Scale Sequence Joint Deep Learning (SS-JDL) for land use and land cover classification

Ce Zhang [a, b*], Paula A. Harrison [b], Xin Pan [c, d], Huapeng Li [e], Isabel Sargent [f], Peter M. Atkinson [g, h, i, j*]

[a] Lancaster Environment Centre, Lancaster University, Lancaster LA1 4YQ, UK; [b] Centre for Ecology & Hydrology, Library Avenue, Bailrigg, Lancaster LA1 4AP, UK; [c] School of Computer Technology and Engineering, Changchun Institute of Technology, 130012 Changchun, China; [d] The Key Laboratory of Changbai Mountain Historical Culture and VR Technology Reconfiguration, Changchun Institute of Technology, 130012 Changchun, China; [e] Northeast Institute of Geography and Agroecology, Chinese Academy of Sciences, Changchun 130102, China; [f] Ordnance Survey, Adanac Drive, Southampton SO16 0AS, UK; [g] Faculty of Science and Technology, Lancaster University, Lancaster LA1 4YR, UK; [h] School of Natural and Built Environment, Queen's University Belfast, Belfast BT7 1NN, Northern Ireland, UK; [i] Geography and Environmental Science, University of Southampton, Highfield, Southampton SO17 1BJ, UK; [j] Institute of Geographic Science and Natural Resources Research, Chinese Academy of Sciences, 11A Datun Road, Beijing 100101, China

Abstract Choosing appropriate scales for remotely sensed image classification is extremely important yet still an open question in relation to deep convolutional neural networks (CNN), due to the impact of spatial scale (i.e., input patch size) on the recognition of ground objects. Currently, the optimal scale selection processes are extremely cumbersome and time-consuming requiring repetitive experiments involving trial-and-error procedures, which significantly reduces the practical utility of the corresponding classification methods. This issue is crucial when trying to classify large-scale land use (LU) and land cover (LC) jointly (Zhang *et al.*, 2019). In this paper, a simple and parsimonious scale sequence joint deep learning (SS-JDL) method is proposed for joint LU and LC classification, in which a sequence of scales is embedded in the iterative process of fitting the joint distribution implicit in the joint deep learning (JDL) method, thus, replacing the previous paradigm of scale selection. The sequence of scales, derived autonomously and used to define the CNN input patch sizes, provides consecutive information transmission from small-scale features to large-scale representations, and from simple LC states to complex LU characterisations. The effectiveness of the novel SS-JDL method was tested on aerial digital photography of three complex and heterogeneous landscapes, two in Southern England (Bournemouth and Southampton) and one in North West England (Manchester). Benchmark comparisons were provided in the form of a range of LU and LC methods, including the state-of-the-art joint deep learning (JDL) method. The experimental results demonstrated that the SS-JDL consistently outperformed all of the state-of-the-art baselines in terms of both LU and LC classification accuracies, as well as computational efficiency. The proposed SS-JDL method, therefore, represents a fast and effective implementation of the state-of-the-art JDL method. By creating a single, unifying joint distribution framework for classifying higher order feature representations, including LU, the SS-JDL method has the potential to transform the classification paradigm in remote sensing, and in machine learning more generally.

Keywords: multi-scale deep learning; optimal scale selection; convolutional neural network; joint classification; hierarchical representations

1 Introduction

Land use and land cover (LULC) information is essential for diverse applications in geospatial domain, such as urban and regional planning, environmental monitoring and management (Liu *et al.*, 2017, Zhang *et al.*, 2019). LULC information can also provide insights to tackle a multitude of socioeconomic and environmental challenges, including food insecurity, poverty, climate change and disaster risk (Stürck *et al.*, 2015). Recent advances in sensor technologies have led to a constellation of satellite and airborne platforms, from which a large amount of very fine spatial resolution (VFSR) remotely sensed imagery is available commercially. While great opportunities are offered by VFSR imagery to capture fine-grained LULC detail, information extraction and retrieval is still immature and inefficient, primarily undertaken by means of traditional field survey and manual interpretation (Hu and Wang, 2013). Such routine tasks are labour-intensive and time-consuming. At the same time, our environment is constantly changing requiring frequent updates of LULC information to support scientific decision-making. It is, therefore, of paramount importance to develop highly efficient and effective techniques to derive LULC information in an automatic and intelligent fashion.

