



Methods in the spatial deep learning: current status and future direction

Bhogendra Mishra^{1,2} · Ashok Dahal³ · Nirajan Luintel^{1,4} · Tej Bahadur Shahi⁵ · Saroj Panthi⁵ · Shiva Pariyar⁶ · Bhoj Raj Ghimire^{1,7}

Received: 23 April 2021 / Revised: 27 November 2021 / Accepted: 4 December 2021
© The Author(s), under exclusive licence to Korean Spatial Information Society 2022

Abstract A deep neural network (DNN), evolved from a traditional artificial neural network, has been seamlessly adapted for the spatial data domain over the years. Deep learning (DL) has been widely applied for a number of applications and a variety of thematic domains. This article reports on a systematic review of methods adapted in major DNN applications with remote sensing data published between 2010 and 2020 aiming to understand the major application area, a framework for model development and the prospect of DL application in spatial data analysis. It has been found that image fusion, change detection, scene classification, image segmentation, and feature detection are the most commonly used application areas. Based on the publication in these thematic areas, a generic framework has been devised to guide a model development using DL based on the methods followed in the past. Finally, recent trends and prospects in terms of data, method, and application of deep learning with remote sensing data are discussed. The review finds that while DL-based

approaches have the potential to unfold hidden information, they face challenges in selecting the most appropriate data, methods, and model parameterizations which may hinder the performance. The increasing trend of application of DL in the spatial domain is expected to leverage its strength at its optimum to the research frontiers.

Keywords Deep learning · Machine learning · Remote sensing · Classification · Convolution neural network (CNN) · Long short-term memory (LSTM) · Autoencoder

1 Introduction

Remote sensing image analysis has been one of the most popular research areas over the last two decades for wide areas of application including but not limited to classification, change detection, disaster monitoring, pattern recognition, image fusion, segmentation etc. to unfold the natural phenomenon, and monitoring the earth's surface dynamics among others. Along with the advancement of remote sensing technologies, high performance of computing machines and low cost of resources, an abundance of remote sensing big data is available [1, 2]. Physical models have been the major methodological framework for analyzing remote sensing data. Machine learning techniques, also known as universal approximators, have been used for computational efficiency gain, empirical model development, and to solve classification problems [3]. Due to its ability to handle high dimensional remote sensing data even with a limited number of samples, Support Vector Machines (SVM), and easy to use ensemble algorithms like Random Forest (RF), have been the choice of the remote sensing community in the past years [4, 5]. Deep learning has recently claimed a strong presence with

✉ Bhogendra Mishra
bmishra@pri.gov.np

¹ Policy Research Institute, Narayanhiti Palace, Kathmandu, Nepal

² Science Hub, Kathmandu, Nepal

³ University of Twente, Enschede, The Netherlands

⁴ Kathmandu Center for Research and Education, CAS-TU, Kirtipur, Nepal

⁵ Central Department of Computer Science and IT, Tribhuvan University, Kirtipur, Nepal

⁶ Ministry of Forest, Environment and Soil Conservation, Pokhara, Gandaki, Nepal

⁷ Nepal Open University, Lalitpur, Nepal

significant success in image analysis for various applications over conventional machine learning techniques.

A number of factors like choice of data selection techniques, modeling approach, hyper parameterization; validation; are important for good model development. Contemplation to best practices in model development is crucial in any modeling endeavor; it is even more important in the development of deep learning models since they are modeled based on the data and not underlying physical processes. The selection of an inappropriate model and hyper-parameterization would misguide the orientation of the overall project and hence the chances of wrong decision prevail.

Although a number of review papers on Deep Learning (DL) in using remote sensing-based data have been published, the focus of these papers are primarily the application, architectural framework, and their performance [6–8], none of them addressed the overall methodological framework starting from input/output data selection to model evaluation. Therefore, the motivation for our study was to conduct a comprehensive review of the methodology adopted in major sub-areas of the remote sensing related to deep learning: Spatial Deep Learning (SDL), including image fusion, image registration, scene classification, object detection, LULC classification, image segmentation, and other tasks. The purpose of this work is to analyze the methods for modeling DL framework using remote sensing-based data for a wide variety of applications (Fig. 1).

2 Method

To identify the articles for the review, a title, abstract, and keyword search was done in the Scopus database. using the terms “deep learning” and “remote sensing”, (search date: June 18, 2020). The search gives 23,751 results. They fall under various subject areas; Computer Science, Earth and Planetary Sciences, Engineering, Physics and Astronomy, Mathematics etc. Among them, we are interested in the peer-reviewed publications of Earth and Planetary Sciences. Hence others were filtered out and 427 left. This database was used as the basis for further review and analysis. After reviewing the title, abstract and keyword, with major application areas of spatial analysis we ended up with 97 articles for the final revision.

We have reviewed 97 papers that have been published in well-known international journals from 2010 to 2020 and presented a synthesized methodological framework in detail in this work based on these 97 papers. The rest of the paper is organized as follows: Sect. 3 is results that include two sub-sections: subsection 3.1 summarize the selected major applications and methods following to develop the deep neural network (DNN) models based on remote sensing images. subSect. 3.2 outlines the synthesized framework based on the review in subsect. 3.1 and major steps in the DNN model development process are explained. The discussion and outlook toward future research are given in Sect. 4. Finally, the summary of the review is presented in Sect. 5.

Although a number of review papers on Deep Learning (DL) in using remote sensing-based data have been published, the focus of these papers is primarily the

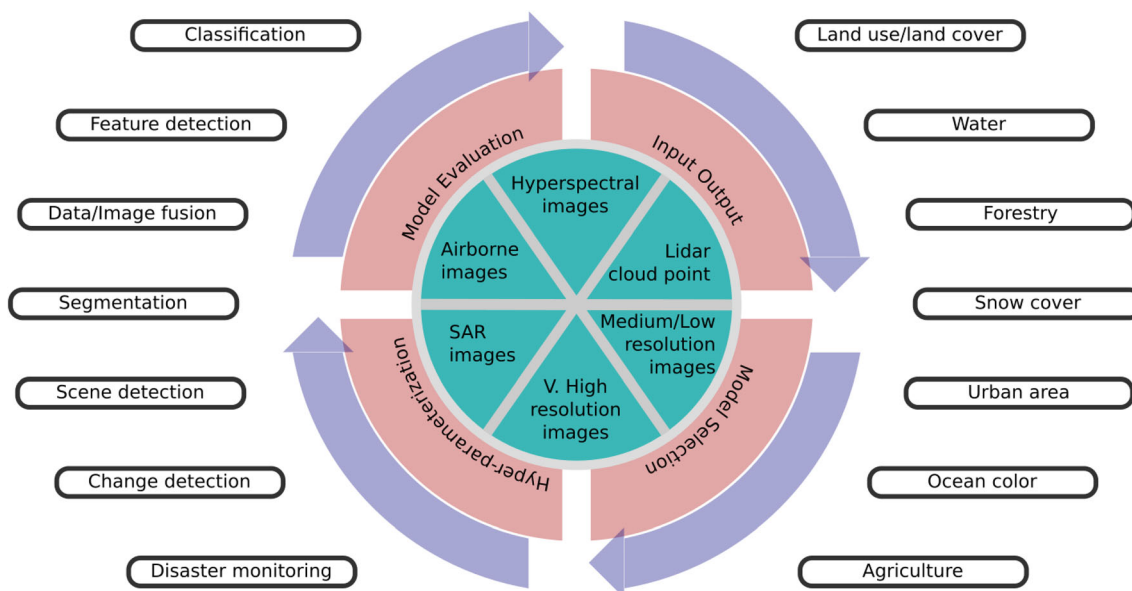


Fig. 1 The general framework to synthesize current status on spatial deep learning

application, architectural framework and their performance, none of them has addressed the overall methodological framework starting from input/output data selection to model evaluation. Therefore, in this work, we systematically reviewed and analyzed the methods for modeling DL framework using remote sensing-based spatial data analysis for a wide variety of applications. The major steps and consideration of the review is depicted in Fig. 1 and defined as follows. (1) To provide an overview on the use of deep learning with remote sensing data and their approach in various thematic and application areas of in spatial analysis such as image classification, image segmentation, image fusion, change detection, etc. (2) To develop a framework used for DN model development. (3) Uses the Bibliometrix to identify the trend of the deep learning application in spatial data analysis and synthesize the future direction.

3 Applications of deep learning with remote sensing data

The deep learning (DL) has a wide variety of applications in different thematic areas such as natural language processing, pattern detection, image processing among others. Though, image processing is one of the major applications of DL, the remotely sensed images have different characteristics and significances compared to the application than the common images. The remotely sensed images have their features such as spatially referenced, multi/hyperspectral and often available on regular basis—timeseries, that all introduce more complexities comparing the natural images. Therefore, they could have diverse applications and may require more resources for processing due to rich content and temporal availability. The volume is exceptionally increasing over time and the trend is likely to continue in the coming days too. On the other hand, DL has the capacity in feature representation [6] and to model very complex patterns successfully. Similar to many other fields, DL is gaining popularity in the remote sensing-based spatial dataset for various thematic areas such as change detection, image fusion, science detection, etc. The subsequent sections below include a comprehensive discussion of DL application based on remote sensing images, the methodology adopted, and the overall summary of the methodology while fitting the model.

3.1 Image fusion

Deep learning has been used in data fusion for different applications such as tree species recognition and tree mapping, urban/man-made feature/function. Furthermore, there also have been some studies on image fusion without

the focus on a specific application and with many application possibilities and also on image simulation using a different type of data source [9]. Based on the different data categories, most of the reviewed paper used multi-spectral image pair [10], multispectral and hyperspectral image [11], multispectral and panchromatic image pair, multispectral and SAR imagery [9, 12], hyperspectral and LiDAR data [9] and fusion of remote sensing and social data [13].

Most of the papers have used the different architecture of deep learning for the fusion of the dataset, there were major two approaches for the fusion of the images, first, the one-step approach where the fusion of images was done directly without extraction of features and second, two-step approach where extraction of feature or probability layer was followed by the fusion [14–17]. In the first approach, Nezami et al. [11] have used three dimensional convolutional neural network(3D-CNN) for the tree species identification, where the network consisted of 3 convolution layers, 3 batch normalization layers, 2 max pool layers, and 1,1 Softmax and rectified layers. Similarly, Yuan et al. [16] used their multiscale and multi-depth convolution neural network (MSDCNN) model which was an improvement to CNN. The model had 3 layers of CNN and a deeper CNN with two multiscale layers [16]. Lahaye et al. [17] used a model similar to a deep belief network that had restricted Boltzmann machines (RBM) and each RBM was trained separately. He & Yokoya [9] used the conditional generative adversarial network (cGAN) model with generator and discriminator sub-network where the generator network took the optical and the synthetic aperture radar (SAR) data as input and created an optical image which then fed into the discriminator network to identify real or fake simulation.

