and silt tidal flat, and used the SBAS technique and geotechnical models to continuously monitor and analyze the ground settlement, revealing that certain areas experienced significant ground subsidence. Following that, Evers et al. [23] employed different methods for smoothing the time-series and decomposing its trend and periodic components. This was achieved through techniques such as polynomial regression. Once the data was cleaned, they treated the identification of anomalies as a statistical outlier detection problem. In a recent study by Letsios et al. [24], the Bayesian change-point step detector algorithm was used alongside statistical variables such as fitting index, temporal coherence, residual height, and radar cross-section to identify newly constructed buildings based on the appearance

TABLE I
STATISTICS OF INSAR-PSP DATABASE COVERING THE CITY OF ROME

	Training Set	Test Set
Number of PS time-series	443,705	99,028
Number of buildings	68,055	15,043
PS instance per building	1 – 420	1 – 619
Area per building	8 m ² – 3500 m ²	8 m ² – 3700 m ²

and continued presence of corresponding PS points in the time-series.

Overall, PS-InSAR is a powerful tool for monitoring building deformation over time. While linear regression models have been widely used for PS-InSAR data analysis, recent studies have shown that non-linear models, clustering techniques, and statistical methods can significantly improve the accuracy of deformation detection. The integration of different techniques may also offer promising avenues for future research in this field.

III. DATABASE

A. Creation of Training and Test Datasets

The research made use of data obtained from the European Ground Motion Service (EGMS) [16]. EGMS data, derived from Sentinel-1 (S1) satellite radar images using Persistent Scatterer Interferometry (PSI), identifies stable “scatterers” in S1 images, typically fixed ground features. By comparing the phase difference of the radar signal from these scatterers in sequential images, displacement or ground motion is detected and measured, forming the PS time-series.

The specific EGMS dataset utilized for our study encapsulates a defined rectangular region around Rome, Italy, and it comprises approximately 500 000 PS time-series, each with 300 time stamps indicating deformations estimated in millimeter resolution along the line of sight (LOS) with the statistics given in Table 1 and Fig. 1. These time-series measurements were aggregated over 80 000 building footprints and spanned the period from 2015 to 2020. The data acquisition process involved the following steps.

- 1) The EGMS data were clipped to a rectangular area surrounding Rome, whose extent is defined by the longitude 12.3699°E – 12.6195°E and latitude 41.7933°N – 41.9924°N.
- 2) The buildings’ footprints were downloaded from the OpenStreetMap website.
- 3) A spatial join was conducted between the polygons of the PS instances and the building footprints as illustrated in Fig. 2. This assigned each PS instance to a building polygon with a unique identifier, and the resulting database was saved as a CSV file.

Thus, the database comprises clusters of points that correspond to footprints, and it is imperative that the training and test splits preserve the internal relationships among PS instances of the same building, and ensure that the splits are disjoint sets in

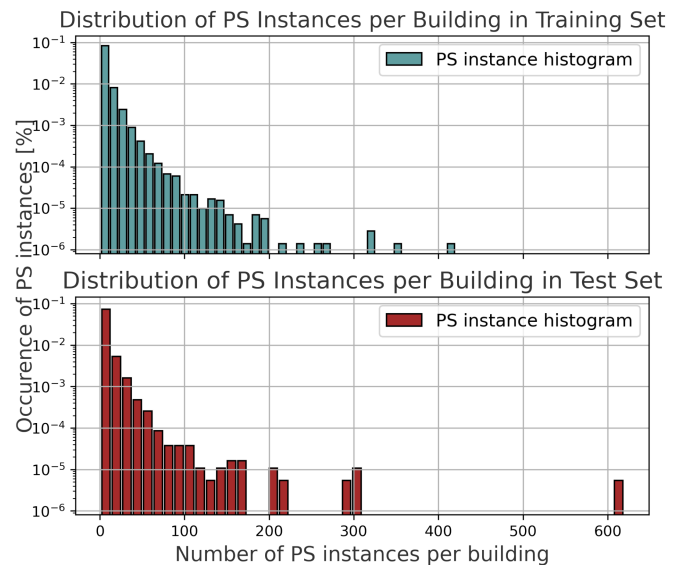


Fig. 1. Distribution of PS per building in the dataset. (The average number of PS per building is 6.52 in the Training set and 6.58 in the Test set).

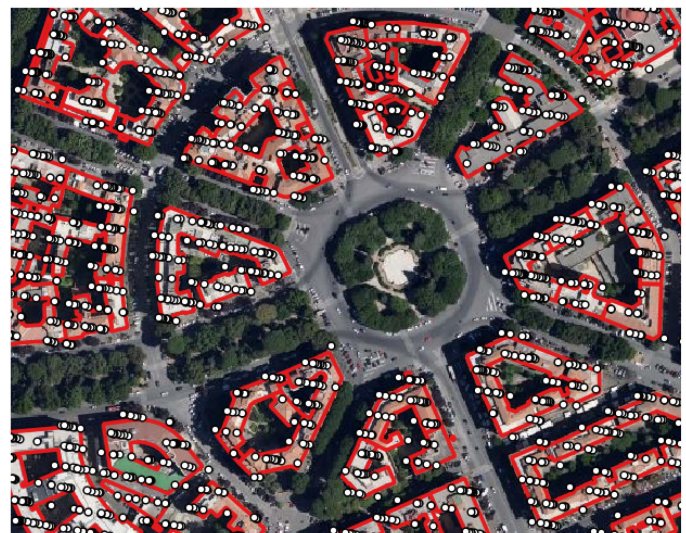


Fig. 2. Demonstration of building footprints (red lines) and their corresponding PS instances (white dots).

terms of building footprints. To achieve this, the following steps were taken.

- 1) An initial 80 : 20 ratio between the training and test set was determined from PS instances.
- 2) PS instances falling within a particular building were assigned to the respective training or test set.
- 3) The final split on points was utilized to obtain disjoint building footprints in terms of the training and test sets. This resulted in an actual training:test ratio of 82 : 18.

B. Ground-Truth Generation

In the absence of ground-truth anomalies, a common approach is to generate synthetic deformation to train or validate proposed

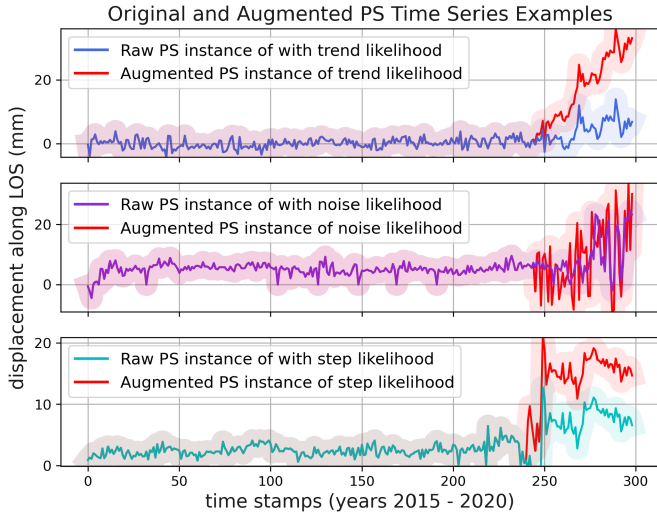


Fig. 3. Comparison augmented PS examples per anomaly type to be utilized as ground truths in performance evaluation. The *trend* anomaly type is created by adding a slope to the LOS values. The *noise* anomaly type is generated by amplifying the LOS values of a certain portion of the original time-series. The *step* anomaly type is produced by vertically shifting the LOS values of a certain fragment in time stamps, either positively or negatively.

methods. An example of this is demonstrated by Anantrasirichai et al. [17], who simulated the presence of slow deformation in both training and validation by creating a synthetic velocity map. Lattari et al. [13] also employed a similar approach, augmenting their data with change points of varying slopes to simulate slow deformation. Additionally, Shakeel et al. [15] added Gaussian peaks to the test data in order to simulate instant deformations caused by earthquakes. Moreover, some studies suggest that building deformation signals can be grouped as linear/trending, periodic, and random noise, based on time stamps of PS time-series [11], [23].

Given these practices and the absence of actual in situ reference measurements from the region of interest for our experiments, we introduced three deformation scenarios as synthetic ground truths, representing common anomaly types mentioned in Section I. These scenarios are as follows.

- 1) *Trend anomaly scenario* which indicates gradual changes in building positions caused by subsidence, heave, creep, or ongoing construction.
- 2) *Noise anomaly scenario* which represents high and frequent deviations in building positions due to creep, vibration, expansion, and contraction.
- 3) *Step anomaly scenario* which indicates sudden and significant changes in building positions caused by lateral movement and differential settlement.

By considering these scenarios, to evaluate the effectiveness of our proposed approaches in distinguishing anomalous building motions from nonanomalous ones, we introduced a data augmentation approach to randomly replace the raw PS time-series in the evaluation partition with their augmented, anomalous counterparts, and we labeled them as *trend*, *noise*, and *step* anomaly types, as shown in Fig. 3. Later in the inference phase, we gradually substituted the instances of the original PS

time-series of each building with their augmented counterparts, incorporating up to 25% anomaly instance rate to simulate the anomaly presence of varying proportions in evaluation samples, as there are no benchmark values available for the actual percentage of anomalous PS time-series observations in the dataset. It is important to note that the nonaugmented (raw) partition of the evaluation data may still contain anomalous PS examples that have more prominent anomaly attributes than the artificially created augmented ones. This may result in the measured accuracy of displacements being lower than the actual accuracy in our proposed prediction models.

C. Initial Investigation on Data

Before constructing deep feature representations, we conducted an examination on the dataset to analyze the proximity of the samples to the anomaly types, namely *trend*, *noise*, and *step*. Our goal was to gain an overall understanding of the data distribution with respect to the anomaly clusters. To achieve this, we performed *k*-means clustering on the synthetic time-series to identify the cluster centres of each anomaly type. However, traditional Euclidian distance-based clustering methods are invariant to time shifts and ignore the time dimension of the data. As a result, if two time-series are highly correlated but one is shifted by a single time step, Euclidean distance will incorrectly measure them as farther apart. Therefore, we utilized a variant of dynamic time warping (DTW) to compare and cluster time-series, which is a technique that measures the similarity between two temporal sequences that do not align exactly in time due to variable size, shift, or dilation across the time dimension [25].