Over the past twenty years, significant efforts have been made towards the automation of LULC classification methods using VFSR images. Traditional techniques can be categorised into pixel-based and object-based approaches. Pixel-based methods focus on classifying individual pixels based on spectral reflectance, which often result in speckle noise effects with limited classification accuracy, given the spectral and spatial complexity presented in VFSR remotely sensed imagery. Textures (Herold *et al.*, 2003) and contextual information (Wu *et al.*, 2009) can be integrated to characterise spatial patterns using moving kernels or windows. These approaches, however, are built on arbitrarily structured images (e.g. squares), whereas real world objects are often irregularly shaped and structured in specific patterns (Herold *et al.*, 2003). Object-based methods are now adopted widely for LULC image classification based on segmented objects (group of pixels), thereby allowing the extraction of discriminative features (e.g., spectral, texture, shape) within the objects and contextual information between adjacent regions. However, those object-based approaches are often challenged by selecting appropriate segmentation scales to achieve meaningful objects (e.g., particular land cover categories), with under- and over-segmentation occurring within the single image (Ming *et al.*, 2015). Besides, the extracted features that characterise the objects are essentially hand-coded via feature engineering, which is subject to individual user experience and expertise, making it difficult to achieve comparable results when transferring the classifier to other datasets. Additionally, the spatial configurations of land use objects can be extremely difficult to hand-code into explicit features, thus, limiting representation and discrimination through traditional methods. Moreover, traditional methods lack a clear definition of the classification hierarchy (i.e. the level of representations of the landscape) and LULC classes are often used interchangeably in remotely sensed image classification. Ontologically, however, land cover (LC) and land use (LU) are manifested at different levels of representation: LC represents low-level states whereas LU characterises high-level functions of the landscape.

Recently, deep learning-based methods have attracted enormous interest in the field of pattern recognition and computer vision, owing to their capability to learn the most representative and discriminative features hierarchically in an end-to-end fashion (Arel *et al.*, 2010). Deep convolutional neural network (CNN), as a popular deep learning method, has achieved significant breakthroughs in image processing and analysis (Krizhevsky *et al.*, 2012), with impressive results beyond the state-of-the-art in a variety of disciplines, not only in classical computer vision fields such as visual recognition, target detection and robotics, but also in many other practical applications (Hu *et al.*, 2015; Nogueira *et al.*, 2017). In the remotely sensed domain, the CNN has shown huge potential in diverse tasks through high-level feature representations, such as road extraction (Cheng *et al.*, 2017), vehicle detection (Dong *et al.*, 2015), scene classification (Liu et al., 2018), semantic segmentation (Wang *et al.*, 2017), and LULC image classification (Zhang *et al.*, 2018a; 2018b).

Within a CNN network, a patch-based architecture is used to learn and extract higher-level features in image patches autonomously through a hierarchy of filters. As a consequence, the choice of image patch size, as a key CNN parameter, has a significant influence on the scale of representations that are manifested over the landscape and, consequently, the accuracy of remotely sensed image classification. These scales are also dependent on the definition of the LULC classification hierarchy, which is unclear so far. Therefore, the determination of the CNN scale for a specific LULC classification task is still an open question in the remote sensing community, and a common approach is to consider scale variations, that is, not constrain to a single scale representation (Pan and Zhao, 2018). Previous research has attempted to incorporate multiple scales into the CNN network to improve spatial feature representations across different scales (e.g., Lv *et al.*, 2018; Yang *et al.*, 2018; Zhang *et al.*, 2018b). For example, a set of CNNs with different patch sizes and scales were integrated by Deng *et al.* (2018) and Liu *et al.* (2018) to enhance feature representations across multiple scales, thereby achieving increased accuracy of scene classification. Yang *et al.* (2018) utilised multi-scale CNNs to differentiate complex scenes (e.g., airport, residential, commercial) in remotely sensed imagery, and demonstrated increased accuracy compared with single-scale CNN networks. Deep features at a range of scales have also been embedded into the CNN to identify vehicles (e.g., ships, cars) within remotely sensed scenes, leading to increased accuracy of target detection (Li *et al.*, 2018). In remotely sensed image classification, Lv *et al.* (2018) combined region-based CNNs at multiple scales to differentiate land cover objects with high accuracy and efficiency. In addition, object-based CNNs comprising of two distinctive scales were developed to solve the complex land *use* classification task (Zhang *et al.*, 2018b). Finally, deep features at multiple scales were extracted through CNN networks, and used to boost land cover classification accuracy for hyperspectral images (He *et al.*, 2019). A challenge for these multi-scale CNN techniques, however, is to determine the optimal scales (patch sizes) from a large sampling space that is extremely difficult to explore exhaustively across the full range of scales.