Secondly, the authors who used a two-step approach of feature extraction and fusion of features, also have used interesting methods such as for fusion of remote and social sensing data, 1D-CNN and long short term memory (LSTM) model were used where those models extracted the social sensing feature and then fused [13]. Similarly, for the deep representation of RS images, Yu et al. [18] used a model with layers consisting of convolution, pooling, and non-linear operators which transformed the input images into features followed by fusion operators. Cheng et al. [19] used two networks for segmentation and edge detection which were later coupled by fusion, the edge detection network used backpropagation for fine-tuning the results. For the fusion of hyperspectral and LiDAR data, Liao et al. [20] used a two-stage deep fusion framework where LiDAR and hyperspectral images were passed separately and then the probability maps were generated and those probability maps were fused. Chaib et al. [14] used the VGG net for the deep feature extraction from very high

resolution (VHR) images and it was followed by discriminant correlation analysis (DCA) transformation. At last, the fusion of the transformed feature was done, the final output represented the input images by a single feature [14]. For urban land use mapping, Feng et al. [21] used a combination of two networks of hyperspectral and LiDAR branches. From the hyperspectral branch, spatial-spectral features were extracted and from the LiDAR branch height features were extracted [21]. The extracted features were then fused using the feature fusion model and classification was done by Softmax classifier [21]. Shao & Cai [15] extracted features from panchromatic and multispectral images separately, to overcome the difference in spatial resolution Wald's protocol was used. The model had deeper architecture to exploit the high nonlinearities and each layer consisted of 64 filters with a size of $3 \times 3 \times 64$, after the feature extraction from both images, they were fused [15].

3.2 Disaster monitoring & post-disaster estimation

For disaster management, mostly deep learning and remote sensing images are used for monitoring and detection of natural or manmade disasters. Some of the papers have focused on disaster preparedness and disaster management side such as Antoniou & Potsiou [22]. Most of the paper focused on the monitoring/detection of the hazard itself than that of mapping its impact. Sun et al. [23] used deep learning for the prediction of the ground PM2.5 values instead of monitoring the physical hazard on the ground. Ma et al. [24] used deep learning for mapping the elements at risk (buildings) instead of mapping the hazard.

The methods used for disaster detection and management are mostly detection/recognition algorithms such as hazard detection, the element at risk detection, and facilities detection and prediction/modeling algorithms [22]. Firstly, for detection algorithms, Liu & Wu [25] used the autoencoder model with arbitrary neurons and backpropagation for the detection of landslides in their WavDAE-2 model. The deep learning framework with constraints (DLWC) model for landslide detection using hyperspectral images proposed by [26] used the constrain which shows the probability of having landslides and which limits the false positive classification in the model. Zhao et al. [27] used an approximately symmetrical deep neural network (ASDNN) for change detection in heterogeneous images, the model consisted of multiple hidden layers with stacked Restricted Boltzmann machine (RBMS), the model focuses on the novel algorithm rather than that of the application. For the detection of the soccer fields for helicopter landing, Antoniou & Potsiou [22] used a deep learning autoencoder model with three encoding layers. After each deep layer batch normalization was also performed. Ma et al. [24]

have simplified the You Only Look Once YOLOv3 model by replacing the DARKNET53 CNN with lightweight shuffleNet v2 and other loss functions were replaced by generalized intersection over union loss; by doing that the model became quite simpler. Some papers also have used deep learning in unmanned aerial vehicle (UAV) images as well as SAR images. Baur et al. [28] used a faster regional-convolutional neural network (Faster R-CNN) model on the UAV images for the detection of landmines. Paper by Zeng & Wang [29] used SAR images with an oil spill convolution network (OSNet), the model is based on the VGG-16 and uses the CNN for detection of oil spills.

Secondly, for the prediction/modeling of the different parameters such as drought or PM2.5 which cannot be seen directly in the image or mapped directly through the image, different covariates have been used together with the remote sensing imagery [28]. Shen et al. [30] used the drought index as a dependent variable with remote sensing imagery as independent variable. Furthermore, the author used the deep feedforward neural network (DFNN) model to train the data with all the covariates [30]. In the case of monitoring PM2.5 using remote sensing imagery, Sun et al. [23] used PM2.5-DNN model to monitor PM2.5 using satellite imagery. The author also used ReLU and linear functions for activation of the source and output layer and generated the PM2.5 concentration from neurons.

3.3 Scene classification

Deep learning (DL) has been widely used in scene classification, land use classification [31], classification and reconstruction of urban buildings and tree classification [32].

Most of these studies used UC Merced Land-Use Data Set [33], NWPU-RESISC45 Data Set [34], WHU-RS data Set [35], Aerial Image data [36], Google data set [37], Brazilian coffee scene data set [38], SAT-4 [39], SAT-6 [40], RSSCN7 dataset for land use classification [41], and tree classification [42]. Likewise, both terrestrial and aerial laser scanning point clouds were used for the classification and reconstruction of the urban building [43]. Second, the use of DL was further broadened with image retrieval [42, 44], image processing [45] multimodal image registration [46] and image classification [47]. For these, Spot-7 PXS and S2 time series were used in image processing, UCMD and dual-source remote sensing image data set containing panchromatic and multispectral images in image retrieval, and Google Earth, World View-2, Landsat 8, TerraSAR-x, and Google maps were used in multimodal image registration.

Most of the papers considered in the study have used data preprocessing and the different architecture of deep learning for scene classification. At first, dataset curation;

use of edge-aware resampling algorithm and ground point removal; noise removal, and main direction adjustment of the tree trunk were used as initial steps for data preprocessing [42, 47]. This also includes radiometric correction of the images. While preparing the dataset, the images were resized to be well-matched with the required input size during feature extraction [40] and DCA fusion [14]. In some cases, dual samples were randomly tiled from the pairs of remote sensing images [44].

After the completion of preprocessing, the images were passed through a series of steps based on the different architecture of the model used. Gu et al. [39] proposed a multiple deep rule-based (DRB) classifiers model trained with the segments of remote sensing images at different levels of granularity. This DRB ensemble of a multilayer classifier is a modular/layered form with (1) rotation layer; (2) segmentation layer; (3) scaling layer; (4) feature extraction layer and (5) massively parallel rule-based system. Chaib et al. [14] suggested a three-step model for scene classification containing feature extraction with deep VGG-Net, the use of DCA approach to fusing the extracted features, and the use of a support vector machine (SVM) as a classifier. The multiscale CCN framework also consists of three main steps. First, a multiscale image set was generated by rescaling and one of its rescaled images was sampled from its multiscale image set. Second, these images were fed into the F-net and V-net to compute the Softmax loss and similarity loss. Finally, the partial derivatives of the weights shared by the F-net and V-net were computed and the shared parameters were updated for classification [41]. Liu et al. [47] also proposed a deep few-shot learning method. Initially, spectral-spatial features were extracted to reduce the labeling uncertainty via a deep residual 3-D convolutional neural network. Then, the network was trained by episodes to learn a metric space where samples from the same class are close and those from different classes are far and the testing samples were classified by the nearest neighbor classifier in the learned metric space. For tree classification, Zou et al. [32] proposed a novel voxel-based deep learning method that includes three major steps such as individual tree extraction based on the density of the point clouds, low-level feature representation through voxel-based rasterization and the classification.

Likewise, some of the studies involved a two-step model for scene classification. Zou et al. [48] suggested a deep belief network (DBN) for layer-wise feature abstraction and reconstruction weight fine-tuning. The DBN used a family of Restricted Boltzmann Machines to calculate the layer-wise reconstruction weights which were then back-propagated to fine-tune the weights. Gong et al. [49] recommended diversity-promoting deep structural metric learning (D-DSML) for scene classification. In the training

phase, training samples were passed through initializing CNN and then through the training D-DSML. Then these samples were fine-tuned with Softmax classifier. In a semi-supervised generative framework, two independent classifiers (each of them is constructed by a deep feature extractor CNN and a one vs rest SVM) and one discriminative evaluator were involved in image scene classification. Boualleg et al. [35] devised a two-step deep forest model, an ensemble-learning method of deep learning. It consists of convolution layers and fully connected layers in pre-trained CNN for feature extraction followed by multi-grained scanning and cascade forest structure in the deep forest for feature classification. Liu et al. [47] suggested a Siamese network model containing two images obtained from the same class as a positive pair and a different class as a negative pair. These networks further compose two identical CNN models, three additional convolutional layers, and one square layer. A metric learning regularization term was imposed on the features as learned through CNN models. A new hybrid deep learning model-Inception-LSTM model which combines the Inception-V3 model and the LSTM model is also a two-step scene classification model [37]. In this model, features were extracted from the pre-trained Inception-V3 model through the BN layer network structure as a bridge, and the scene was finally classified by the Softmax classifier. In contrast, Weng et al. [31] developed a single step land use classification via Extreme Learning Classifier (ELM) where preprocessed images were pre-trained as CNN excluding fully connected layer and the ELM was trained as the classifier and tested on the CNN features.

3.4 Landcover classification

Deep learning has been used widely in land use and land cover classification in various ways with a diverse dataset. Major applications include classifying residential area, park, buildings, storage tanks, tennis court, parking lot, harbor, intersection, forest, agriculture, chaparral, river, beach, golf, overpass, and runway among others. However, few others focus only on specific sub-classes such as a tree, shrub, and grass classification, delineation of impervious surfaces [50], urban built-up area classification, complex wetland (bog, fen, marsh, swamp, and shallow-water) classification, scrub, willow swamp CP hammock, CP/Oak, slash pine, oak broadleaf, hardwood, swamp, marsh, mudflats, water [51–53].

The majority of the works are based on the pre-processed dataset they include the University of California (UC) Merced, which is the most popular in the land use and land cover classification using DL methods, similarly, KSA and AID, ImageNet database, the Space Net dataset for the cross-domain classification [54]. Additionally, an airborne

dataset is also found commonly used with DL algorithms such as the SAT-4 and SAT-6, RSOD datasets (prepared by Long et al.; it contains 446 aircraft images [54], AVIRIS sensor [55], Aerial photos of S1 [56]). Further, a large number of workers are based on very high-resolution satellite images they include but not limited to IKONOS-2, GeoEye-1, and WorldView-2 with spatial resolutions of 1, 0.5, and 0.5 m large-scale aerial images of size 600×600 with multiresolution (8–0.5 m) [57], Demons-2 and QuickBird, Pleiades -1A images, RapidEye, Quickbird II, Gaofen-1, Jilin-1, Ziyuan-3, Google Earth data [58]. Some works are based on the Lidar point cloud dataset while others are based on multispectral images from various satellite sensors [50] however, hyperspectral images [56], images RGB only images [53], SAR images mainly Sentinel-1A RS satellites [59] and Polarimetric synthetic aperture radar (PolSAR) imagery [60], and AIRSAR sensor [61]. are also used. Furthermore, Sentinel-2, Landsat 8 images and MERIS and ENVISAT etc. are also widely used sensors with DL [62].