The *soft-DTW* is a differentiable version of the DTW that computes the optimal warping path recursively by accumulating the distance of each possible path through two sequences. The minimum accumulated distance path is selected using the following formula [26]:

$$\text{soft-DTW}^\gamma(\mathbf{x}, \mathbf{x}') = \min_{\pi \in \mathcal{A}(\mathbf{x}, \mathbf{x}')} \gamma \sum_{(i,j) \in \pi} d(x_i, x'_j)^2. \quad (1)$$

Here, π represents the alignment path, \mathcal{A} is the matrix of alignment between the sequences \mathbf{x} and \mathbf{x}' , \min^γ denotes the *soft-min* operator, d is the distance between the elements of the compared sequences, and γ is a parameter that controls the degree of softness. The soft-min operator is defined as $-\gamma \log \sum_i e^{-a_i/\gamma}$, where γ is a positive constant and a_i are the values being compared.

After identifying the cluster centres of anomalies related to *trend*, *noise*, and *step* changes using (1) as a distance measurement method for *k*-means clustering, the pairwise distances between each cluster centre and all PS time-series were separately calculated. The resulting probability density of distances for all PS instances to each cluster centre is presented in Fig. 4. An anomaly threshold was set for each distribution using the *lower-whisker* value of the distance distribution, which is 2.698σ , where σ stands for standard deviation.

As seen in Fig. 4, the analysis among the PS points below the threshold revealed that the percentage of a *trend* anomaly is the

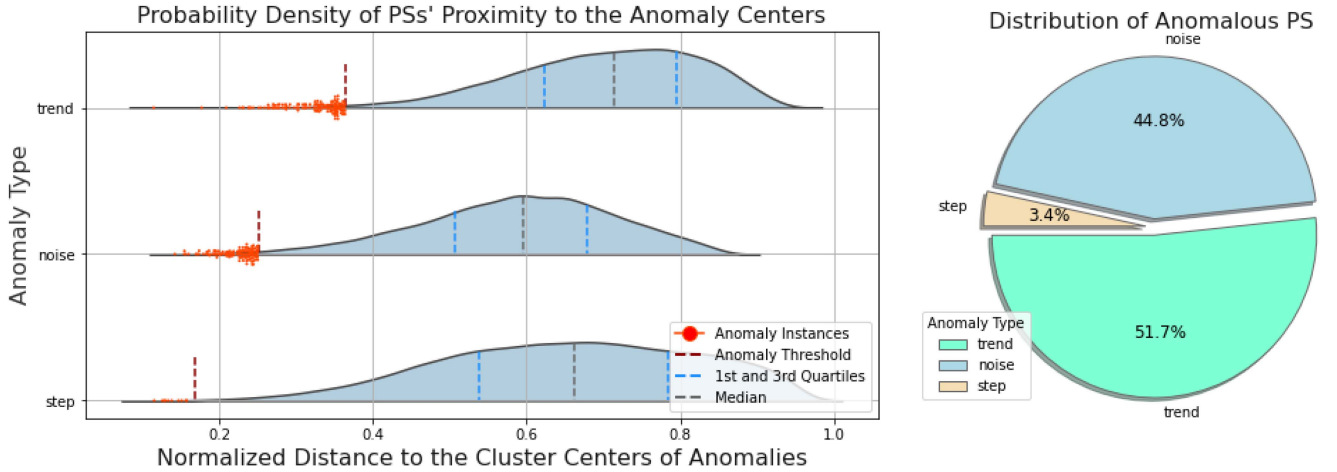


Fig. 4. Left: The probability densities of PSs' proximity to the cluster centres of synthetic anomalies. Right: The distribution of anomaly instances per type.

highest, while that of the *step* anomaly is the lowest. Therefore, if a PS instance is predicted as an anomaly, it is more likely due to a *trend* anomaly indicating gradual changes over time, rather than a *step* anomaly indicating sudden and significant changes. This information is particularly relevant given that no major earthquakes causing lateral movement occurred in the region of interest between 2015–2020. Besides, the skewness of the proximity distributions is more prominent towards the cluster centre of *trend* anomaly, while it is more symmetrical towards the centre of *noise* anomaly. This observation suggests that *noise* may be a common phenomenon among all observations, but its magnitude could be a distinguishing factor in determining whether the observed noise is due to an anomaly or not.

IV. METHODOLOGY

This part provides a brief explanation of the benchmark approaches used for comparisons in Section IV-A, and presents the details of our proposed approach in Section IV-B.

A. Benchmark Methods

We chose four different anomaly detection methods for our comparison, including two classical ML methods and two DL methods. Our selection of these methods was based on several criteria, such as their simplicity, suitability for detecting anomalies in data streams, and their success in various real-world anomaly detection scenarios. The methods included in the evaluation are as follows.

- 1) maximally divergent intervals (MDI);
- 2) robust random cut forest (RRCF);
- 3) autoencoders;
- 4) graph augmented normalizing flows (GANF), which are further explained below:

1) *Maximally Divergent Intervals*: MDI is a method for anomaly detection in time-series data that aims to find the time interval where the difference between two related series is the greatest [27]. It works by comparing the *Kullback–Leibler divergence* between the current time-series and a reference series and identifies the time interval where the divergence is maximal. In other words, given a multivariate time-series \mathcal{T} and its

subsequences \mathcal{S} where $\mathcal{S} \subseteq \mathcal{T}$, the MDI finds the most anomalous subsequence $\tilde{\mathcal{S}}$ by solving the underlying optimization problem:

$$\tilde{\mathcal{S}} := \arg \max_{\mathcal{S} \subseteq \mathcal{T}} \mathcal{D}(p_{\mathcal{S}}, p_{\Omega(\mathcal{S})}) \quad (2)$$

where $p_{\mathcal{S}}$ is the probability density of a subsequence \mathcal{S} , and p_{Ω} is the probability density of the remaining part of the time-series $\Omega(\mathcal{S}) := \mathcal{T} \setminus \mathcal{S}$.

The main competence of MDI is its ability to handle a variety of types of anomalies, including point anomalies, collective anomalies, and contextual anomalies, making it a versatile tool for anomaly detection in various applications. Besides, MDI can also handle data with missing values, making it applicable to real-world scenarios where data may not always be complete.

2) *Robust Random Cut Forest*: RRCF is an algorithm that builds a collection of binary trees to detect anomalies based on their isolation scores, as a modification of the well-known Isolation Forest [28]. The RRCF is combined with a windowing approach to process data streams in mini-batches, reducing computational costs while maintaining detection accuracy. It has an advantage over Isolation Forest and RCF (Random Cut Forest) due to its anomaly scoring function, *Collusive Displacement*. This function makes RRCF robust to the presence of duplicates or near-duplicates that could otherwise mask the detection of outliers. Collusive Displacement accounts for duplicates or near-duplicates by removing a subset of “colluders” alongside the point of interest. However, for the detection of anomalous subsequences, an additional preprocessing step for constructing window-based features could be considered. RRCF's simplicity and comprehensibility make it an attractive choice for anomaly detection.

3) *Autoencoders*: Autoencoders are a type of neural network that learns to reconstruct its input data, often used for unsupervised feature learning and dimensionality reduction [29]. Let $E: \mathbb{R}^n \mapsto \mathbb{R}^l$ and $D: \mathbb{R}^l \mapsto \mathbb{R}^n$ be the encoder and decoder networks, respectively, where $l, n \in \mathbb{N}$ and $l < n$. Thus, the encoder is responsible for producing hidden code $\mathbf{z} = E(\mathbf{x})$, and the decoder for constructing the output $\mathbf{y} = D(\mathbf{z})$. Here, minimization of the reconstruction loss \mathcal{L} is the main learning

objective of autoencoders to achieve $y \approx x$, such that

$$\arg \min_{\beta, \theta} \mathcal{L}(y, D_{\beta}(E_{\theta}(x))) \quad (3)$$

where β and θ are the model parameters and $\arg \min_{\beta, \theta}$ represents the value of the parameters that minimize the loss.

In the context of unsupervised anomaly detection, the autoencoder learns a normal profile of the time-series \mathcal{T} and detects anomalous input sequences $\tilde{\mathcal{S}}$ with a high reconstruction error [30]. Our experiments involved the use of a dense autoencoder with two hidden layers in both the encoder and decoder, with each layer having twice the number of neurons in the latent space and ReLU activations as nonlinearities.

4) *Graph Augmented Normalizing Flows*: GANF is a deep learning-based method for anomaly detection that combines normalizing flows with a graph neural network (GNN) to model the normal behavior of time-series data [31]. It first converts the multivariate time-series into a graph structure where each node represents a data point and the edges indicate the temporal order of the data points. The GNN then learns embeddings of the nodes, which are used as inputs to a normalizing flow model that estimates the probability distribution of the normal behavior of the time-series. During inference, the probability of a new data point is computed using the learned distribution, and if it is below a threshold, the point is classified as an anomaly, based on the assumption that anomalous data points typically have low probability densities, allowing the estimated densities to be used as an anomaly score.

One advantage of GANF is its ability to detect anomalies in sequences with nonlinear temporal dependencies and complex patterns. The use of normalizing flows allows for efficient and accurate density estimation, making it possible to scale the approach to large datasets and high-dimensional data.

B. Proposed Approach

In our approach, the input sequence is learned to be reconstructed by the LSTM-autoencoder model after being randomly permuted. The assumption behind this is that if there is any anomaly in some time-series, the model would not be able to arrange the corresponding input sequence in the correct order [32]. Furthermore, our approach is distinguished by the utilization of the Hann window for smoothing the time-series [33] which results in improved performance for detecting certain anomaly types, and the employment of a custom loss function derived from the *soft-DTW* [34], which is more suitable for time-series. The novel aspect of incorporating the *soft-DTW* distance metric as a loss function is its capability to handle similar patterns occurring at different time steps, acknowledging that reconstructing the exact time-series pattern may not be feasible when anomalies are present, particularly when patterns hold more significance than their specific occurrence. By integrating these elements as also shown in Fig. 5, our approach demonstrates enhanced performance in the detection of building anomalies.