In summary, current LULC classification approaches (both traditional and deep learning methods) suffer from two major issues: (1) definition of the classification hierarchy; and (2) definition of the optimal scale to represent the landscape. In terms of the classification hierarchy, land use (LU) and land cover (LC) are often defined interchangeably, without differentiating their intrinsic differences in semantic meaning. LC represents the physical characteristics of the Earth's surface, whereas LU is defined as a higher-order function within a particular space through a mosaic of different LC categories. The spatially nested and hierarchical relationships between LU and LC are given little consideration in LULC image classification, except for the recently proposed joint deep learning (JDL) method (Zhang *et al.*, 2019). As for the choice of scale, it is challenging to determine an optimal scale that can represent the entire scene of a complex and heterogeneous landscape, and multi-scale feature representations are often incorporated to capture large or small land features over different scales. These multiple scales are searched exhaustively through trial and error and tested through extensive experiments with different combinations of candidate scales (Kim *et al.*, 2011; Ming *et al.*, 2015). For deep learning methods (e.g., CNN), such scale parameterisation processes are extremely time-consuming with a large amount of CNN model training. The process can be labour-intensive with repetitive experiments, especially for joint LU and LC classification such as through the JDL method. Furthermore, the selected multiple scales are considered independently as individual evidence to support integrated decisions, which do not capture the mutual connections among the different scales. As such, these scale selection processes are far from operational for deep learning in remotely sensed image classification.

The objective of this research was to develop an automatic approach that is applicable in engineering practices to model the nested relationships between LU and LC, with the ability to address scale issues effectively and efficiently in remotely sensed image classification. A novel scale sequence joint deep learning (SS-JDL) method for LU and LC classification is proposed, in which, scales (input patch sizes) of the CNN networks are autonomously derived as a sequence of representations. The scale sequence is designed to mimic the human cognition of image pattern recognition through continuously increasing scales, with information transmission between neighbouring scales from small-scale features to large-scale visual representations. The SS-JDL has the key advantage that it is simple and parsimonious in the way that it constructs the sequence of scales and determines an efficient solution, such that the cumbersome and time-consuming process of optimal scale selection is avoided. The rest of the paper is organized as follows: the proposed method is detailed in section 2; followed by experiments and results analysis in section 3; discussions and conclusions are made in section 4 and 5, respectively.

2 Methods

2.1 Multilayer perceptron (MLP)

A multilayer perceptron (MLP) is a feed forward neural network that transforms the input data (e.g., image pixels) into the output representations (e.g., LC labels) (Atkinson and Tatnall, 1997). Typically, a MLP is composed of input, hidden, and output layers with computational nodes fully connected by weights and biases (Del Frate *et al.*, 2007). These weights and biases are learned through backpropagation using a specific loss function (e.g., cross-entropy), to minimise the distinction between model predictions and the desired results.

2.2 Convolutional Neural Networks (CNN)

A convolutional neural network (CNN) takes an image patch (a group of pixels) as its input to predict high level feature representations (e.g., LU categories). The CNN network is basically cascaded by multiple convolutional, max-pooling, and batch normalisation layers to characterise the functional semantics at abstract and deep levels. Specifically, the convolutional layers involve a kernel function to convolve across input feature maps to recognise spatial features, followed by an activation function, such as Rectified Linear Unit, to strengthen and enhance the non-linearity. The max-pooling layers sub-sample the feature maps to enhance the generalisation capability with a reduced number of parameters (Romero *et al.*, 2016). The batch normalisation layers are used to accelerate the training process of the deep network by standardising the training sample batches (Li *et al.*, 2018). The parameters within the CNN network (e.g., kernel weights and biases) are learnt by a stochastic gradient descent in a feed-forward fashion (LeCun *et al.*, 2015). Finally, a fully connected layer is utilised together with a softmax classification to predict the final output.