Prior to the classification, satellite images were corrected for atmospheric effects, provided the benchmark [63], re-sampling and registration [50], geometric transformation [64], normalized and formalized by the proposed self-adaptive cellular-based method [52], used initial whitening [58], pan-sharpening [53] as data pre-processing steps. Feature extraction was done before training the models for the land cover classification [62]. For example, Ammour et al. (2018) [57] used pre-trained CNN for the feature extraction and denoising the input images using autoencoder (DAE) for dimensionality reduction.

Architecture used for the land use and land cover classification are the variant of CNN [62]. Deep Q-network (DQN) [43], multilayer perceptron (MLP) [55], LSTM [52], Deep Fully Convolutional Network [54], DNN [64], a semantic segmentation-based deep learning method, DeepLabV3 + , are applied for classification [47], adaptive multiscale deep fusion residual network (AMDF-ResNet) [51], ImageNet Pretrained Networks [44], ResNet [46], Hybrid of principle component analysis (PCA), deep learning architecture, and logistic regression [58], a multiscale dense network (MSDN) [35], GoogLeNet network [48] are used for classification.

Performance of classification models was evaluated by the overall accuracy (OA), kappa coefficient, average accuracy (AA), confusion matrix, and paired T-test [54]. Each dataset was divided into training and test set having fivefold cross-validation [65]. The scatter diagram is plotted based on the values of the estimated percentage of impervious surface (PIS) and real PIS to show the accuracy and effectiveness of AEIDLMS (Automatic extraction of urban impervious surfaces based on deep learning and multi-source remote sensing data). The employed metrics

for the analysis are the co-efficient of determination (R^2), root means square (RMSE), and the mean absolute error (MAE) [50].

3.5 Change detection

One of the major applications of DL using remote sensing images is change detection. Monitoring of urbanization [66] and large-scale deforestation which are rather challenging via field survey but remote sensing based on DL algorithm has been emerging as a reliable approach. Some papers explored the prospect of transfer learning on a change detection task when small training samples are available. In addition, the possibility of fully automated change detection based on pre-classification using unsupervised learning or spectral indices was also used. Gong et al. [49] demonstrated that the use of superpixels improves the change detection task.

Architectures used for change detection were mainly variants of convolutional neural networks and deep belief networks. Authors of some of the reviewed papers tested multiple architectures to evaluate their performance while some focused on single architecture. Ortega Adarme et al. [4] tested three architectures: Early Fusion with 3 convolutional layers and 2 fully connected layers using a concatenated double temporal image, Siamese Network with two parallel subnetworks having 3 convolutional layers shared between the double temporal images, and Convolutional Support Vector Machine with 3 convolutional layers with SVM as convolutional filters. De Bem et al. [67] used relatively deeper networks with skip connections from the encoder to corresponding decoder layers: SharpMask with 114 layers, U-Net with 69 layers and ResUNet of 93 layers with residual network replacing conventional convolution block. Song and Choi [68] built a change detection network with a combination of 3D FCN having nine 3D convolutional layers, LSTM, and two 2D FCN where the first 3D FCN layer was a multiscale convolutional layer and weights of 3D FCN were transferred from pretraining of 3D FCN as segmentation network. Yang et al. [69] jointly trained two parallel U-Net based network of 9 layers with lower layers shared between the source domain and target domain and fine-tuned the upper layers of the target domain using pseudo labels from unsupervised learning. Networks built using supervised classifiers such as extreme learning machines and SVM on top of DBN, DBN was first trained unsupervised manner and subsequently fine-tuned with labeled data [70, 71]. For change detection, deep learning was used in an end-to-end manner or with features extracted from the input images as input for the deep learning model [70].

Some papers focused on automating the entire change detection task using high confidence labels—from

unsupervised learning or a rule-based method using spectral indices sensitive to the target surface—as the input labels for a deep learning network. Gong et al. [49] first performed preliminary classification on superpixels with Otsu method using spectral feature map, textural feature map, and spatial feature maps separately, and labeled them as changed or unchanged based on majority voting, then trained deep belief network and sparse denoising autoencoder using pseudo labels to perform classification on uncertain pixels. Huang et al. [72] used spectral indices to detect buildings in both the images, then fused the difference and ratio images and dilated the shadow, thus to obtain few high confidence changed and unchanged labels to use for fine-tuning deep belief network-extreme learning machine (DBF-ELM) model. Yang et al. [69] used the unsupervised algorithm robust semi-supervised fuzzy C-means (RFLICM) to generate preliminary change maps, selected high confidence changed and unchanged pixels based on a difference in local land cover variation and distance from the cluster center, which was subsequently used to train the change detection network.

3.6 Image segmentation

Deep learning was implemented for semantic segmentation of remotely sensed images as well as 3D point clouds. General land use land cover classification was the major application of remote sensing image segmentation. Besides, some papers focused on the detection of target objects with specific applications such as automatic road detection for updating road maps, water bodies detection for monitoring water resources, and rice lodging detection for crop damage assessment [73]. The availability of benchmark datasets facilitated experiments for new architectures for segmentation tasks without specific application. Pierdicca et al. [74] used 3D point cloud data for the segmentation of cultural heritage sites useful for 3D documentation of cultural monuments and recognizing historical architectural elements in the heritage sites. Some papers demonstrated the possibilities of successful transfer learning using synthetic data [70] or using secondary manually labeled data, and the effectiveness of dimensionality reduction of multispectral satellite images for reducing computation load without compromising accuracy [75]. Yan et al. [71] focused on data preprocessing and the development of an auxiliary network called rotation density network to simplify as well as improve the segmentation of 3D point clouds of the mining area.

Two categories of deep learning architectures were more frequently used for segmentation: convolutional network for regular imageries and point-based network for 3D point clouds [61]. Some of the papers reviewed demonstrated the effective adoption of existing deep learning models for

remote sensing applications while others developed models specifically for remote sensing images [57]. Majority of segmentations on regular imageries employed fully convolutional neural network (FCN) based architectures. SegNets, ResNets, UNets, and DeepLabV3 + were frequently used for segmentation of remote sensing images. SegNets and FCN-8 s were widely used as benchmark architectures for demonstrating the effectiveness of recently developed networks. Audebert et al. [76] branched the decoder of SegNet to get outputs at several resolutions in order to improve the segmentation results; they also build an early fusion network named V-FuseNet by fusing features maps from orthophoto and auxiliary data in the encoder layers and a late fusion network by fusing outputs of two independent network using SegNet and ResNet-34. Kemker et al. [70] used SharpMask and RefineNet built upon ResNet-50 backbone by connecting feature maps from encoder to decoder through refinement modules. Wu et al. [77] used FCN with ResNet-50 backbone and UNet with 19 layers to perform semantic segmentation using transferred weights and a small number of color-coded PolSAR images. SegNet and FCN-AlexNet were used to assess rice lodging using vegetation indices as input data [73]. Henry et al. [78] tested three network architectures of different depth FCN-8 with VGG19 backbone (19 layers), Residual UNet with 15 layers, and DeepLabV3 + with 65 layers for road segmentation using SAR data. SegNet was used to demonstrate the effectiveness of dimensionality reduction on multispectral remote sensing data [75]. Li et al. [79] modified DeepLabV3 + to use input images at multiple scales for the extraction of water bodies.

However, newer architectures were also proposed for remote sensing image segmentation [80, 81]. Chen et al. [82] proposed ResNet-like “Shortcut block” and used it to develop very deep networks (67 and 139 layers) SNFCN and SDFCN with and without skip connections from the encoder to a decoder. Mi et al. [83] developed a hybrid model consisting of two branches: the main branch comprising of four layers of ResNet-101, 4 parallel layers of atrous spatial pyramid pooling, and differential decision trees, and an auxiliary branch with several convolutional layers for feature extraction followed by super-pixel segmentation. Deep learning at multiple-scales was realised by branching the decoder to get outputs at several layers [76], feeding neural network with input at multiple scales [79], or by combining features at several resolutions within the network [80]. A deep interactive segmentation network that used user interactions to generate guidance maps as attention criteria for the subsequent Attention-Guided Multi-scale Segmentation Network (AGMSSeg-Net) was developed by Li et al. [80]. The authors used the guidance maps and features extracted with pre-trained VGG-19 as extra channels of the input image [80]. Although the use of

a graph model for image segmentation was uncommon, Attention Graph Convolution Network (AGCN) was developed by stacking attention mechanism layer and two graph convolution layers to perform segmentation on big SAR data by using super pixels as nodes of the graph [1]. Semi supervised method of image annotation was developed by Yao et al. [84] using the low level features, that are, gist features, color histogram features and SIFT features as input to stacked discriminative sparse autoencoder (SDSAE); the weights learnt by training SDSAE on land cover classification data was transferred, and classification was performed by additional classification layer and image patch-level labelled data, thus reducing the overall computation time.

Segmentation operations with point clouds were performed directly on 3D points rather than on projected surfaces or voxels. Bachhofner et al. (2020) used U-Net architecture-based generalized sparse convolutional neural network (GSCNN) built with sparse convolution blocks to segment 3D points generated from tri-stereo satellite imagery [85]. The segmentation of archeological and cultural heritage sites was performed with a dynamic graph convolution neural network (DGCNN) constructed with several blocks of edge convolutional layers [74]. Yan et al. [71] developed an auxiliary network called Rotation Density Network that efficiently extracts structural features based on the density of point cloud and used it in conjunction with point-based networks PointNet or PointCNN to enhance the point cloud segmentation of the mining area.

The issue of insufficient training samples for remote sensing data was addressed via data augmentation techniques such as flipping, rotating, Gaussian transform, Gaussian blur, random scaling, random translation, and via transfer learning [83, 85]. CRF was used as a post-processing technique to fine-tune the segmentation results by some authors [84, 86].