The following steps are taken in the proposed approach.

In the training phase

- 1) Normalize the PS instances to zero mean - unit variance.
- 2) Apply Hann windowing to smooth the input instances.
- 3) Randomly permute time stamps of the windowed inputs.

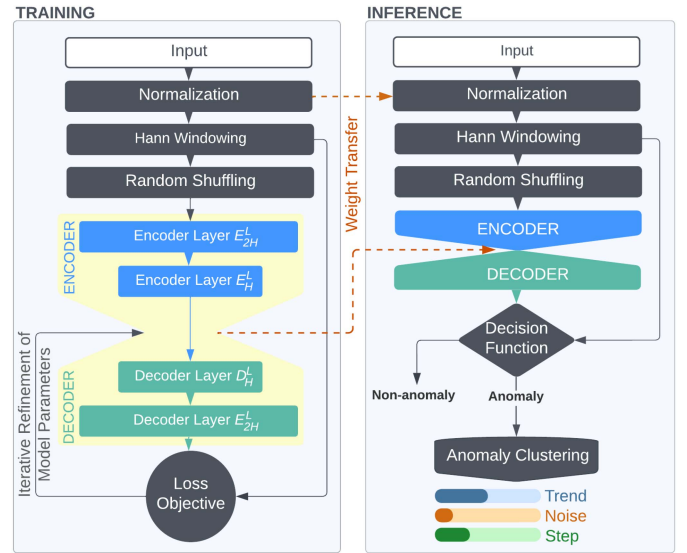


Fig. 5. Training and Inference pipelines based on LSTM autoencoders.

- 4) Then, train the LSTM autoencoder with the windowed-permuted inputs.
- 5) Minimize the objective function during the training to reconstruct the nonpermuted but windowed input, i.e., the smoothed PS instance with timely-ordered PS points.

In the inference phase

- 1) Estimate the reconstruction loss for a PS instance and later set it as an anomaly if the loss is greater than a predetermined threshold τ .
- 2) If an anomalous PS instance is detected, estimate its distance to the cluster centres of each anomaly using (1).
- 3) Assign the type of detected anomaly for the PS instance based on the proximity to the cluster centres.

If the preprocessing, training, and inference steps are detailed

1) *Preprocessing*: Given a dataset $X_{tr} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M\}$ consisting of M number of PS instances reserved for training where each instance $\mathbf{x} = \{x_1, x_2, \dots, x_T\}$ with T time points, then the standardized instance \mathbf{x}' can be obtained as

$$\mathbf{x}' = \frac{\mathbf{x} - \mu}{\sigma} \quad (4)$$

where μ is the mean, σ is the standard deviation of X along the dimension M . Then, considering the Hann window is

$$a(n) = 0.5 - 0.5 \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \leq n \leq N \quad (5)$$

where n is the index of the sample, N is the length of the window, and $a(n)$ is the attenuation factor applied to the sample. Thus, \mathbf{x}' is smoothed by convolving a over \mathbf{x}'

$$\mathbf{x}'_a(n) = \mathbf{x}'(n) * a(n) = \sum_{m=-\infty}^{\infty} \mathbf{x}'[m] \cdot a[n-m]. \quad (6)$$

Finally, the time points of a smoothed PS instance \mathbf{x}'_a can be permuted with random shuffling operation R as

$$\tilde{\mathbf{x}}'_a = R(\mathbf{x}'_a), \quad \tilde{\mathbf{x}}'_a \forall (x'_a \in \mathbf{x}'_a). \quad (7)$$

2) *Training LSTM Autoencoder*: LSTM is a type of RNN architecture used for processing sequential data, particularly suited for tasks involving long-term dependencies. An LSTM cell processes one time step at a time. Thus, the entire set of computations performed by an LSTM cell can be expressed using the following equations [29]:

$$\begin{aligned}
i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \\
f_t &= \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \\
g_t &= \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \\
o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\
h_t &= o_t \odot \tanh(c_t)
\end{aligned} \tag{8}$$

where x_t is the input at time step t , h_{t-1} is the hidden state of the previous time step, i_t , f_t , g_t , o_t , c_t , and h_t are the input gate, forget gate, cell input, output gate, cell state, and hidden state at time step t , respectively. The operator σ is the sigmoid function, \odot is the element-wise multiplication, and \tanh is the hyperbolic tangent function. The weights and biases are denoted by W and b , respectively.

Hence, utilizing the encoder and decoder blocks defined in Section IV-A3 and considering (8), we can construct an LSTM encoder E^L and an LSTM decoder D^L as

$$\begin{aligned}
E^L &:= L_H^T \circ L_{2H}^T : \tilde{\mathbf{x}}'_a \mapsto \mathbf{z} \\
D^L &:= L_{2H}^T \circ L_H^T : \mathbf{z} \mapsto \mathbf{y}.
\end{aligned} \tag{9}$$

Here, L refers to an LSTM layer, T represents the number of LSTM cells in the given layer, and H denotes the number of hidden dimensions in each LSTM cell.

The goal of the LSTM autoencoder network is to learn the reconstruction of the original time-series \mathbf{x}'_a using the permuted sequence $\tilde{\mathbf{x}}'_a$ as input, with the aim of obtaining $\mathbf{y} \approx \mathbf{x}'_a$. To achieve this objective, the model is trained to minimize the reconstruction loss, as expressed by the following equation:

$$\arg \min_{\beta, \theta} \text{soft-DTW}^\gamma(\mathbf{y}, \mathbf{x}'_a), \quad \mathbf{y} = D_\beta^L(E_\theta^L(\tilde{\mathbf{x}}'_a)). \tag{10}$$

Here, β and θ denote the decoder and encoder model parameters, respectively. The *soft-DTW* loss function is used to compute the reconstruction loss, where the degree of softness in the function is controlled by the parameter γ .

3) *Inference*: Assuming that a PS instance from the test dataset, denoted as $\mathbf{x} \in X_{te}$, undergoes normalization, windowing, and permutation as presented in (4), (6), and (7), respectively, resulting in \mathbf{x}' , \mathbf{x}'_a , and $\tilde{\mathbf{x}}'_a$; \mathbf{y} is the reconstructed time-series by the autoencoder as presented in (10), then the decision function D can be expressed as

$$D(\mathbf{x}) := \begin{cases} \text{anomaly} & \text{if } \text{soft-DTW}^\gamma(\mathbf{y}, \mathbf{x}'_a) \geq \tau \\ \text{nonanomaly} & \text{otherwise.} \end{cases} \tag{11}$$

Here, the threshold for the decision function is denoted by τ . Thus, let $B = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$ be a building with m PS

TABLE II
HYPERPARAMETER SEARCH SPACE

Training Parameters		Architectural Parameters	
Training epochs	90	Hann window size	[1 – 13]
Batch size	1024	LSTM hidden dimensions	[1 – 64]
ADAM learning rate	[0.00001 – 0.01]	Number of encoding layers	[1 – 6]
Regularization rate (λ)	[0.00001 – 0.01]	Number of decoding layers	[1 – 6]
<i>Soft-DTW</i> parameter (γ)	[0.1 – 10]	Network initialization	<i>Xavier Uniform</i>

instances, then

$$D(B) := \begin{cases} \text{anomaly} & \text{if } D(\mathbf{x}_i) = \text{anomaly for } \exists \mathbf{x}_i \in B \\ \text{nonanomaly} & \text{otherwise.} \end{cases} \tag{12}$$

Let the support score S represent the degree of confidence in a decision made about a building B . The support score can be determined by calculating the ratio of the anomalous PS instances to the total number of PS instances in the building. Mathematically, this can be expressed as

$$S(B) := \frac{\sum_{i=1}^m [D(\mathbf{x}_i) = \text{anomaly}]}{m}. \tag{13}$$

Furthermore, when a PS instance \mathbf{x} is detected as anomalous, its corresponding anomaly type is determined based on its closeness to the cluster centres. Thus, the class assignment is employed using the following equation:

$$\arg \min_{c \in \{1, \dots, K\}} \text{soft-DTW}^\gamma(\mathbf{x}, \mathbf{m}_c). \tag{14}$$

Here, K denotes the number of clusters and \mathbf{m}_c refers to the centre of the c th cluster obtained by performing k -means clustering, as already detailed in Section III-C.

V. EXPERIMENTAL FRAMEWORK

This part of the document outlines the methods used to experiment with various learning techniques and measurements to confirm the effectiveness of our approach.

A. Network Initialization and Optimization

The deep learning architectures utilized in this study were trained using adaptive moment estimation (ADAM) [35] with a batch size of 1024 and weight decay regularization [36]. The learning rate and weight decay λ were initialized in the range [0.00001 – 0.01] during ADAM optimization, and the learning rate was halved once a plateau was reached. The maximum number of training epochs was set to 90, with the learning process halted when the validation loss indicated appropriate convergence. Weight initialization was performed using *Xavier Uniform* [37]. The size of the Hann window was searched within odd numbers between [1 – 13]. For the LSTM autoencoder, the number of hidden dimensions for LSTM cells was sought between the values [1 – 64] with power-of-2 increment steps, while the total number of stacked encoding and decoding layers was sought between [1 – 6]. During optimization, L_1 loss and *soft-DTW* objective functions were compared. For *soft-DTW* based optimization, the value of γ was sought in the range [0.1 – 10]. To minimize loss, a Bayesian hyperparameter search was conducted on the parameter ranges specified in Table II.