2.3 Object-based Convolutional Neural Network (OCNN)

Object-based CNNs (OCNN) were designed on the basis of CNN models to classify segmented objects into specific LU classes (Zhang *et al.*, 2018c). Different from the standard pixel-wise

CNN that predicts image patches densely overlap at the pixel level, the OCNN places an image patch at the centroid of an object for prediction, which significantly enhances the computational efficiency while reducing the uncertainties caused by the convolutional process (e.g., geometric distortion). The image patch size is empirically tuned as sufficiently large to capture patterns of objects and their contexts. In Zhang *et al.* (2018c), the OCNN was trained to learn LU semantics through deep networks, and the boundaries of each object were maintained through image segmentation. The prediction of LU for each object was then assigned to the constituent pixels to formulate the final land use thematic map.

2.4 Scale sequence joint deep learning (SS-JDL)

The proposed scale sequence joint deep learning (SS-JDL) method has two major aspects: the creation and use of a scale sequence and joint learning between the LU and LC predictions at each scale in the scale hierarchy. The scale sequence is composed of a set of observational scales (image patch sizes) that transfers the information from a small scale to larger scales sequentially, in which fine details produced by convolution over a small window are integrated into a broader context through convolution over increasingly larger windows. Within each scale, the LU and LC are represented at different classification hierarchies and jointly classified through iteration. The general procedure of the proposed SS-JDL method is illustrated by Figure 1, where the LU and LC classifications are jointly derived across the scale sequence.

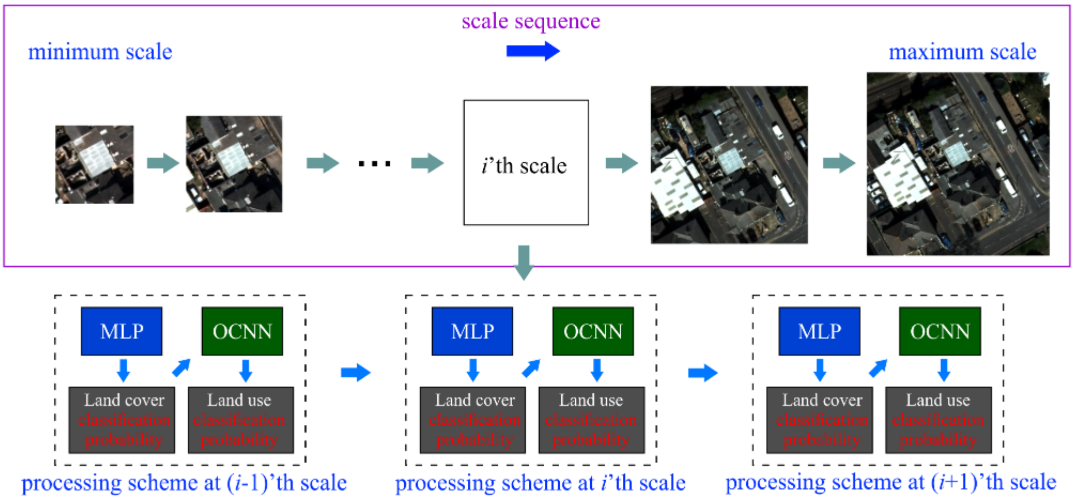
scale sequence
minimum scale
maximum scale
i'th scale
MLP
OCNN
Land cover classification probability
Land use classification probability
MLP
OCNN
Land cover classification probability
Land use classification probability
MLP
OCNN
Land cover classification probability
Land use classification probability
processing scheme at (i-1)'th scale
processing scheme at i'th scale
processing scheme at (i+1)'th scale

Figure 1. The general workflow of scale sequence joint deep learning (SS-JDL) for land cover and land use classification

In the SS-JDL method, a scale sequence (denoted as the set **S**) is needed to characterise the LU and LC across different scales. The **S** requires the parameterisation of the minimum scale ($\theta_{\min}$), the maximum scale ($\theta_{\max}$), and the total number of elements within **S** (n), in which the scale is derived by Eq. 1 as:

$$\mathbf{S} = \mathit{Linespace}(\theta_{\min},\ \theta_{\max},\ n) \tag{1}$$

Where, *Linespace* refers to the function of linear interpolation. By using Eq. 1, a scale sequence $\mathbf{S} = (s_1, s_2, \ldots, s_i, \ldots, s_n)$ is obtained, in which s_i ($i \subset [1, n]$) corresponds to the i-th scale value. Both $\theta_{\min}$ and $\theta_{\max}$ are computed based on the sizes of objects segmented from the imagery. The $\theta_{\min}$ is equal to or smaller than the minor axis of the smallest object, whereas the $\theta_{\max}$ is larger than the major axis of the largest object.