3.7 Feature detection

Among many applications, feature extraction from remote sensing images using deep learning methods, especially the convolutional neural network is very common. These applications include tree detection for forest management [86, 87] parcel detection for precision agriculture, building, road, bridge, and built-up area detection for urban planning, and Geographic Information System, global reservoir detection. Few studies on feature detection using deep learning were not focused on a specific application but aimed to detect a various object in remotely sensed images. Based on the data source, the majority of reviewed works consider the satellite imagery as a primary data source while few studies consider another aerial imaginary such as

Unmanned Aerial Vehicle (UAV) and synthetic aperture radar (SAR). The data source varies in their spectral resolution, spatial resolution and temporal resolution.

Most of the paper used a variation of CNN for feature extraction such as Mask R-CNN for tree detection, RI-CNN with rotational invariant layer for object detection, Siamese U-Net and Modified U-Net with Xception for Building extraction, CNN-ResNet-50 global reservoir detection, Building Region Proposal CNN for building extraction, Multi-task learning with U-Net for road extraction, Multi-Scale Feature CNN for cloud detection, Double stream deep CNN [81, 88]. Each modified version of CNN consists of multiple layers: the convolutional layer, the pooling layer, fully connected layer. The convolutional layer and pooling layer are the key components of these networks for feature map generation.

In Mask RCNN [87] the first few layers extract the feature map, and this feature map goes to later layers for object detection. Similarly, in [89] the U-Net is used to extract features. It consists of encoder and decoder block, where the encoder block has three convolutional layers and one max pooling layer while the decoder block has one convolutional layer, normalization, and activation followed by another same set of layers. In Hui et al. [90] U-Net architecture is modified with the Xception module to extract the effective feature from remote sensing images. Also, Multitasking is implemented to incorporate the structure information of the building. The convolutional block of the encoder of the U-net is replaced with the Xception module. The U-net architecture consists of a one-pre convolution layer, followed by five successive Xception modules in the encoder side whereas the face-up-conv block of the decoder is similar to the original U-Net.

Liu et al. [47] building region proposal network based on CNN (BRPN) was used to generate candidate building areas instead of sliding windows used in the fast RCNN model. Unlike Regional Proposal Network is embedded in Fast-RCNN, BRPN constructs the network by combining the spatial hierarchies of the multilevel image training dataset. In this work, a shared feature from CNN is used to generate a building region proposal in BRPN. However, few studies explore the other possibilities for feature extraction rather than using CNN. Chen et al. [88] implemented an attention and balanced feature pyramid (ABFP) network to generate feature maps that use the concepts of Balanced Feature Pyramid and Attention mechanism. Since BFP network can aggregate low level and high-level feature better while attention mechanism passes the useful information to next level and ignore the useless information, the integration of these two techniques improves the overall precision of object recognition in SAR images. Similarly, a fully convolutional network to segment the image on a pixel by pixel basis with the symmetric encode-

decoder module to extract multi-scale features and residual connections to train the network effectively was proposed [91].

4 Methods used for DN model development

The review of the application domain and methods followed in each domain by past researchers using remote sensing data has suggested that a set of common steps are to be followed to develop the DN model regardless of the problem domain [15, 92]. In this sub-section a generic framework comprises these common set of steps, synthesized out of the review, is presented which is expected to work as a guideline for devising DN model using remote sensing dataset. It should be noted that the model development steps covered here have been enhanced to suit the deep learning from the [92], which covered the ANN in hydrology. Figure 2 illustrates the generic framework for the DN model development.

The very first step of the model development is started from the problem objective, contextualizing the modeling problem, and conceptualizing the systems that include the identification of the model output and a set of potential input variables available to train and test the model. As the deep learning models are data-driven, the set of input variables/indices are very important. This mostly depends

on the data availability and prior knowledge of the modeler. Once the model input dataset and output variable are confirmed, the next steps would be data preparation that suits to fit the models followed by the model development step followed by architecture selection, hyper-parameterization, and model evaluation. The details of each step are explained in the subsequent subsections as follows.

4.1 Data input–output selection and preprocessing

4.1.1 Choice of potential inputs and outputs

The outcome variable/data is defined based on the study objective, while, the input dataset is first selected based on the expert’s knowledge. However, different approaches have been applied to filter out the most irrelevant dataset and finalize the input variables for the model. The most common methods of feature selection include filter, wrapper, and hybrid methods. The filter-based method is based on different statistical measures such as variance, chi-square, or correlation coefficients, a variance of inflation factor (VIF), and then selects the highest-ranked features. The wrapper method sequentially searches the best input subset of the input dataset to fit the model. It uses the best set of input variables that yield the best accuracy. It uses the accuracy of the previous model and judges if a new input variable is to be included or needs to be

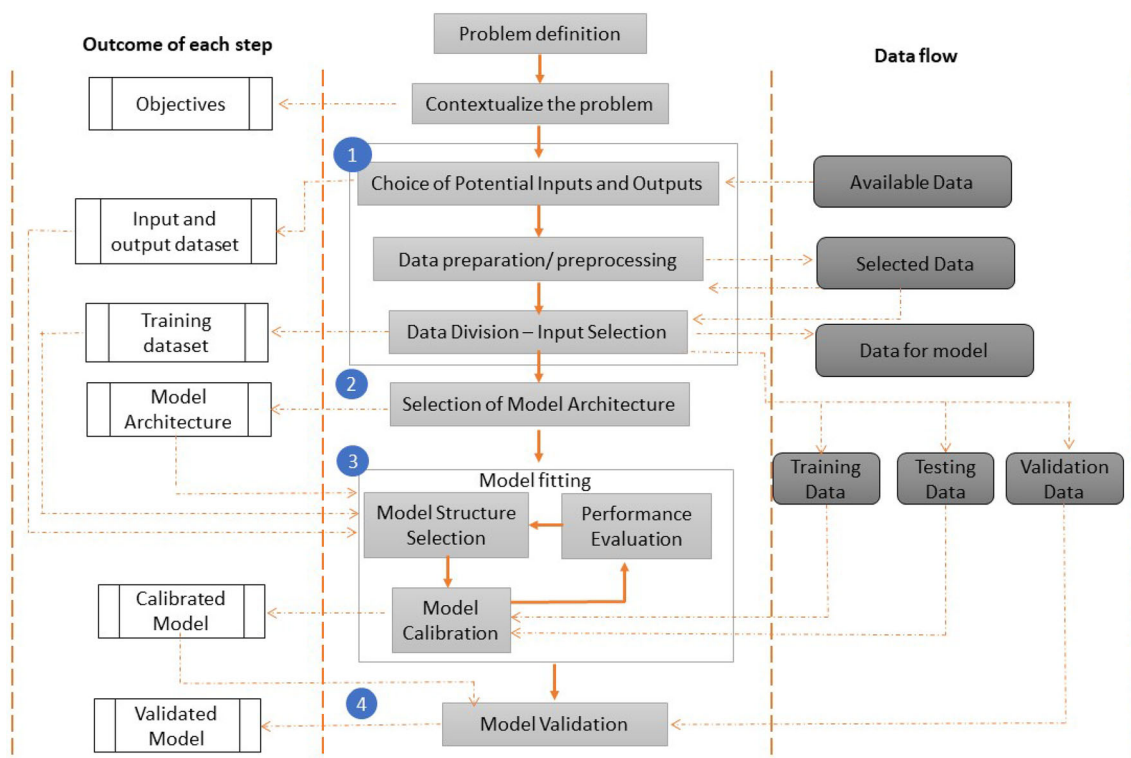


Fig. 2 Deep learning model development process

removed. It is done in three ways, exhaustive search: tries all possible combinations and select the best performing input set, forward selection: it works from the single input variables and sequentially adding one feature at a time and evaluate the best set of input variables. The backward selection: it starts with all input variables and removes one feature at a time and evaluates the performance and choose the input variable sets that yield the best outcome. The hybrid methods incorporate the characteristics of filter and wrapper feature selection methods [62].

In the remote sensing-based dataset, the input pixel values come as an array and extract the features in different layers. Such layers of features are not designed by the analyst; they are learned from the data using the general-purpose learning procedure. Such a network itself discovers the underlying structures in a high-dimensional dataset. Therefore, the input dataset selection in the remote sensing application could work differently than the general-purpose applications and have not been discussed in detail about the input data selection in any papers [41].

4.1.2 Data pre-processing/preparation

The pre-processing, or data preparation for the input images is very important in remote sensing. The very first step of the study should go through image correction such as radiometric correction, band line repair, atmospheric correction, co-registration, and resample if uses the multi-sources and multi-resolution. The data preprocessing includes several operations that suit the different types of input data and the output variables. Data augmentation is one of the most common operations to increase the data volume artificially that could reduce the overfitting in case there is not enough input dataset. It was performed by a rotation of training tiles (90° , 180° , 270°) and horizontal and vertical flipping. Pre-classification of the input images was also done as a preprocessing which is the input/output data set for some deep learning models [49]. Crop the images into various patches of different sizes of different dimensions such as 16×16 , 224×224 , 300×300 pixels, and 450×450 [93]. The patch size is selected based on the spatial resolution, targeted applications mostly through expert knowledge or hit and trial methods. Such patches are labeled using the reference dataset [70]. Labeled the feature in an input image are for the input sample this is done manually, such as major road type, road, water, built-up, vegetation, specific crop fields, crop properties such as crop lodging, ridge and background, etc. The majority of such labeled is done using some reference datasets such as Google Earth images using annotated tools like “LabelMe” 87. However, many studies are based on different image processing tools to label the input dataset,

such as Bachhofner et al. [85] labeled the dataset manually into five classes.

4.1.3 Data division – input and output selection

The input datasets are divided into train, validation, and test set in various ratios such as 50%, 25%, and 25% respectively. This proportion is different in different studies, there is neither a rule of thumb nor an objective method for this. The input images were divided for training and testing in 3:1 ratio (for Vaihingen, 16 + 4 and Potsdam 18 + 6) by Chen et al. [82]; similarly, data was separated into training (80%) and test sets (20%) by Wu et al. 2017 [77]. 100 images were used for training and 10 images for testing by [75]. The dataset was separated into training, cross-validation, and test sets (60 + 20 + 20%) with (latest) 2019 data used only for testing in [73]. Vaihingen and Potsdam datasets were divided into training/test sets with 16/17 and 24/14 tiles. Similarly, 80% used for training and 20% for testing were used in [85] and 75% data as a training sample and 25% as testing in [71]. Braga et al. [87], divided the 19,656 samples into 15,122 as a training sample and 4535 images as test samples. A very large number of studies were based on the various image libraries. The most commonly used libraries include UC Merced for the land use classification [18], Google Earth Images for the reference dataset, WHU-RS data set, GoogleLeNet network trained on ImageNet data set[57].