TABLE III
COMPARISON OF BENCHMARK MODELS: BUILDING ANOMALY DETECTION ACCURACY FOR DIFFERENT PERCENTILES OF PRESUMED ANOMALY

Benchmark model	Input data permutation	Smoothing window size	Anomalous building detection accuracy (%) per displacement scenario											
			τ percentile @75				τ percentile @90				τ percentile @98			
			Trend	Noise	Step	Avg	Trend	Noise	Step	Avg	Trend	Noise	Step	Avg
MDI	No	7	96.31	16.42	59.95	57.26	88.50	06.10	34.27	42.81	62.88	00.80	08.60	24.81
RRCF			14.01	36.81	22.66	24.59	02.65	14.57	07.63	08.32	00.00	01.12	01.79	00.94
GANF			34.78	35.14	57.68	42.40	14.60	14.53	32.20	20.34	02.84	02.07	12.01	05.39
Autoencoders			00.44	53.87	05.30	20.18	00.12	23.97	00.92	08.46	00.00	06.06	00.54	02.26
LSTM autoencoders			82.07	40.33	22.49	48.37	66.64	18.06	07.93	30.94	39.65	03.99	01.61	15.67
LSTM autoencoders	Yes		99.05	24.18	59.17	60.54	96.90	08.31	40.29	48.33	93.20	01.11	15.95	37.73

TABLE IV
COMPARED MODELS AND THEIR BRIEF SPECIFICATIONS

Benchmark model	Category	Learning mechanism	Anomaly score
MDI [27]	ML	Density estimation	KL divergence
RRCF [28]	ML	Isolation forest	Collusive displacement
GANF [31]	DL	Density estimation	Density
Autoencoders [29]	DL	Reconstruction	L_1 , <i>soft-DTW</i>
LSTM autoencoders	DL	Reconstruction	L_1 , <i>soft-DTW</i>

B. Software Specifications

The experimental protocols were implemented using version 1.10 of the PYTOCH framework [38]. In addition, the following Python libraries were utilized for numerical processing: NUMPY, SCIPY, PANDAS, and SCIKIT-LEARN. Bayesian optimization was performed using OPTUNA. To ensure the reproducibility of the machine learning pipeline for interested parties, the software dependencies were included in a DOCKER container, which can be accessed through the project source code.

C. Evaluation Metrics

During our experiments, we used a training and validation dataset that contained only real-world measurements, with a split ratio of 75:25. The test dataset was a combination of real-world PS observations and synthetic PS observations created using data augmentation techniques discussed in Section III-B. To assess the performance of our model at different levels of anomaly detection, we varied the ratio of raw to synthetic observations in the test dataset between 75:25 and 99:1. We also adjusted the anomaly threshold τ for each percentile increment, increasing it from 75% to 99%, which corresponded to a decrease in the assumed percentile of anomalies in the evaluation dataset from 25% to 1%. By doing this, we were able to calculate the accuracy of anomaly detection at each τ increment using the following formula:

$$\text{Accuracy}_{\tau@k} = \frac{S_{\tau@k}}{S_k} \quad (15)$$

where $S_{\tau@k}$ represents the number of correctly detected synthetic PS anomalies when the anomaly threshold $\tau = k \in [75 - 99]$, and S_k denotes the total number of synthetic PS anomalies added into the test dataset with a ratio of $k : 100 - k$ between real-world and synthetic observations.

VI. RESULTS AND DISCUSSION

This part presents a benchmark comparison in Section VI-A, conducts an ablation study on our proposed approach in Section VI-B, and provides additional qualitative comparisons based on detected anomalies in Section VI-C.

A. Benchmark Comparisons

Our comparative analysis involved four established benchmark methodologies, as well as our proposed LSTM autoencoder, as outlined in Table IV. While Autoencoder-based anomaly detection methods typically rely on the L_1 or *MSE* distance metrics in the literature, we additionally compared its performance to that of the *soft-DTW* metric, to ensure a fair comparison with our proposed LSTM-autoencoder approach. Table III and Fig. 6 show that when the smoothing size is fixed at 7 and input time-series permutation is avoided, MDI and our proposed LSTM autoencoders perform better than other methods in terms of overall accuracy (Avg). However, if input permutation is applied, our proposed method significantly outperforms the others.

For the detection of *trend* anomalies, our proposed approach consistently outperforms all other benchmark methods across various presumed percentiles of buildings with anomalies, as depicted in Fig. 6. Notably, even with only 2% of buildings exhibiting anomalies, our approach achieves an accuracy of 93.20% in distinguishing these anomalies from the remaining buildings. These results underscore the effectiveness of our approach, particularly in accurately predicting *trend* anomalies.

On the other hand, it is important to note that autoencoders achieve the highest accuracy rate only for *noise* anomaly. Besides, in terms of accuracy, there is no significant difference between the use of L_1 loss and *soft-DTW* loss over autoencoders, which is why only the latter one is exploited in the benchmark comparisons, but loss selection is important for LSTM autoencoders, which will be discussed further in Section VI-B. Furthermore, the LSTM autoencoder emerged as the second most effective approach for detecting *noise* anomalies, but this was achieved without incorporating input permutation.

Regarding *step* anomaly, if the presumed anomalous instance rate is high in the evaluation dataset (i.e., $\tau@75$ meaning that 25% of data instances have anomaly), MDI is the best approach. Nevertheless, in scenarios where the presumed occurrence of anomaly instances is lowered throughout the dataset ($\tau > 80$),

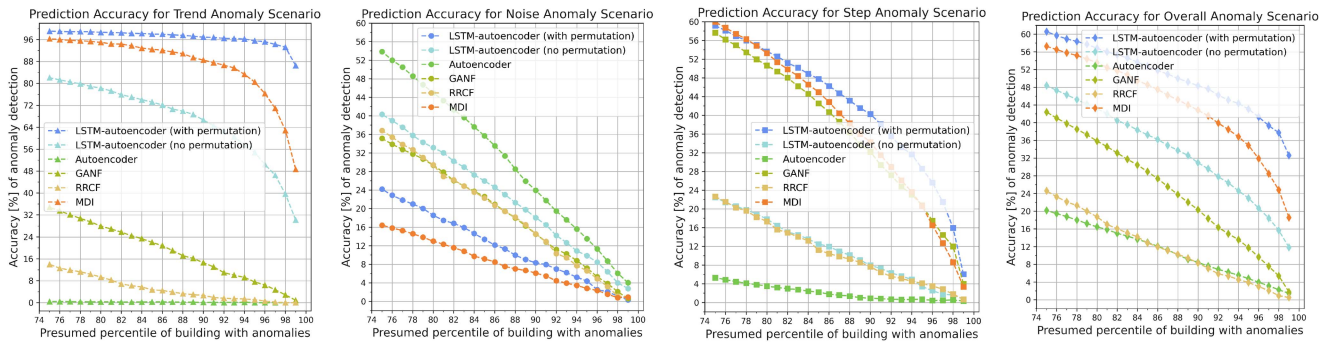


Fig. 6. Comparison of models: accuracy for different percentiles of anomaly (hidden embedding dimension $H = 32$, and Hann window size $n = 7$).

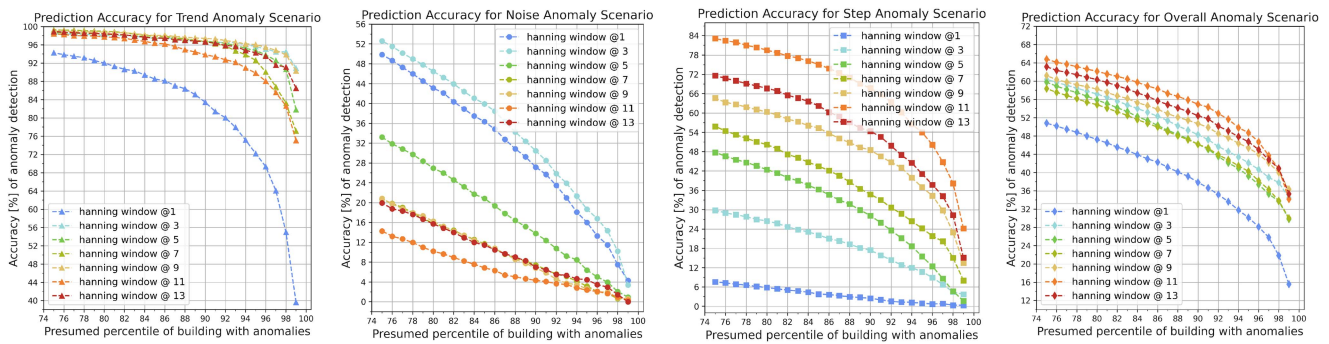


Fig. 7. Comparing the accuracy of LSTM autoencoder for different Hann window sizes in input smoothing (hidden embedding dimension $H = 32$).

which aligns more closely with real-world conditions, the proposed LSTM autoencoder outperforms MDI in identifying *step* anomalies.

In summary, when considering the overall accuracy across all anomaly types, our proposed approach consistently demonstrates the highest performance across all percentiles of anomalies, followed by the MDI approach. Furthermore, when comparing individual anomaly types, our proposed approach excels particularly in detecting building anomalies that manifest as patterns in the time-series, such as *trend* and *step* anomalies. On the other hand, the standard autoencoder outperforms in detecting anomalies where the deviation is primarily in magnitude rather than exhibiting a specific pattern, as observed in *noise* anomalies.

B. Ablation Study for the Proposed Approach

In this section, we comprehensively analyze the impact of preprocessing techniques, model attributes, and reconstruction losses on our proposed anomaly detection pipeline.