At each scale, the LU and LC classifications are derived from a pixel-based MLP classifier and a patch-based OCNN classifier, respectively (Zhang *et al.*, 2019). The LU classification probabilities are conditional on the LC classification probabilities, and the results of i-th iteration are influenced by the previous iteration. Such a hierarchical classification framework is formulated as a Markov process as:

$$P(\mathrm{LU}(\theta)^{i}, \mathrm{LC}^{i}) = P(\mathrm{LU}(\theta)^{i}, \mathrm{LC}^{i} \mid \mathrm{LU}(\theta)^{i-1}, \mathrm{LC}^{i-1}) \tag{2}$$

Where i denotes the number of iterations within the Markov process. The θ parameter provides the CNN input window size as the scale of the current iteration. The $\mathrm{LU}(\theta)^{i}$ in Eq. 2 refers to the LU classification probabilities at the i-th iteration. The $\mathrm{LC}^{\,i}$ corresponds to the land cover classification probabilities at the i-th iteration.

Given a scene of remotely sensed imagery M ($\mathbf{x}$, $\mathbf{y}$) with x and y representing the spatial coordinates, the training samples of LU and LC are described as $\mathbf{T}_{\mathrm{LC}} = (t_{\mathrm{LC}1}, t_{\mathrm{LC}2}, \ldots, t_{\mathrm{LC}i}, \ldots, t_{\mathrm{LC}u})$ and $\mathbf{T}_{\mathrm{LU}} = (t_{\mathrm{LU}1}, t_{\mathrm{LU}2}, \ldots, t_{\mathrm{LU}i}, \ldots, t_{\mathrm{LU}v})$, where u and v denote the total numbers of LU and LC training samples, respectively, and $t_{\mathrm{LC}i}$ and $t_{\mathrm{LU}i}$ refer to the i-th samples of LU and LC respectively. $t_{\mathrm{LC}i} = \{x_i, y_i, \mathcal{L}_{\mathrm{LC}}\}$ refers to the LC class label ($\mathcal{L}_{\mathrm{LC}}$) of the i-th sample and its spatial location (x_i, y_i) on imagery M, whereas $t_{\mathrm{LU}i} = \{x_i, y_i, \mathcal{L}_{\mathrm{LU}}\}$ denotes the LU class label ($\mathcal{L}_{\mathrm{LU}}$) and its position ($x_i$, y_i) in image M. The $\mathbf{T}_{\mathrm{LC}}$ and $\mathbf{T}_{\mathrm{LU}}$ were used to train the MLP and OCNN models to predict the LU and LC classification probabilities, respectively (Figure 1).

Based on Eq. 2, for the image M, the classification results of LU at previous iteration $\mathrm{LU}(\theta)^{i-1}$ (NULL for the first iteration), LC samples $\mathbf{T}_{\mathrm{LC}}$, LU samples $\mathbf{T}_{\mathrm{LU}}$, and the scale value of the current iteration θ serve as the input data and parameters. The probabilistic outputs of the LC ($M_{\mathrm{LCpro}}(i)$) and LU ($M_{\mathrm{LUpro}}(i)$) classifications are achieved through the iterative process. Detailed methods for achieving LU and LC classification probabilities and their output maps are demonstrated as follows:

(i) LC classification probabilities

LU classification probabilities at previous iteration $\mathrm{LU}(\theta)^{i-1}$ and the original image M are integrated as conditional probabilities for land cover classification ($M_{\mathrm{LC}}{}^{i}$) as:

$$M_{\mathrm{LC}}{}^{i} = Concate(M,\ \mathrm{LU}(\theta)^{i-1}) \qquad (3)$$

Where, *Concate* is a function to concatenate the image M with the LU classification probabilities at the previous iteration (i-1). Note, Eq. 3 corresponds to the case of i>1. If i=1, $M_{\mathrm{LC}}{}^{i}$ is equivalent to the original image M as the LU probabilities are empty (NULL) initially.