4.2 Selection of model architecture

Given easy-to-use machine learning libraries like scikit-learn, Keras, multiple deep learning models are available to fit a predictive model with the set of data. It is the process of choosing the most appropriate model from the number of candidate models considering multidimensional influential factors, for example, time taken to train the model, resources need, their performances, available resources, etc. The model selection can be done using probabilistic, random train/test and resample approaches. The random train and test split work based on the division of the input dataset into train and test ratio wise. There are few libraries such as `train_test_split` function in scikit-learn python library. The different proportions of the data split can use to test the model performance with the unseen dataset. In the resampling technique, the model is chosen based on the performance of the model, which is done by resampling the data set into train and through a set of iteration. The performance is computed with the out-of-sample data. They include the K-Fold, Stratified K-Fold etc. The probabilistic model selection involves scoring the model’s performance using the probability framework such as maximum

likelihood estimation to choose the most appropriate model among the number of candidate models.

Referring to the peer-reviewed articles in applications of remote sensing datasets, the formal process of model selection was not explained in most of the papers. The majority of them seems to be based on the expert knowledge and the historical application of the methods, primarily, the LSTM was used for the time series dataset, while in the image processing that likely influences spatial information such as classification, scene segmentation, changes detection, CNN is used in most of the cases. The majority of such studies have directly adopted and used the pre-provided libraries dataset to train and test them. Such as ResNET-50 [77], SegNet [75], U-Net [77], DeepLabV3+ [79], FCN-AlexNet [73], Rotated Density Network (RD-Net) [71]. A large portion of them is concerned only with the model performance. None of the manuscripts have discussed the availability and requirement of the resources while selecting the models.

CNN's are usually used where convolutional operations are a good assumption; however, in some applications the CNN has been used for feature extraction [50] followed by other models. In terms of application, autoencoders, CNN and RBM or DBN do not have a clear separation; however, most of such studies choose CNN. While the problem of dimensionality reduction is raised, autoencoders are using along with other architectures, for example, Ammour et al. [57] used the pre-trained CNN for the feature extraction and autoencoder for dimensionality reduction. The DBN is considered an ideal deep learning model while the labeled training sample size is small as it is good in feature learning capability [19]. The VGG net was used for the extraction of deep features from the input VHR images [14]. Autoencoders were implemented to learn features of the large dataset in an unsupervised manner [57].

Alternatively, many works are based on the pre-trained image classification networks that have already learned to extract powerful and informative features from natural images and use it as a starting point to learn a new task. The majority of the pre-trained networks are trained on a subset of the ImageNet database which is used in the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) [94]. These networks have been trained on more than a million images and can classify images into 1000 object categories, such as a keyboard, coffee mug, pencil, and many animals. Using a pre-trained network with transfer learning is typically much faster and easier than training a network from scratch but a pre-trained network dedicated to remote sensing is not available in the market.

4.3 Model fitting

4.3.1 Model structure selection

Deep learning models consist of two types of parameters (i) hyperparameters – set before start the training and (ii) model parameters – learned during the model training such as weight of neurons. It develops a functional form of the relationship $f(\cdot)$ between model inputs and outputs(s) during the training time [16]. Optimizing a number of hyperparameters to develop an optimum relationship between inputs and output(s) while working with deep learning is a challenging task. The optimal network structure should balance between the network complexity – network size, transfer function, etc., and the performance of the network. If the network complexity is too low i.e. a number of layers and the hidden neuron is low, processing speed is high, however, the network may not well capture the best relationship between input and output dataset, and increase the processing speed [95]. On the contrary, it is too complex while increasing the number of layers and neurons, increase the complexity, perhaps it works well but it is too time-consuming and resource-demanding. The most important characteristics are network accuracy, speed, and size. Choosing an optimum network is generally a tradeoff between these characteristics. The most common way is to start with a basic model and improve the model iteratively. That could take a list of pixels as input and class-label output [84] in case of remote sensing-based classifications, segmentation or data fusion. The model structure selection/ hyperparameters tuning involves selecting the number of layers, number of hidden units per layer, activation function, optimizer, hidden layers, learning rate, initialization, batch size, number of epochs, dropout, and normalization layer.

A large number of layers may introduce overfitting, vanishing, and exploding gradient problem while a lower number could introduce the large bias and perform less. In case of the number of hidden units no rule of thumb is available, but it should be less than the number of inputs. If the data size is smaller, a small number of hidden units could work better, the higher the hidden unit better the learning and feature representation capacity [47] in the complex problems. It is, in general, the tradeoff between high bias and variance. Rectified Linear Unit (ReLU), Sigmoid, tanh, and variant of ReLU such as LeakyReLU and ReLU6 [7] are the most common activation function. Deep network may be better with ReLU and its variant while Sigmoid/Tanh are recommended for shallow networks. Adam, and SGD are the most common optimizers. Adam is the most common for deep learning which converges faster than others, while SGD is better for shallow networks and AdaDelta and AdaGrad are recommended for

sparse datasets. If learning rate is high, the learning process might not converge to minima while if it is low, the model runs extremely slow. The choice of optimizer governs the learning rate, for SGD the most common is 0.1, while it is 0.001/0.01 for Adam [85]. The option of using the decay parameter is available to speed up the learning rate Initialization often works with the default configuration; however uniform initialization is preferred with ReLU. Xavier initialization is used to make sure that the variance of input and output remains equal which can prevent the signal to go too high or low [29]. Though, random weight initialization is especially faster in the pre-trained network, both supervised and unsupervised pre-training could initialize a network in a better way than random initialization [74]. Batch size is often recommended to be less than the number of sample size If it is small, the pattern would be less repeating and convergence could be difficult [96]. If it is high, the learning rate is slow as the number of iterations will increase. The recommended batch size is the power of 2 [25] The higher number of epochs may lead to the model to overfit that cannot generalize the problems and could lead to vanishing and exploding gradient problems. However, a lower number of epochs may limit the potential of the model, therefore it is the tradeoff between training time and performance [28]. The majority of the works have adopted 0.5 as the standard dropout value. Batch normalization minimizes the distribution shift that improves the optimization of the network. It stabilizes, speeds up the training process and sometimes improves the accuracy of the network by standardizing the input parameters.

4.3.2 Model calibration

Once set the hyperparameter, their optimizations in the deep learning are done in four ways (a) manual searches: a trial and error method, that is tuned manually, and one of the most common approaches, using this approach is easy but not guaranteed that the hypermeters are optimized; (b) random search: fit the model with the sample drawn randomly from the problem space, and perform the cross-validation such as k-fold validation, evaluate the model with optimization metrics and choose the best performing model; (c) grid search: an approach that systematically tests the multiple values of hypermeters by retraining the model for each value of the parameters automatically, it could be very slow if the number of hypermeters is high and (d) automated hypermeter tuning: Bayesian optimization method test model with different combinations of hyperparameters value, evaluate the results and choose the new set of hyperparameters based on the results, the process ends up with a set of possible hyperparameter and optimal model function [7]. The process can be automated

through different platforms like scikit-learn, Grid-SearchCV, and Hypers.

4.3.3 Model evaluation

The performance of the model needs to be accessed while train the model regularly. It provides a test platform for fine-tuning a model's parameters and selecting the best performing model – best set of hyperparameterization. A number of metrics for the model evaluation have been used to quantify the model performance, and it depends on the the given machine learning task such as classification, regression, ranking, clustering, among others [97]. Some metrics, such as precision-recall have been used for various task but classification accuracy, and confusion matrix, logarithmic loss, F-score are used for classification. But, root mean square error (RMSE), and mean absolute error (MAE).

4.4 Model validation

The fitted models need to be evaluated in order to confirm their reliability and confidence in their estimated results against the observed dataset. Two standard ways of model evaluation are found (i) holdout – using a separate hold or test sample that used after the end of model training to evaluate the performance with the unseen dataset [14] and (ii) k-fold cross-validation that allows to train the model with the multiple splits and estimates the overall accuracy by taking the average of the k individual accuracy.

The number of evaluation metrics is available either in holdout or in k-fold cross-validation to quantify the model performance has been implemented in different studies, it depends on the nature of the model and their applications. Based on the application and data nature two types of evaluation metrics are found: (i) classification metric – works with the classification problems, (ii) regression metric – works with the regression problem. The classification metric consists of confusion matrix, the most common method, overall accuracy, precision, recall, F1 score, the area under the curve (AUC), Kappa coefficients, scatter diagram are also frequently used approaches. All these metrics are applicable for both approaches holdout and k-fold cross-validation approach that is done multiple times, using different nonoverlapping train and test random splits of the dataset. Accuracy levels are averaged to report a synthetic performance measure of all cases [64]. To evaluate the model visual as well as the average silhouette width method to analyze the quality of the output which provided the compactness and separation of the clusters [17].

The coefficient of determination (r^2) is the most common evaluation metric for the regression problems, root

mean square error (RMSE), mean square error, mean absolute error is also commonly used measures. The r^2 describes how much of the variance between the estimated and real values is described by the linear fit. It varies between 0 and 1, the higher the value, the better the performance. While the RMSE is a quadratic scoring rule that measures the average quantity of error between the estimated value and real value. The MAE is the sum of the absolute differences between prediction and observed values.

5 Discussion

In this paper, we reviewed articles that used DL for remote sensing data. in several applications like image fusion, land cover classification, scene classification, disaster monitoring, image segmentation, change detection and feature detection. We have synthesized a generic framework to develop the DL model, which includes input/output preparation, architecture selection, hyper-parameterization and validation of the models.

Over the last couple of years, AI and DL methods had a transformative impact in a number of fields of geospatial data science such as segmentation, image fusion, and object detection among others. These techniques have already demonstrated the high potential to empower the next generation of geoinformatics. Location-based discovery and near human-level perception has been obtained with the analysis of relevant information obtained through satellite and field-based sensors.

A large number of papers under consideration focus on implementation of DL algorithms for remote sensing and prefer to use stable benchmark datasets such as RSSCN7, UC-Merced dataset and WHU-RS dataset but avoid using the real localized dataset [97]. Therefore, the focus has to shift from a mere implementation of model to the development of the training datasets and the model itself [7]. Transfer learning [71] has been heavily used due to its strength to train the model with limited number of the datasets and with low resources in a short time interval.

CNN is very popular due to its ability to extract features automatically from a huge set of data across the application without manual intervention. For example, Ammour et al. [57] has used a pre-trained CNN network for feature extraction and autoencoder for denoising automatically [57]. However, it is difficult to evaluate such automatically extracted features, as visualization in deep learning rarely goes beyond the second layer, which can represent only very basic features such as edge and gradient. Few trials have been done to visualize automatically extracted features in different applications). However, such works have not been found in the remote sensing datasets.