1) *Smoothing Input PS Time-Series*: Hann windowing, illustrated in Fig. 7, has a beneficial impact on detecting displacements, especially *trend* and *step*, by suppressing noise. Previous studies have also demonstrated the effectiveness of such techniques by decomposing signals and eliminating periodic and noisy components [23]. However, the detection of displacements such as vibration, expansion, and contraction of buildings, referred to as *noise* in our dataset, might be negatively

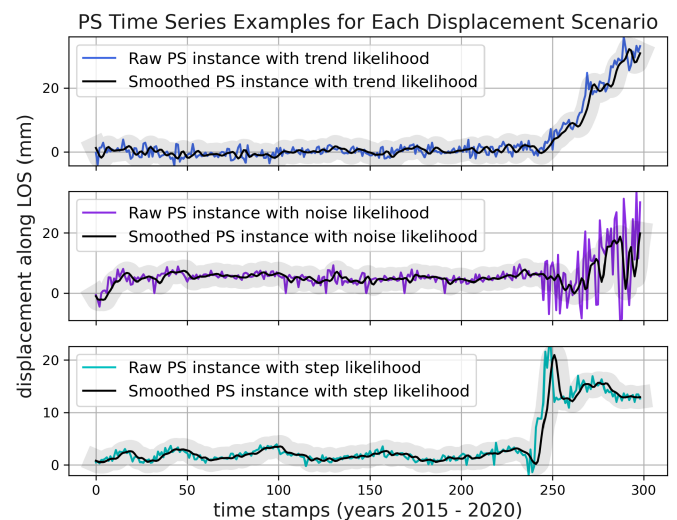


Fig. 8. Comparing the impact of applying the Hann window to PS time-series. The plots at the top and bottom demonstrate that windowing with a size of $n = 13$ retains the slope and vertical shifts associated with *trend* and *step* anomalies. However, the middle plot shows that windowing penalizes high-frequency components associated with *noise* anomalies, leading to a decrease in performance in this category.

affected by smoothing. This is because the distinguishing feature of these anomalies is the sharp changes in LOS magnitude, as discussed in Section III-C. Although the Hann window does not

TABLE V
A COMPARATIVE ANALYSIS OF THE IMPACT OF PREPROCESSING AND MODEL TRAINING PARAMETERS ON ANOMALY DETECTION

Preprocessing and model training parameters				Anomalous building detection accuracy (%) per displacement scenario								
Input data permutation	Smoothing window size	Hidden embedding dimension	Reconstruction loss function	τ percentile @75			τ percentile @90			τ percentile @98		
				Trend	Noise	Step	Trend	Noise	Step	Trend	Noise	Step
Yes	1	1	<i>soft-DTW</i>	96.14	46.45	28.59	89.93	26.33	12.07	65.25	06.06	02.15
No	1	1	<i>soft-DTW</i>	43.08	40.89	20.48	38.34	19.87	18.46	34.76	02.07	15.23
Yes	13	1	<i>soft-DTW</i>	35.54	26.79	27.45	30.83	11.08	20.00	23.85	02.23	12.01
No	13	1	<i>soft-DTW</i>	71.43	26.88	42.55	59.38	11.80	27.55	41.55	03.51	07.35
Yes	1	64	<i>soft-DTW</i>	93.22	51.22	07.83	81.65	29.87	02.49	56.24	10.37	00.89
No	1	64	<i>soft-DTW</i>	92.70	45.06	17.43	78.18	23.65	06.64	41.55	05.42	01.08
Yes	13	64	<i>soft-DTW</i>	99.53	16.30	73.69	96.85	06.26	50.50	90.21	01.60	23.12
No	13	64	<i>soft-DTW</i>	33.33	39.75	15.96	10.48	17.95	02.86	00.32	03.19	00.36
Yes	1	1	L_1	06.80	24.57	42.69	02.94	10.03	18.87	00.47	02.87	02.15
No	1	1	L_1	05.85	23.77	36.31	02.57	07.63	16.64	00.63	01.75	01.25
Yes	13	1	L_1	07.50	30.42	28.50	03.22	11.88	10.17	00.16	02.55	00.90
No	13	1	L_1	12.05	30.35	28.29	07.18	11.56	13.28	02.69	02.87	03.58
Yes	1	64	L_1	21.57	25.97	55.67	10.73	12.57	33.44	01.74	02.87	07.71
No	1	64	L_1	00.00	48.26	02.20	00.00	19.67	00.17	00.00	04.63	00.00
Yes	13	64	L_1	36.08	26.49	52.19	17.21	12.04	28.76	01.26	03.03	04.30
No	13	64	L_1	08.81	31.55	22.72	01.96	12.36	08.17	00.00	00.80	02.33
Yes	5	32	<i>soft-DTW</i>	99.31	30.59	50.38	97.84	12.69	30.41	94.15	02.07	06.45
No	5	32	<i>soft-DTW</i>	57.25	46.43	13.00	34.58	24.73	04.36	15.01	07.34	00.90
Yes	5	32	L_1	23.70	27.94	52.96	11.09	12.73	29.05	01.74	02.87	05.02
No	5	32	L_1	04.05	31.33	22.27	00.77	10.20	08.71	00.00	01.75	01.25
Yes	7	32	<i>soft-DTW</i>	99.05	24.18	59.17	96.90	08.31	40.29	93.20	01.11	15.95
No	7	32	<i>soft-DTW</i>	82.07	40.33	22.49	66.64	18.06	07.93	39.65	03.99	01.61
Yes	7	32	L_1	33.45	28.94	52.95	16.39	11.84	28.42	01.58	02.71	04.84
No	7	32	L_1	17.74	34.78	15.66	11.95	11.60	05.81	07.74	01.91	01.61

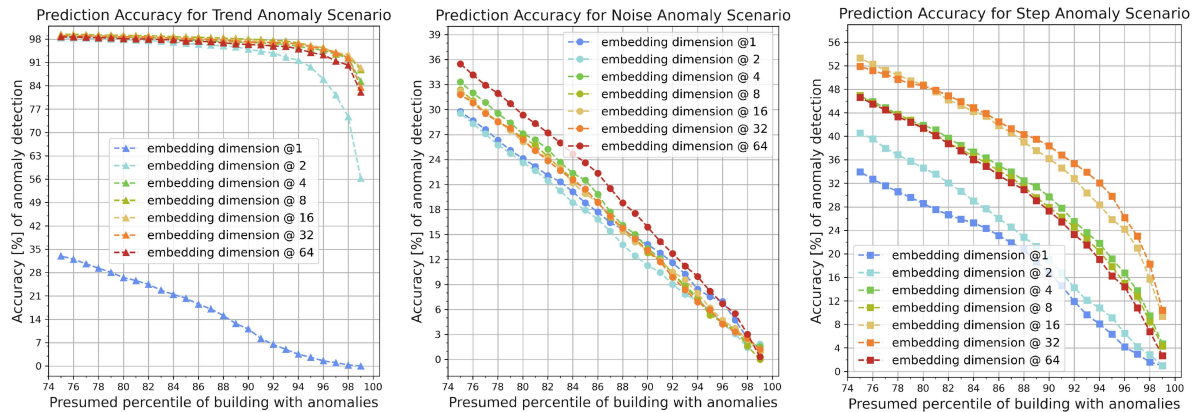


Fig. 9. Comparison of anomaly detection accuracy per each displacement scenario with different embedding sizes in the encoder (Hann window size $n = 5$).

heavily penalize high-frequency components in comparison to average windowing, it still removes important characteristics of *noise* anomalies, as illustrated in Fig. 8.

It is noteworthy that smoothing can enhance the performance of anomaly detection; however, its effectiveness may vary depending on the type of anomaly being detected. Therefore, it is essential to take into account the data characteristics and the anomalies being targeted when deciding whether to implement smoothing as a preprocessing step.

2) *Random Permutation of Input PS Time Series*: By shuffling the order of PS time-series and learning to reconstruct their original order, the accuracy of anomaly detection significantly improved for all types of anomalies from an accuracy of 15.67% to 37.73% as shown in the left and middle

plots of Fig. 10. When Hann window size $n = 7$ and embedding dimension $H = 32$, it can be observed that with a threshold $\tau@98\%$, the accuracy of detecting *trend* anomalies increased from 39.65% to 93.20%. Similar performance improvements were also observed for *step* anomalies, which increased from 1.61% to 15.95%. However, the permutation caused a drop in performance for *noise* anomalies, even though the overall performance increased. Nevertheless, when smoothing was not applied (Hann window size $n = 1$) and *soft-DTW* was used as detailed in Table V, the permutation did not cause any performance drop for the *noise* anomaly.

3) *Impact of Embedding Dimension and Layer Numbers*: Increasing the number of hidden dimensions as embedding size

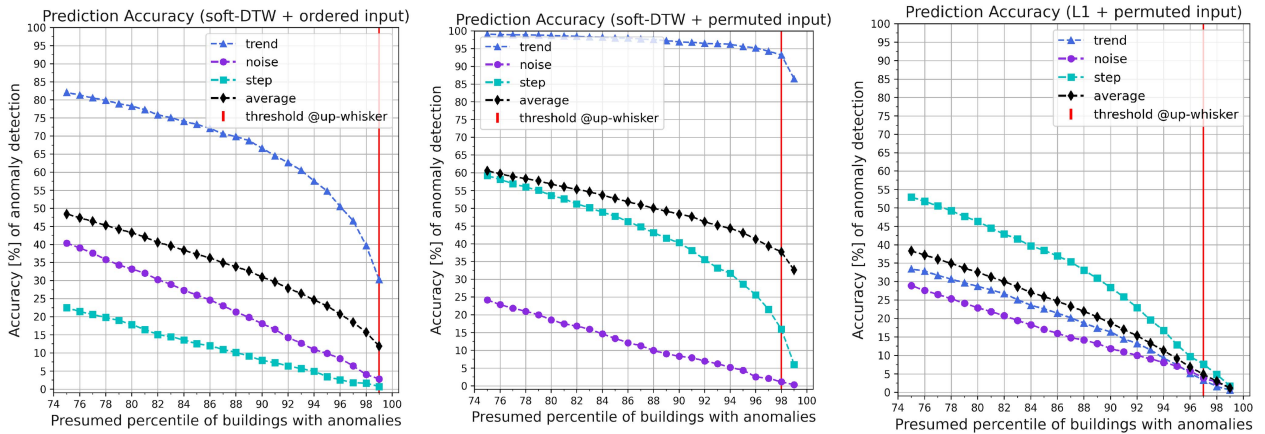


Fig. 10. Comparison when the input is either permuted or not, and when using L_1 or DTW (Hann window size $n = 7$ and embedding dimension $H = 32$).

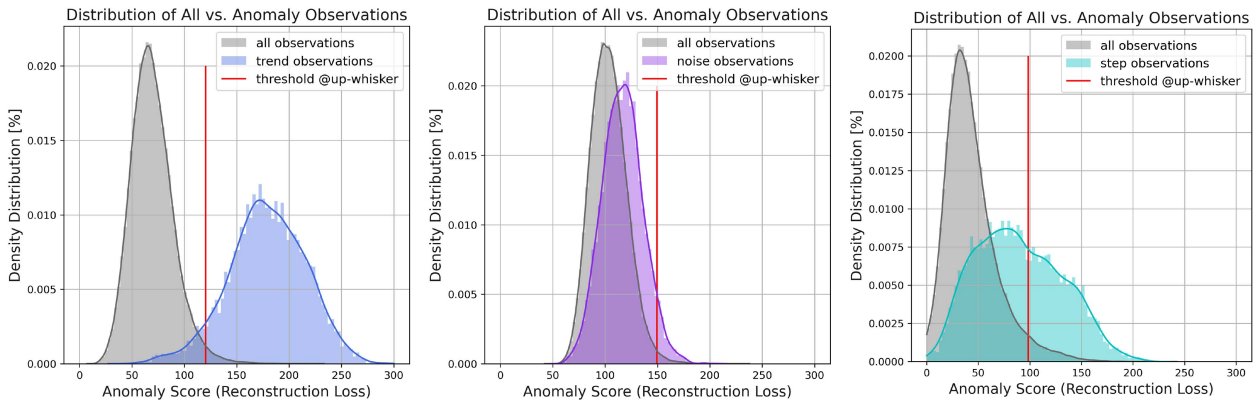


Fig. 11. Comparing the density distribution of anomaly scores between all observations and anomalous observations.

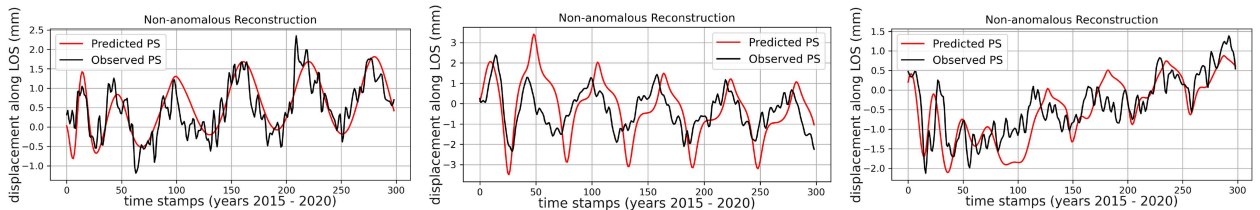


Fig. 12. Comparison of PS instances with low reconstruction loss (nonanomalies) through observed and predicted time-series.

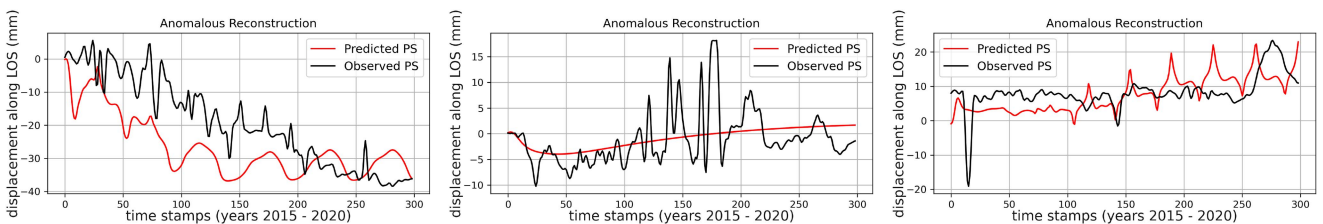


Fig. 13. Comparison of PS instances with high reconstruction loss (anomalies) through observed and predicted time-series.



Fig. 14. Detected anomalies when the upper whisker threshold: 98%. Each location pin shows a displaced building with at least five anomalous PS instances. The heatmap shows the anomalous PS density per area.

in the LSTM autoencoder results in improved accuracy for each anomaly type, as shown in Fig. 9. However, there is no significant improvement in anomaly detection for an embedding dimension greater than 32. Additionally, the depth of the encoder and decoder layers was examined, but it was found that having more than two layers did not lead to any improvements.

4) *Choice of Loss Function*: The middle and right plots in Fig. 10 illustrate that there is a considerable enhancement in

detecting all types of anomalies when switching from the L_1 loss to the *soft-DTW* loss for the reconstruction. The improvement is particularly noticeable for *trend* anomalies. This pattern is consistent across multiple experiments with different parameter settings, as indicated in Table V, where the best accuracy of each column is shown in bold.

The use of *soft-DTW* as the reconstruction loss metric could potentially explain the increased accuracy in anomaly detection. Unlike the L_1 loss, which focuses only on the magnitude differences between the predicted and true values, *soft-DTW* considers the shape of the time-series by measuring the distance between their features, thereby preserving temporal information.

5) *Density Distributions of Anomaly Scores*: The density distribution in Fig. 11 compares the scores of all observations to those of anomalous observations. Each plot corresponds to the model parameters with the highest accuracy for the given anomalous instance rate at $\tau@75$ in Table V. The results indicate that the observations with *trend* anomaly are the most distinguishable from the remaining observations. This finding can help explain the high accuracy achieved in detecting this type of displacement. Lowering the threshold can detect all anomalies but at the cost of high false positives, or vice versa. In contrast, the distribution of *noise* anomaly has the worst separable pattern, as shown in the middle plot. The anomalous and normal observations share the same pattern except for a slight shift in the horizontal axis, which is consistent with the low accuracy observed in detecting this type of anomaly.

C. Qualitative Evaluation on Detected Anomalies

Our approach trains the LSTM autoencoder to reconstruct the input sequence after a random permutation. The idea is that the model would fail to properly arrange the input sequence if an anomaly is present in the time-series [32]. We demonstrate this idea through examples of PS time-series with low reconstruction loss, corresponding to nonanomalous instances, in Fig. 12, and examples with high reconstruction loss, corresponding to *trend*, *noise*, and *step* anomalies, in Fig. 13. Successful reconstruction results in predicted PS that follow the periodicity and slope of the observed signals with some shift and a lack of high-frequency components. Failed reconstruction, on the other hand, leads to predicted and observed signals with different patterns.

Apart from that, Fig. 14 shows the locations of anomalous buildings in Rome with a support score $S > 10\%$ and at least five anomalous PS instances using coloured pins. The heatmap depicts the density of detected anomalous PS points, regardless of their support score. Furthermore, Figs. 15–20 provide aerial and street view evaluations of some detected anomalies. For further evaluation examples, the interested audiences can explore the interactive map provided.¹

¹[Online]. Available: <https://maps.co/map/63fc2919d50c00331050310cd4dd22a>



Fig. 15. Farm house with the detected anomaly of *trend*. Upon inspection of both aerial and street views, it was observed that the building is old and poorly maintained, which could possibly account for the detected displacement.

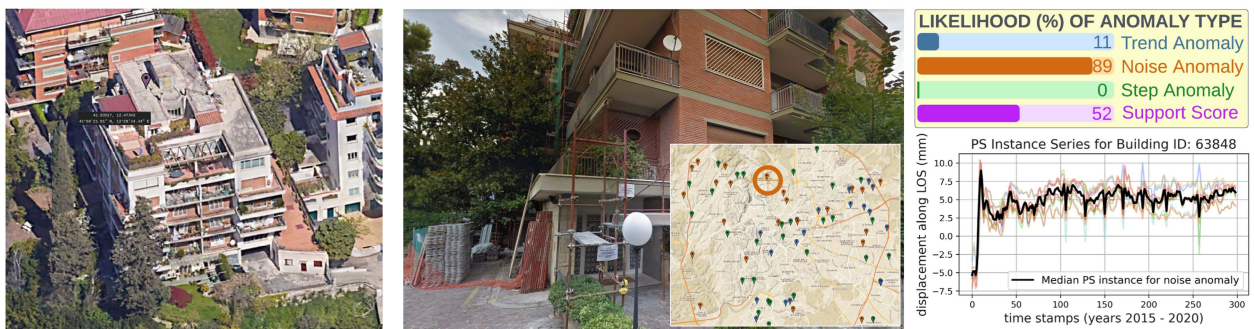


Fig. 16. Apartment with the detected anomaly of *noise*. Upon conducting an empirical evaluation using street views, it appears that the apartment has undergone some maintenance activities which could have caused changes to the facade and roof, and may explain the presence of the detected displacement.



Fig. 17. Building with the detected anomaly of *step*. Empirical evaluation based on aerial views shows that the building is next to a subway construction site which might be a reason for the observed displacement.



Fig. 18. Business center with the detected anomaly of *step*. Based on the aerial and street views, the center is newly built and there are ongoing landscape construction activities surrounding it, which could potentially explain the detected “displacement” in the area.



Fig. 19. Anomaly detected for a museum is classified as *step*. However, the corresponding *support score* is only 13%, indicating that the prediction is not trustworthy enough to claim an anomaly. Still, a further examination might be required since the museum comprises multiple building complexes.