Additionally, unsupervised DL models that are capable to train without using the unlabeled dataset, might overcome the primary issue like limited availability of the training sample. Hence, it justifies the need for unsupervised DL models in RS.

Deep learning tasks such as segmentation and object detection are used in digital mapping like topographic mapping and cadastral mapping. Most of the articles reviewed in this paper target single features. However, in order to generate a complete digital topographic map, many features like building, roads, elevation, vegetation might be needed along with their names. Such an approach is missing in the literature. Semantic segmentation has been tried in some cases to identify all the features at once but this approach also lacks capacity to provide the textual information about the location such as road names without which the map is not complete. Hence, we can safely say that the current development of deep learning can generate segmentation and boundaries with some uncertainties, but it still lacks the capacity to generate geocoding and understanding topological relationships.

Figure 3a depicts the bib text analysis that shows the trend of the publications over the year while searching the key-words “deep learning” and “remote sensing” in Scopus. The deep learning-based publications using remote sensing-based dataset was published for the first time in the history in 2008 which has a very slow pace until 2015 after that it goes to an exponential rate. The Fig. 3b shows the cloud tag of the keywords used in the reviewed publications. Remote Sensing data, CNN model and application area like semantic segmentation, and object detection have been found most popular in the last decade. In these applications DL is expected to discover spatial concepts and inherit expert knowledge through the mathematic models. Although DL has successfully been applied in spatial applications such as object detection, scene detection, data fusion, their uses as a tool for the practical application is still not widely available. The major challenges include the availability of sufficient training samples, strong non-linearity, and low signal-to-noise ratios. We suggest to focus on above mentioned standing issues for the future research.

Availability of higher resolution spatial and temporal dataset is always a big issue in an application of remote sensing-based data due to the tradeoff between spatial resolution and the re-visit time of the satellite. Handling multi-source multi-temporal, multi-resolution and multi-platform, dataset is always a big challenge. One of the biggest advantages of DNN is its efficiency to make use of multi-sources dataset having various disparities to solve a problem. Even though, a large effort is required while preparing the multi-sources dataset to fit the model, it helps a lot to overcome the standing bottleneck of limited

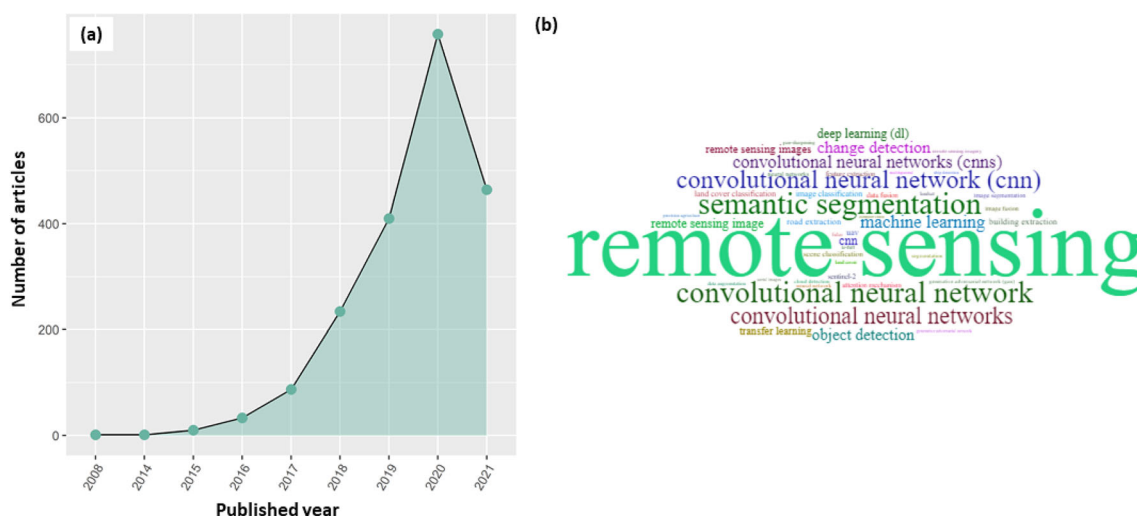


Fig. 3 Number of publications identified in Scopus with the keyword “deep learning” and “remote sensing” (a) and cloud tag obtained from the abstract and keyworks in the reviewed publication (b)

number of input samples. Apart from that, making use of multi-model dataset can also ease the data availability that helps to increase the quality and reliability of DL methods.

While considering the DL applications with spatial dataset, majority of the work was found in segmentation, clustering, object detection, scene detection, i.e. focused in the feature but not in the complete cycle of the geospatial analysis. The application of DL in the disaster monitoring is very limited. Hence, more exploration is necessary in this sector. Additionally, digital mapping would another promising application with the DL in future. Another frontier in the application of DL is the smart digitization. So far, the research is focused on extracting the geometric features and segmentation networks. The DL can be further leveraged to extract the topological relationship, geotagging, and help explainable digitization. Additionally, capacity of visualization of automatically extracted features in different applications) has not fully been explored with the spatial dataset. Therefore, further research is recommended to visualize and validate the automatically extracted features in spatial science.

Preparing manually labeled huge dataset for the model training and testing is a time-consuming and may not be feasible in some applications. Therefore, semi-supervised or unsupervised learning is required to overcome the dependency on label-based datasets. Similarly, to increase the reusability of a trained network for different datasets and task, transfer learning could be a very promising approach for the network re-usability. In which, optimized parameters for one dataset can be used as initialization of hyper-parameterization for learning with the different dataset; the only need to do is fine-tuning with real dataset from the specific application. That can save the huge

resources and time required for labelled data collection and train the network with random hyper-parameterization.

After a very impressive performance of the DL, an explainable DL model is required in order to widespread acceptance and building trust. The clear-box DL could help understand how the model process input dataset and explain the prediction and gain access to the process rather than the output alone. In many sectors, people are already working intensively for the clear-box DL models however, a very limited papers are found in the remote sensing application in this direction. Therefore, a whitebox/gray box DL is strongly proposed in coming days.

6 Conclusion

In this review, methods and application areas of DL approaches using geospatial datasets are critically analyzed and a general-purpose DL framework is synthesized. Recent past and, future trends are discussed for the readership who has embarked on their journey in the DL-based application with the spatial dataset. Naturally, DL methods have created both opportunities and challenges in the geospatial frontier. It has already been implemented in a number of applications yielding promising results, however, more advanced DL approaches and their application must now be explored. Data, method and application are the three prominent aspects found while summering the roadmap for applying DL in different geospatial tasks. Firstly, the possibility of using a multi-model dataset should be well explored. Semi-supervised or unsupervised approaches should be well considered to reduce the dependency on the labor-intensive dataset labeling activity. Enough focus is needed for, model re-usability – transfer

function and model transparency with white-box model development. Finally, the implementation of DL for smart digital mapping and relatively less explored and highly important domain of disaster management has a high potential. The strength of DL in spatial science has not been fully leveraged yet, so it has a fair chance to dominate the research for a couple of decades. Future researchers are expected to unfold further strengths of DL.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

- Ma, F., Gao, F., Sun, J., Zhou, H., & Hussain, A. (2019). Attention graph convolution network for image segmentation in big SAR imagery data. *Remote Sensing*, *11*(21), 1–21. <https://doi.org/10.3390/rs11212586>
- Sedona, R., Cavallaro, G., Jitsev, J., Strube, A., Riedel, M., & Benediktsson, J. A. (2019). Remote sensing big data classification with high performance distributed deep learning. *Remote Sensing*, *11*(24), 1–19. <https://doi.org/10.3390/rs11243056>
- Lary, D. J., Alavi, A. H., Gandomi, A. H., & Walker, A. L. (2016). Machine learning in geosciences and remote sensing. *Geoscience Frontiers*, *7*(1), 3–10. <https://doi.org/10.1016/j.gsf.2015.07.003>
- Adarme, M. O., Feitosa, R. Q., Happ, P. N., De Almeida, C. A., & Gomes, A. R. (2020). Evaluation of deep learning techniques for deforestation detection in the brazilian amazon and cerrado biomes from remote sensing imagery. *Remote Sensing*. <https://doi.org/10.3390/rs12060910>
- Ghimire, B. R., Nagai, M., Tripathi, N. K., Witayangkurn, A., Mishara, B., & Sasaki, N. (2017). Mapping of Shorea robusta forest using time series MODIS data. *Forests*. <https://doi.org/10.3390/f8100384>
- Yuan, Q., Shen, H., Li, T., Li, Z., Li, S., Jiang, Y., Xu, H., Tan, W., Yang, Q., Wang, J., Gao, J., & Zhang, L. (2020). Deep learning in environmental remote sensing: Achievements and challenges. *Remote Sensing of Environment*, *241*, 111716. <https://doi.org/10.1016/j.rse.2020.111716>
- Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., & Johnson, B. A. (2019). Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, *152*, 166–177. <https://doi.org/10.1016/j.isprsjprs.2019.04.015>
- Zhu, X. X., Tuia, D., Mou, L., Xia, G. S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, *5*(4), 8–36. <https://doi.org/10.1109/MGRS.2017.2762307>
- He, W., & Yokoya, N. (2018). Multi-temporal sentinel-1 and -2 data fusion for optical Image Simulation. *ISPRS International Journal of Geo-Information*, *7*(10), 1–11. <https://doi.org/10.3390/ijgi7100389>
- Mishra, B., & Shahi, T. B. (2021). Deep learning-based framework for spatiotemporal data fusion: An instance of Landsat 8 and Sentinel 2 NDVI. *Journal of Applied Remote Sensing*, *15*(03), 1–13. <https://doi.org/10.1117/1.jrs.15.034520>
- Nezami, S., Khoramshahi, E., Nevalainen, O., Pölonen, I., & Honkavaara, E. (2020). Tree species classification of drone hyperspectral and RGB imagery with deep learning convolutional neural networks. *Remote Sensing*. <https://doi.org/10.3390/rs12071070>
- Mishra, B., Susaki, J., & Engineering, E. R. (2014). Optical and sar data integration for automatic change pattern detection. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. <https://doi.org/10.5194/isprsannals-II-7-39-2014>
- Cao, R., Tu, W., Yang, C., Li, Q., Liu, J., Zhu, J., Zhang, Q., Li, Q., & Qiu, G. (2020). Deep learning-based remote and social sensing data fusion for urban region function recognition. *ISPRS Journal of Photogrammetry and Remote Sensing*, *163*, 82–97. <https://doi.org/10.1016/j.isprsjprs.2020.02.014>
- Chaib, S., Liu, H., Gu, Y., & Yao, H. (2017). Deep feature fusion for VHR remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, *55*(8), 4775–4784. <https://doi.org/10.1109/TGRS.2017.2700322>
- Shao, Z., & Cai, J. (2018). Remote sensing image fusion with deep convolutional neural network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *11*(5), 1656–1669. <https://doi.org/10.1109/JSTARS.2018.2805923>
- Yuan, Q., Wei, Y., Meng, X., Shen, H., & Zhang, L. (2018). A multiscale and multidepth convolutional neural network for remote sensing imagery pan-sharpening. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *11*(3), 978–989. <https://doi.org/10.1109/JSTARS.2018.2794888>
- Lahaye, N., Ott, J., Garay, M. J., El-Askary, H. M., & Linstead, E. (2019). Multi-modal object tracking and image fusion with unsupervised deep learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *12*(8), 3056–3066. <https://doi.org/10.1109/JSTARS.2019.2920234>
- Yu, Y., Gong, Z., Wang, C., & Zhong, P. (2017). An unsupervised convolutional feature fusion network for deep representation of remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, *15*(1), 23–27. <https://doi.org/10.1109/LGRS.2017.2767626>
- Cheng, D., Meng, G., Xiang, S., & Pan, C. (2017). FusionNet: Edge aware deep convolutional networks for semantic segmentation of remote sensing harbor images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(12), 5769–5783. <https://doi.org/10.1109/JSTARS.2017.2747599>
- Liao, W., Van Coillie, F., Gao, L., Li, L., Zhang, B., & Chanussot, J. (2018). Deep learning for fusion of APEX hyperspectral and full-waveform LiDAR remote sensing data for tree species mapping. *IEEE Access*, *6*(c), 68716–68729. <https://doi.org/10.1109/ACCESS.2018.2880083>
- Feng, Q., Zhu, D., Yang, J., & Li, B. (2019). Multisource hyperspectral and LiDAR data fusion for urban land-use mapping based on a modified two-branch convolutional neural network. *ISPRS International Journal of Geo-Information*. <https://doi.org/10.3390/ijgi8010028>
- Antoniou, V., & Potsiou, C. (2020). A deep learning method to accelerate the disaster response process. *Remote Sensing*. <https://doi.org/10.3390/rs12030544>
- Sun, Y., Zeng, Q., Geng, B., Lin, X., Sude, B., & Chen, L. (2019). Deep learning architecture for estimating hourly ground-level PM 2.5 using satellite remote sensing. *IEEE Geoscience and Remote Sensing Letters*, *16*(9), 1343–1347. <https://doi.org/10.1109/Lgrs.2019.2900270>
- Ma, H., Liu, Y., Ren, Y., & Yu, J. (2020). Detection of collapsed buildings in post-earthquake remote sensing images based on the improved YOLOv3. *Remote Sensing*. <https://doi.org/10.3390/rs12010044>
- Liu, Y., & Wu, L. (2016). Geological disaster recognition on optical remote sensing images using deep learning. *Procedia*

- Computer Science*, 91(Itqm), 566–575. <https://doi.org/10.1016/j.procs.2016.07.144>
26. Ye, C., Li, Y., Cui, P., Liang, L., Pirasteh, S., Marcato, J., Goncalves, W. N., & Li, J. (2019). Landslide detection of hyperspectral remote sensing data based on deep learning with constrains. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(12), 5047–5060. <https://doi.org/10.1109/JSTARS.2019.2951725>
 27. Zhao, W., Wang, Z., Gong, M., & Liu, J. (2017). Discriminative feature learning for unsupervised change detection in heterogeneous images based on a coupled neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 55(12), 7066–7080. <https://doi.org/10.1109/TGRS.2017.2739800>
 28. Baur, J., Steinberg, G., Nikulin, A., Chiu, K., & de Smet, T. S. (2020). Applying deep learning to automate UAV-based detection of scatterable landmines. *Remote Sensing*, 12(5), 1–16. <https://doi.org/10.3390/rs12050859>
 29. Zeng, K., & Wang, Y. (2020). A deep convolutional neural network for oil spill detection from spaceborne SAR images. *Remote Sensing*. <https://doi.org/10.3390/rs12061015>
 30. Shen, R., Huang, A., Li, B., & Guo, J. (2019). Construction of a drought monitoring model using deep learning based on multi-source remote sensing data. *International Journal of Applied Earth Observation and Geoinformation*, 79(219), 48–57. <https://doi.org/10.1016/j.jag.2019.03.006>
 31. Weng, Q., Mao, Z., Lin, J., & Guo, W. (2017). Land-use classification via extreme learning classifier based on deep convolutional features. *IEEE Geoscience and Remote Sensing Letters*, 14(5), 704–708.
 32. Zou, X., Cheng, M., Wang, C., Xia, Y., & Li, J. (2017). Tree classification in complex forest point clouds based on deep learning. *IEEE Geoscience and Remote Sensing Letters*, 14(12), 2360–2364.
 33. Liu, X., Zhou, Y., Zhao, J., Yao, R., Liu, B., & Zheng, Y. (2019). Siamese convolutional neural networks for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 16(8), 1200–1204.
 34. Zou, Q., Ni, L., Zhang, T., & Wang, Q. (2015). Deep learning based feature selection for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 12(11), 2321–2325.
 35. Boualleg, Y., Farah, M., & Farah, I. R. (2019). Remote sensing scene classification using convolutional features and deep forest classifier. *IEEE Geoscience and Remote Sensing Letters*, 16(12), 1944–1948. <https://doi.org/10.1109/LGRS.2019.2911855>
 36. Cheng, G., Yang, C., Yao, X., Guo, L., & Han, J. (2018). When deep learning meets metric learning: remote sensing image scene classification via learning discriminative CNNs. *IEEE Transactions on Geoscience and Remote Sensing*, 56(5), 2811–2821. <https://doi.org/10.1109/TGRS.2017.2783902>
 37. Dong, Y., & Zhang, Q. (2019). A combined deep learning model for the scene classification of high-resolution remote sensing image. *IEEE Geoscience and Remote Sensing Letters*, 16(10), 1540–1544. <https://doi.org/10.1109/LGRS.2019.2902675>
 38. Gong, Z., Zhong, P., Yu, Y., & Hu, W. (2018). Diversity-promoting deep structural metric learning for remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 56(1), 371–390. <https://doi.org/10.1109/TGRS.2017.2748120>
 39. Gu, X., Angelov, P. P., Zhang, C., & Atkinson, P. M. (2018). A massively parallel deep rule-based ensemble classifier for remote sensing scenes. *IEEE Geoscience and Remote Sensing Letters*, 15(3), 345–349. <https://doi.org/10.1109/LGRS.2017.2787421>
 40. Han, W., Feng, R., Wang, L., & Cheng, Y. (2018). A semi-supervised generative framework with deep learning features for high-resolution remote sensing image scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145, 23–43.
 41. Liu, Y., Zhong, Y., & Qin, Q. (2018). Scene classification based on multiscale convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 56(12), 7109–7121.
 42. Li, Y., Zhang, Y., Huang, X., Zhu, H., & Ma, J. (2017). Large-scale remote sensing image retrieval by deep hashing neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 56(2), 950–965.
 43. Zhang, L., & Zhang, L. (2017). Deep learning-based classification and reconstruction of residential scenes from large-scale point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 56(4), 1887–1897.
 44. Li, Y., Zhang, Y., Huang, X., & Ma, J. (2018). Learning source-invariant deep hashing convolutional neural networks for cross-source remote sensing image retrieval. *IEEE Transactions on Geoscience and Remote Sensing*, 56(11), 6521–6536.
 45. Cresson, R. (2019). A framework for remote sensing images processing using deep learning techniques. *IEEE Geoscience and Remote Sensing Letters*, 16(1), 25–29. <https://doi.org/10.1109/LGRS.2018.2867949>
 46. Zhang, H., Ni, W., Yan, W., Xiang, D., Wu, J., Yang, X., & Bian, H. (2019). Registration of multimodal remote sensing image based on deep fully convolutional neural network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(8), 3028–3042.
 47. Liu, B., Yu, X., Yu, A., Zhang, P., Wan, G., & Wang, R. (2018). Deep few-shot learning for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(4), 2290–2304.
 48. Wang, R., Meng, Y., Zhang, W., Li, Z., Hu, F., & Meng, L. (2019). Remote sensing semantic segregation for water information extraction: Optimization of samples via training error performance. *IEEE Access*, 7, 13383–13395. <https://doi.org/10.1109/ACCESS.2019.2894099>
 49. Gong, M., Zhan, T., Zhang, P., & Miao, Q. (2017). Superpixel-based difference representation learning for change detection in multispectral remote sensing images. *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*, 55(5), 16–16. <https://doi.org/10.2307/j.ctt1ffjff6.16>
 50. Huang, F., Yu, Y., & Feng, T. (2019). Automatic extraction of urban impervious surfaces based on deep learning and multi-source remote sensing data. *Journal of Visual Communication and Image Representation*, 60, 16–27. <https://doi.org/10.1016/j.jvcir.2018.12.051>
 51. Li, W., Liu, H., Wang, Y., Li, Z., Jia, Y., & Gui, G. (2019). Deep learning-based classification methods for remote sensing images in urban built-up areas. *IEEE Access*, 7, 36274–36284. <https://doi.org/10.1109/ACCESS.2019.2903127>
 52. Chen, Y., Lin, Z., Zhao, X., Wang, G., & Gu, Y. (2014). Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7, 1–14.
 53. Ayhan, B., & Kwan, C. (2020). Tree, shrub, and grass classification using only RGB images. *Remote Sensing*, 12, 1333.
 54. Liu, Q., Xiang, X., Wang, Y., Luo, Z., & Fang, F. (2020). Aircraft detection in remote sensing image based on corner clustering and deep learning. *Engineering Applications of Artificial Intelligence*, 87, 103333. <https://doi.org/10.1016/j.engappai.2019.103333>
 55. Marmanis, D., Datcu, M., Esch, T., & Stilla, U. (2016). Deep learning earth observation classification using imagenet pretrained networks. *IEEE Geoscience and Remote Sensing Letters*, 13(1), 105–109.
 56. Zhang, C., Harrison, P. A., Pan, X., Li, H., Sargent, I., & Atkinson, P. M. (2020). Scale sequence joint deep learning (SS-