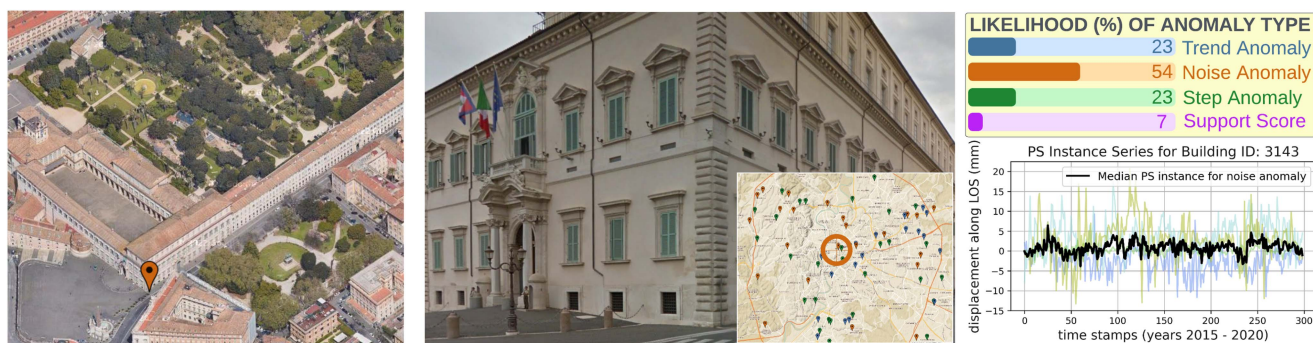


Fig. 20. Anomaly detected for the presidential palace is classified as *noise*. However, the corresponding *support score* is only 7%, indicating that the prediction is not trustworthy enough to claim an anomaly. Still, a further examination might be required since the palace comprises multiple building complexes.

VII. CONCLUSION

This study presented a new method for detecting anomalies in Persistent Scatterer (PS) time-series data related to building displacements. The approach utilized an LSTM-autoencoder model and a special time-series reconstruction loss function called Soft-DTW. Moreover, windowing and permutation techniques on input time-series were employed to enhance anomaly detection performance. Experiments conducted on real PS data from Rome, Italy, revealed that the proposed method outperformed conventional methods in detecting various types of building displacements. Qualitative evaluations also confirmed its effectiveness in identifying potentially anomalous buildings in Rome. Thus, this approach shows great promise for detecting anomalies in PS time-series data, and its importance is further amplified through the utilization of the recently established European Ground Motion Service, demonstrating the potential of the service for AI-based applications in scalable urban monitoring and infrastructure management at the continental level.

REFERENCES

- [1] F. Foroughnia, S. Nemati, Y. Maghsoudi, and D. Perissin, "An iterative PS-InSAR method for the analysis of large spatio-temporal baseline data stacks for land subsidence estimation," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 74, pp. 248–258, 2019.
- [2] R. Lanari, P. Lundgren, M. Manzo, and F. Casu, "Satellite radar interferometry time series analysis of surface deformation for Los Angeles, California," *Geophys. Res. Lett.*, vol. 31, no. 23, pp. 1–5, 2004.
- [3] P. F. Giordano, G. Miraglia, E. Lenticchia, R. Ceravolo, and M. P. Limongelli, "Satellite interferometric data for seismic damage assessment," *Procedia Struct. Integrity*, vol. 44, pp. 1570–1577, 2023.
- [4] R. Bhasin, G. Aslan, and J. Dehls, "Ground investigations and detection and monitoring of landslides using SAR interferometry in Gangtok Sikkim Himalaya," *GeoHazards*, vol. 4, no. 1, pp. 25–39, 2023.
- [5] J. Susaki, N. Maruo, M. Tsujino, and T. Boonyatee, "Detection of differential settlement of man-made structures coupled with urban development by using persistent scatterer interferometry (PSI)," *Remote Sens.*, vol. 10, no. 7, 2018, Art. no. 1048.
- [6] S. Bianchini, F. Pratesi, T. Nolesini, and N. Casagli, "Building deformation assessment by means of persistent scatterer interferometry analysis on a landslide-affected area: The volterra (Italy) case study," *Remote Sens.*, vol. 7, no. 4, pp. 4678–4701, 2015.
- [7] A. Nettis, V. Massimi, R. Nutricato, D. O. Nitti, S. Samarelli, and G. Uva, "Satellite-based interferometry for monitoring structural deformations of bridge portfolios," *Automat. Construction*, vol. 147, 2023, Art. no. 104707.
- [8] M. Bakon, D. Perissin, M. Lazecky, and J. Papco, "Infrastructure non-linear deformation monitoring via satellite radar interferometry," *Procedia Technol.*, vol. 16, pp. 294–300, 2014.
- [9] X. X. Zhu et al., "Global LoD-1 building model from TanDEM-X data," in *Proc. 14th Eur. Conf. Synthetic Aperture Radar*, 2022, pp. 1–4.
- [10] 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.*, 2008, vol. 2, pp. II-449–II-452.
- [11] M. Zhu et al., "Detection of building and infrastructure instabilities by automatic spatiotemporal analysis of satellite SAR interferometry measurements," *Remote Sens.*, vol. 10, 2018, Art. no. 1816.
- [12] N. Cavalagli, A. Kita, S. Falco, F. Trillo, M. Costantini, and F. Ubertini, "Satellite radar interferometry and in-situ measurements for static monitoring of historical monuments: The case of Gubbio, Italy," *Remote Sens. Environ.*, vol. 235, 2019, Art. no. 111453.

- [13] F. Lattari, A. Rucci, and M. Matteucci, "A deep learning approach for change points detection in InSAR time series," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5223916.
- [14] Y. Chen et al., "Prediction of InSAR deformation time-series using a long short-term memory neural network," *Int. J. Remote Sens.*, vol. 42, no. 18, pp. 6919–6942, 2021.
- [15] A. Shakeel, R. J. Walters, S. K. Ebmeier, and N. Al Moubayed, "ALADDIn: Autoencoder-LSTM-Based anomaly detector of deformation in InSAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4706512.
- [16] M. Costantini et al., "European ground motion service (EGMS)," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 3293–3296.
- [17] N. Anantrasirichai et al., "Detecting ground deformation in the built environment using sparse satellite InSAR data with a convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 2940–2950, Apr. 2021.
- [18] G. Milone and G. Scepi, "A clustering approach for studying ground deformation trends in Campania region through PS-InSAR TM time series analysis," *J. Appl. Sci.*, vol. 11, no. 4, pp. 610–620, 2011.
- [19] G. Martin, S. Selvakumaran, A. Marinoni, Z. Sadeghi, and C. Middleton, "Structural health monitoring on urban areas by using multi temporal InSAR and deep learning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 176–179.
- [20] 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.
- [21] 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.
- [22] L. Yu, T. Yang, Q. Zhao, A. Pepe, H. Dong, and Z. Sun, "Residual settlements detection of ocean reclaimed lands with multi-platform SAR time series and SBAS technique: A case study of Shanghai Pudong international airport," in *Proc. SPIE Conf. Remote Sens. Model. Ecosyst. Sustain.*, 2017, vol. 10405, pp. 230–243.
- [23] M. Evers, A. Thiele, H. Hammer, E. Cadario, K. Schulz, and S. Hinz, "Concept to analyze the displacement time series of individual persistent scatterers," *Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. 43, pp. 147–154, 2021.
- [24] V. Letsios, I. Faraslis, and D. Stathakis, "Monitoring building activity by persistent scatterer interferometry," *Remote Sens.*, vol. 15, no. 4, 2023, Art. no. 950.
- [25] M. Cuturi, "Fast global alignment kernels," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 929–936.
- [26] M. Cuturi and M. Blondel, "Soft-DTW: A differentiable loss function for time-series," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 894–903.
- [27] B. Barz, E. Rodner, Y. G. Garcia, and J. Denzler, "Detecting regions of maximal divergence for spatio-temporal anomaly detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 5, pp. 1088–1101, May 2019.
- [28] S. Guha, N. Mishra, G. Roy, and O. Schrijvers, "Robust random cut forest based anomaly detection on streams," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 2712–2721.
- [29] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [30] F. Rewicki, J. Denzler, and J. Niebling, "Is it worth it? Comparing six deep and classical methods for unsupervised anomaly detection in time series," *Appl. Sci.*, vol. 13, no. 3, 2023, Art. no. 1778.
- [31] E. Dai and J. Chen, "Graph-augmented normalizing flows for anomaly detection of multiple time series," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–16.
- [32] S. Saha, F. Bovolo, and L. Bruzzone, "Change detection in image time-series using unsupervised LSTM," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2020, Art. no. 8005205.
- [33] P. Podder, T. Z. Khan, M. H. Khan, and M. M. Rahman, "Comparative performance analysis of hamming, hanning and blackman window," *Int. J. Comput. Appl.*, vol. 96, no. 18, pp. 1–7, 2014.
- [34] X. Cai, T. Xu, J. Yi, J. Huang, and S. Rajasekaran, "DTWNet: A dynamic time warping network," in *Proc. 33rd Int. Conf. Adv. Neural Inf. Process. Syst.*, 2019, vol. 32, pp. 11640–11650.
- [35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [36] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–8.
- [37] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.
- [38] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2019, vol. 32, pp. 8026–8037.



Ridvan Salih Kuzu received the bachelor's degree in electrical and electronics engineering from the Department of Electrical and Electronics Engineering, Boğaziçi University, İstanbul, Turkey, in 2010, and the master's degree in system and control engineering from Boğaziçi University, in 2017, and the Ph.D. degree in applied electronics from Roma Tre University, Rome, Italy, in 2021.

Since 2010, he has held various positions at both private companies and public research institutions, including roles as an R&D Engineer, Consultant, Manager, and Marie-Curie Research Fellow. As of 2021, he is a Senior Researcher with the Remote Sensing Technology Institute, German Aerospace Center (DLR), Wessling, Germany, concurrently serving as an AI consultant with the Helmholtz AI Cooperation Unit. Presently, he is involved in several earth observation projects as a Senior Researcher and Principal Investigator. His research interests include representation learning, hyperspectral image processing, self-supervised and few-shot learning, explainable AI, and visual foundation models.

Dr. Kuzu was the recipient of the Best Demo Award at the 9th GTTI Thematic Meeting on Multimedia Signal Processing in 2019, the Best Paper Award at the International Conference on Pattern Recognition Applications and Methods in 2022, and the Hyperview Challenge Award at the International Conference on Image Processing in 2022. His doctoral thesis was honored with the 15th European Biometrics Research Award by the European Association for Biometrics in 2021.