- JDL) for land use and land cover classification. *Remote Sensing of Environment*, 237, 111593. <https://doi.org/10.1016/j.rse.2019.111593>
57. Ammour, N., Bashmal, L., Bazi, Y., Member, S., Rahhal, M. M. Al, & Zuair, M. (2018). Asymmetric Adaptation of Deep Features for Cross-Domain Classification in Remote Sensing Imagery. *IEEE Geoscience and Remote Sensing Letters* 1–5.
 58. Tong, X.-Y., Xia, G.-S., Lu, Q., Shen, H., Li, S., You, S., & Zhang, L. (2020). Land-cover classification with high-resolution remote sensing images using transferable deep models. *Remote Sensing of Environment*, 237, 111322. <https://doi.org/10.1016/j.rse.2019.111322>
 59. Kussul, N., Lavreniuk, M., Skakun, S., & Shelestov, A. (2017). Deep learning classification of land cover and. *IEEE Geoscience and Remote Sensing Letters*, 14(5), 778–782.
 60. Huang, K., Nie, W., & Luo, N. (2019). Fully polarized SAR imagery classification based on deep reinforcement learning method using multiple polarimetric features. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. <https://doi.org/10.1109/JSTARS.2019.2913445>
 61. Mullissa, A. G., Persello, C., & Stein, A. (2019). PolSARNet: A deep fully convolutional network for polarimetric sar image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. <https://doi.org/10.1109/JSTARS.2019.2956650>
 62. Romero, A., Gatta, C., & Camps-Valls, G. (2015). Unsupervised deep feature extraction for remote sensing image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 1–14.
 63. Zhang, C., Li, G., & Du, S. (2019). Multi-scale dense networks for hyperspectral remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(11), 9201–9222. <https://doi.org/10.1109/TGRS.2019.2925615>
 64. Hamida, A. Ben, Benoit, A., Lambert, P., & Amar, C. Ben. (2018). 3-D Deep learning approach for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 56, 1–15.
 65. Lakhal, M. I., Cevikalp, H., Escalera, S., & Ofli, F. (2018). Recurrent neural networks for remote sensing image classification. *IET Computer Vision*, 12, 1040–1045. <https://doi.org/10.1049/iet-cvi.2017.0420>
 66. Huang, F., Yu, Y., & Feng, T. (2019). Hyperspectral remote sensing image change detection based on tensor and deep learning. *Journal of Visual Communication and Image Representation*, 58, 233–244. <https://doi.org/10.1016/j.jvcir.2018.11.004>
 67. de Bem, P. P., de Carvalho, O. A., Guimarães, R. F., & Gomes, R. A. T. (2020). Change detection of deforestation in the Brazilian Amazon using Landsat data and convolutional neural networks. *Remote Sensing*. <https://doi.org/10.3390/rs12060901>
 68. Song, A., & Choi, J. (2020). Fully convolutional networks with multiscale 3D filters and transfer learning for change detection in high spatial resolution satellite images. *Remote Sensing*, 12. <https://doi.org/10.3390/rs12050799>
 69. Yang, M., Jiao, L., Liu, F., Hou, B., & Yang, S. (2019). Transferred deep learning-based change detection in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9), 6960–6973. <https://doi.org/10.1109/TGRS.2019.2909781>
 70. Kemker, R., Salvaggio, C., & Kanan, C. (2018). Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145, 60–77. <https://doi.org/10.1016/j.isprsjprs.2018.04.014>
 71. Yan, Y., Yan, H., Guo, J., & Dai, H. (2020). Classification and segmentation of mining area objects in large-scale sparse lidar point cloud using a novel rotated density network. *ISPRS International Journal of Geo-Information*. <https://doi.org/10.3390/ijgi9030182>
 72. Huang, F., Yu, Y., & Feng, T. (2019). Automatic building change image quality assessment in high resolution remote sensing based on deep learning. *Journal of Visual Communication and Image Representation*, 63, 102585. <https://doi.org/10.1016/j.jvcir.2019.102585>
 73. Yang, M., Tseng, H., Hsu, Y., & Tsai, H. P. (2020). Semantic segmentation using deep learning with vegetation indices for rice lodging identification in multi-date UAV visible images. *Remote Sensing*. <https://doi.org/10.3390/rs12040633>
 74. Pierdicca, R., Paolanti, M., Matrone, F., Martini, M., Morbidoni, C., Malinverni, E. S., Frontoni, E., & Lingua, A. M. (2020). Point cloud semantic segmentation using a deep learning framework for cultural heritage. *Remote Sensing*, 12(6), 1–23. <https://doi.org/10.3390/rs12061005>
 75. López, J., Torres, D., Santos, S., & Atzberger, C. (2020). Spectral imagery tensor decomposition for semantic segmentation of remote sensing data through fully convolutional networks. *Remote Sensing*, 12(3), 1–21. <https://doi.org/10.3390/rs12030517>
 76. Audebert, N., Le Saux, B., & Lefèvre, S. (2018). Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 140, 20–32. <https://doi.org/10.1016/j.isprsjprs.2017.11.011>
 77. Wu, W., Li, H., Li, X., Guo, H., & Zhang, L. (2019). PolSAR image semantic segmentation based on deep transfer learning - realizing smooth classification with small training sets. *IEEE Geoscience and Remote Sensing Letters*, 16(6), 977–981. <https://doi.org/10.1109/LGRS.2018.2886559>
 78. Henry, C., Azimi, S. M., & Merkle, N. (2018). Road segmentation in SAR satellite images with deep fully convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 15(12), 1867–1871. <https://doi.org/10.1109/LGRS.2018.2864342>
 79. Li, Z., Wang, R., Zhang, W., Hu, F., & Meng, L. (2019). Multiscale features supported deeplabv3+ optimization scheme for accurate water semantic segmentation. *IEEE Access*, 7, 155787–155804. <https://doi.org/10.1109/ACCESS.2019.2949635>
 80. Li, K., Hu, X., Jiang, H., Shu, Z., & Zhang, M. (2020). Attention-guided multi-scale segmentation neural network for interactive extraction of region objects from high-resolution satellite imagery. *Remote Sensing*. <https://doi.org/10.3390/rs12050789>
 81. Lu, X., Zhong, Y., Zheng, Z., Liu, Y., Zhao, J., Ma, A., & Yang, J. (2019). Multi-scale and multi-task deep learning framework for automatic road extraction. *IEEE Transactions on Geoscience and Remote Sensing*, 57(11), 9362–9377. <https://doi.org/10.1109/TGRS.2019.2926397>
 82. Chen, G., Zhang, X., Wang, Q., Dai, F., Gong, Y., & Zhu, K. (2018). Symmetrical dense-shortcut deep fully convolutional networks for semantic segmentation of very-high-resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(5), 1633–1644. <https://doi.org/10.1109/JSTARS.2018.2810320>
 83. Mi, L., & Chen, Z. (2020). Superpixel-enhanced deep neural forest for remote sensing image semantic segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 159, 140–152. <https://doi.org/10.1016/j.isprsjprs.2019.11.006>
 84. Yao, X., Han, J., Cheng, G., Qian, X., & Guo, L. (2016). Semantic annotation of high-resolution satellite images via weakly supervised learning. *IEEE Transactions on Geoscience and Remote Sensing*, 54(6), 3660–3671. <https://doi.org/10.1109/TGRS.2016.2523563>
 85. Bachhofner, S., Loghini, A., Otepka, J., Pfeifer, N., Schiller, N., Köhler, O., & Hochreiter, R. (2020). Generalized sparse convolutional neural networks for semantic segmentation of point clouds derived from tri-stereo satellite imagery. *Remote Sensing*. <https://doi.org/10.3390/rs12081289>

86. Miyoshi, G. T., Arruda, M. D. S., Osco, L. P., Junior, J. M., Gonçalves, D. N., Imai, N. N., Tommaselli, A. M. G., Honkavaara, E., & Gonçalves, W. N. (2020). A novel deep learning method to identify single tree species in UAV-based hyperspectral images. *Remote Sensing*, *12*(8), 1–18. <https://doi.org/10.3390/RS12081294>
87. Braga, J. R. G., Peripato, V., Dalagnol, R., Ferreira, M. P., Tarabalka, Y., Aragão, L. E. O. C., de Campos Velho, H. F., Shigemori, E. H., & Wagner, F. H. (2020). Tree crown delineation algorithm based on a convolutional neural network. *Remote Sensing*, *12*(8), 1–27. <https://doi.org/10.3390/RS12081288>
88. Cheng, G., Zhou, P., & Han, J. (2016). Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, *54*(12), 7405–7415. <https://doi.org/10.1109/TGRS.2016.2601622>
89. Ji, S., Wei, S., & Lu, M. (2019). Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(1), 574–586. <https://doi.org/10.1109/TGRS.2018.2858817>
90. Hui, J., Du, M., Ye, X., Qin, Q., & Sui, J. (2019). Effective building extraction from high-resolution remote sensing images with multitask driven deep neural network. *IEEE Geoscience and Remote Sensing Letters*, *16*(5), 786–790. <https://doi.org/10.1109/LGRS.2018.2880986>
91. Li, Z., Shen, H., Cheng, Q., Liu, Y., You, S., & He, Z. (2019). Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors. *ISPRS Journal of Photogrammetry and Remote Sensing*, *150*, 197–212. <https://doi.org/10.1016/j.isprsjprs.2019.02.017>
92. Maier, H. R., Jain, A., Dandy, G. C., & Sudheer, K. P. (2010). Methods used for the development of neural networks for the prediction of water resource variables in river systems: Current status and future directions. *Environmental Modelling and Software*, *25*(8), 891–909. <https://doi.org/10.1016/j.envsoft.2010.02.003>
93. Liu, Y., Chen, X., Peng, H., & Wang, Z. (2017). Multi-focus image fusion with a deep convolutional neural network. *Information Fusion*, *36*, 191–207. <https://doi.org/10.1016/j.inffus.2016.12.001>
94. Aljawarneh, S., Aldwairi, M., & Yassein, M. B. (2018). Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computational Science*, *25*, 152–160. <https://doi.org/10.1016/j.jocs.2017.03.006>
95. Alam, F. I., Zhou, J., Member, S., Liew, A. W., Member, S., Jia, X., Member, S., & Chanussot, J. (2018). Conditional random field and deep feature learning for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*. <https://doi.org/10.1109/TGRS.2018.2867679>
96. Helber, P., Bischke, B., Dengel, A., & Borth, D. (2019). EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *12*(7), 2217–2226.
97. Scott, G. J., England, M. R., Starms, W. A., Marcum, R. A., & Davis, C. H. (2017). Training deep convolutional neural networks for land – cover classification of high-resolution imagery. *IEEE Geoscience and Remote Sensing Letters*, 1–5.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.