Leonardo Bagagini received the bachelor's and master's degree in mathematics from the University of Rome: Tor Vergata, Rome, Italy, in 2012 and 2014, respectively, the Ph.D. degree in mathematics from the University of Florence, Florence, Italy, in 2017, and the postgraduate degree in meteorology and physical oceanography from the University of Naples: Parthenope, Naples, Italy, in 2018.

From 2019 to 2021, he worked as a Postdoctoral Researcher with the Institute of Atmospheric Sciences and Climate, Italian National Research Council, Rome, Italy, on the development of AI algorithms for the satellite passive microwave precipitation retrieval. Then, he joined the Space Technology Leonardo Lab, Rome, Italy, where he worked on the development of AI methods for optical and SAR data. He is currently a Technical Officer with e-GEOS, Rome, Italy, actively involved in projects related to the development of SAR-based products and algorithms. His research interests include machine learning, statistics, and their application in environmental sciences, particularly atmospheric physics, satellite imagery, and time-series analysis.



Yi Wang (Student Member, IEEE) received the B.E. degree in remote sensing science and technology with Wuhan University, Wuhan, China, and the M.Sc. degree in geomatics engineering with the University of Stuttgart, Stuttgart, Germany, in 2018 and 2021, respectively. He is currently working toward the Ph.D. degree specializing in data science for earth observation with the German Aerospace Center, and with the Technical University of Munich, Munich, Germany.

In 2020, he did an internship with Perception System Group, Sony Co., Stuttgart, Germany. His research interests encompass remote sensing, computer vision, AI, and self-supervised learning.



Corneliu Octavian Dumitru received the B.S. and M.S. degrees in artificial intelligence and pattern recognition from the Faculty of Electronics, Telecommunications and Information Technology and the Ph.D. degree in engineering all from Politehnica University Bucharest (UPB), Bucharest, Romania, in 2001, 2002, and 2006, respectively, and the Ph.D. degree in telecommunications from Pierre and Marie Curie University, Paris, France, in 2010.

From 2001 and 2007, he was a Research Assistant and from 2007 and 2011, a Lecturer with Politehnica University, delivering lectures and seminars and supervising laboratory works in the fields of information and estimation theory, communication theory, and signal processing. From 2007 and 2009, he was a Researcher in industrial projects with Telecom SudParis, France (formerly INT) delivering algorithms for the audio visual and film industry, and for the mobile service provider SFR (Vodafone group in France). Since 2010, he has been a Scientist with the Remote Sensing Technology Institute, German Aerospace Center (DLR), Oberpfaffenhofen, Germany. From 2005 to 2006, and in 2008, he was a Coordinator for two national grants delivered by the Romanian Ministry of Education and Research. Since 2002, he has been supervising/co-supervising bachelor, master, and Ph.D. theses in the field of artificial intelligence, machine learning, data mining, knowledge discovery in databases, semantics, compression, benchmarking datasets creation, impact of global changes, speech recognition, speaker verification, and watermarking. He is currently involved in several space projects as project management in the frame of the European Space Agency (ESA) and the European Commission (EC) Programmes for information extraction, explainable artificial intelligence, machine learning and knowledge discovery, semantics, ontologies, and knowledge graphs using remote sensing imagery. His research applications in Earth Observation are focused on the analysis of urban areas (his main research contribution), changes in polar areas, monitoring of natural disasters, monitoring of the coastal area, and security. His research interests include stochastic process information, model-based sequence recognition and understanding, basics of man-machine communication, information management, semantics, change detection in image time series, data mining, and image retrieval in extended databases.



Giorgio Pasquali received the master's degree in electronic engineering, specializing in space communication and sensing track, with the Università degli Studi di Pavia, Pavia, Italy, in 2019.

As part of his studies, he interned with Ticinum Aerospace, Pavia, Italy, where he conducted his master's thesis focusing on the development of a convolutional neural network for building detection from multispectral satellite images, specifically for damage assessment purposes. Since 2019, he has been with e-GEOS, Rome, Italy, as a Data Analyst. In this role, he is involved in designing AI-based solutions that contribute to the development of new technologies in various fields, including agriculture, maritime, cartography, and surveillance/security, working with multispectral optical data from satellites and aerial platforms.



Filippo Santarelli received the bachelor's, master's, and Ph.D. degrees in mathematics from the University of Camerino, Camerino, Italy, in 2012, 2015, and 2019, respectively.

After the Ph.D degree, he cooperated with the University for Foreigners of Perugia, Perugia, Italy, doing research in Natural Language Processing and with the National Research Council (CNR), Rome, Italy focusing in linear optimization. He began his work experience in computer vision and image processing during the master's degree with Halley Informatica S.r.l., Italy, where he developed algorithms for vision-guided robotics. In 2020, he started to work with e-GEOS, Roma, Italy, where he currently works, as an Algorithm Engineer in the context of SAR interferometry; as such, he took part in many projects, the most relevant of which are: the European Ground Motion Service, aimed at monitoring the whole Europe through SAR interferometry from Sentinel-1 data and multiframe/tracks analysis and integration with external GNSS data; the RepreSent project, in cooperation with ESA, focusing on non-supervised representation learning with Sentinel-1 data and, more specifically, on the classification of PS displacement time-series.



Nassim Ait Ali Braham received the M.Sc. degree in computer science from Ecole nationale Supérieure d'Informatique (ESI), Algiers, Algeria, and the M.Sc. degree in artificial intelligence and data science from Université Paris Dauphine-PSL, Paris, France, in 2019 and 2020, respectively. He is currently working toward the Ph.D. degree specializing in data science for earth observation with the German Aerospace Center, Wessling, Germany, and with the Technical University of Munich, Munich, Germany.

He was with the LIRIS-CNRS Laboratory, Lyon, France, in 2019, and with the LAMSADE-CNRS Laboratory, PSL Research University, Paris, France, in 2020. His research interests include deep learning, computer vision, self-supervised learning, and remote sensing.



Francesco Trillo received the graduation degree in telecommunications engineering from the University of Naples "Federico II", Naples, Italy, with a final grade of 110/110 cum laude, in 2005.

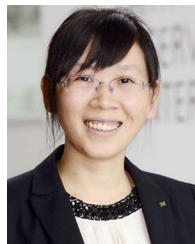
He began his professional career collaborating with the Department of Electronic and Telecommunications Engineering of the University of Naples "Federico II," continuing the development of his thesis work in the field of satellite remote sensing. In 2005, he had a Consultant role with Siemens Italia, Milan, Italy, for the management and configuration of radio links. From 2006, he began collaborating with Telespazio SpA, moving, from 2009, to e-GEOS SpA, Rome, Italy, an ASI/Telespazio company, in the Algorithms Development sector for Earth observation. He has more than 15 years of experience in radar data processing, especially in the field of SAR interferometry. He has authored or coauthored various publications in the field of SAR interferometry and, in general, in the various fields of application of satellite radar data. He is currently the Head of the SAR interferometry unit, e-GEOS.



Sudipan Saha (Member, IEEE) received the M.Tech. degree in electrical engineering from the Indian Institute of Technology Bombay, Mumbai, India, in 2014, and the Ph.D. degree in information and communication technologies from the University of Trento, Trento, Italy, and Fondazione Bruno Kessler, Trento, Italy, in 2020.

He was a Postdoctoral Researcher with the Technical University of Munich (TUM), Munich, Germany, and from 2015 to 2016, as an Engineer with TSMC Limited, Hsinchu, Taiwan. In 2019, he was a Guest Researcher with TUM. He is currently an Assistant Professor with the Yardi School of Artificial Intelligence, Indian Institute of Technology Delhi, New Delhi, India. His research interests include multitemporal remote sensing image analysis, domain adaptation, time-series analysis, image segmentation, deep learning, image processing, and pattern recognition.

Dr. Saha was the recipient of Fondazione Bruno Kessler Best Student Award 2020. He is a Reviewer for several international journals and was a Guest Editor for *Remote Sensing* (MDPI) special issue on “Advanced Artificial Intelligence for Remote Sensing: Methodology and Application.”



Xiao Xiang Zhu (Fellow, IEEE) received the M.Sc., the Dr.-Ing., and Habilitation degrees in signal processing from the Technical University of Munich (TUM), Munich, Germany, in 2008, 2011, and 2013, respectively.

She is the Chair Professor for Data Science in Earth Observation with TUM and was the founding Head of the Department “EO Data Science,” Remote Sensing Technology Institute, German Aerospace Center (DLR), Germany. Since 2019, Zhu is a Co-Coordinator with the Munich Data Science Research School (www.mu-ds.de), Munich, Germany. Since 2019, she also heads the Helmholtz Artificial Intelligence – Research Field “Aeronautics, Space and Transport”. Since May 2020, she is the PI and Director of the International Future AI Lab “AI4EO – Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond”, Munich, Germany. Since October 2020, she is a Director of the Munich Data Science Institute (MDSI), TUM. She was a Guest Scientist/Visiting Professor with the Italian National Research Council (CNR-IREA), Naples, Italy, Fudan University, Shanghai, China, the University of Tokyo, Tokyo, Japan, and the University of California, CA, USA, in 2009, 2014, 2015, and 2016, respectively. She is currently a Visiting AI Professor with ESA’s Phi-lab, Italy. Her main research interests include remote sensing and Earth observation, signal processing, machine learning and data science, with their applications in tackling societal grand challenges, e.g., global urbanization, UN’s SDGs, and climate change.

Dr. Zhu is an Associate Editor for IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING and serves as the Area Editor responsible for special issues of *IEEE Signal Processing Magazine*. She is a Member of Young Academy (Junge Akademie/Junges Kolleg), Berlin, Germany, Berlin-Brandenburg Academy of Sciences and Humanities, Berlin, Germany, the German National Academy of Sciences Leopoldina, Halle Germany, and the Bavarian Academy of Sciences and Humanities, Munich, Germany. She is in the scientific advisory board in several research organizations, among others, the German Research Center for Geosciences (GFZ), Potsdam, Germany, and the Potsdam Institute for Climate Impact Research (PIK), Potsdam, Germany.