Based on Eq. 3, the MLP model is trained through the LC training samples ($\mathbf{T}_{\mathrm{LC}}$) as follows:

$$mlpmodel^{i} = \mathrm{MLP.Train}(M_{\mathrm{LC}}{}^{i},\ \mathbf{T}_{\mathrm{LC}}) \qquad (4)$$

The trained MLP model (*mlpmodel*^{i}) at the *i*-th iteration is used to predict the LC classification probabilities ($M_{\mathrm{LCpro}}{}^i$) as:

$$M_{\mathrm{LCpro}}{}^i = mlpmodel^i.\mathrm{Predict}(M_{\mathrm{LC}}{}^i) \tag{5}$$

Here, the extent of $M_{\mathrm{LCpro}}{}^i$ is equal to the size of image *M*, and the dimensions of $M_{\mathrm{LCpro}}{}^i$ are the same as the number of LC classes, with each dimension corresponding to the probabilities of a specific LC class predicted by the MLP classifier.

(ii) LU classification probabilities

LC classification probabilities derived from the MLP ($M_{\mathrm{LCpro}}{}^i$) are taken as the input image ($M_{\mathrm{LU}}{}^i$) for LU classification. The CNN model is trained by using $\mathbf{T}_{\mathrm{LU}}$ as:

$$cnnmodel^i = \mathrm{CNN.Train}(M_{\mathrm{LU}}{}^i,\ \mathbf{T}_{\mathrm{LU}}, \theta^i) \tag{6}$$

The *cnnmodel*^{i} model is further used to classify the image $M_{\mathrm{LU}}{}^i$ to link the LC probabilities with the LU classifications, and the LU classification probabilities ($M_{\mathrm{LUpro}}{}^i$) are obtained as follows:

$$M_{\mathrm{LUpro}}{}^i = cnnmodel^i.\mathrm{Predict}(M_{\mathrm{LU}}{}^i) \tag{7}$$

In Eq. 7, the object-based CNN is adopted for LU classification (Zhang *et al.*, 2018c), by which the prediction of the *cnnmodel*^{i} is assigned to the constituent pixels of the corresponding object. $M_{\mathrm{LUpro}}{}^i$ has the same image size as *M*, and the dimension is equal to the number of LU classes, with each dimension corresponding to the softmax probabilities acquired at the last layer of the CNN model.

Both land cover ($M_{\mathrm{LCpro}}{}^i$) and land use ($M_{\mathrm{LUpro}}{}^i$) probabilities are achieved in each iteration. The output at the final iteration (*n*) comprises $M_{\mathrm{LCpro}}{}^n$ and $M_{\mathrm{LUpro}}{}^n$, where the LU and LC thematic maps are acquired as:

$$M_{\mathrm{LCresult}} = \arg\max(M_{\mathrm{LCpro}}{}^n) \tag{8}$$

$$M_{\mathrm{LUresult}} = \arg\max(M_{\mathrm{LUpro}}{}^{n}) \tag{9}$$

In Eqs. 8 and 9, the probabilistic land cover ($M_{\mathrm{LCpro}}{}^{n}$) and land use ($M_{\mathrm{LUpro}}{}^{n}$) are converted into the corresponding LC (M_{LCresult}) and LU (M_{LUresult}) classes by outputting the maximum probabilities, respectively.

Essentially, the SS-JDL method inherits all the benefits of the JDL method (Zhang *et al.*, 2019) which are:

1. Joint classification of LU and LC in an automatic manner.
2. Increased classification accuracies for LU and LC through joint reinforcement.
3. Faithful representation of the hierarchical relationships between LU and LC characterisations.
4. Increased model robustness and generalisation capability with small sample size requirement for the CNN.

Combining scale sequencing with the JDL method brings three additional benefits:

1. Incorporation of a sequence of scales (patch sizes) within a single unified JDL framework.
2. Increased computational efficiency with rapid convergence to the optimal solution through simple and parsimonious scale sequence.
3. Autonomous implementation without the need to choose a specific or optimal scale of analysis.

3 Experiment and Results

3.1 Study area and data materials

Three study areas, including Bournemouth (S1), Southampton (S2) and Manchester (S3), and their surrounding terrestrial regions (Figure 2) were chosen in this research. Both S1 and S2 lie on the southern coast of England, whereas S3 is located inland in the north west of England. S1 represents a mixture of anthropogenic and semi-natural environments (e.g., Queen’s Park Golf Course, Heath). S2 is a major port influenced by human activities in both urban and rural settings (e.g., large-scale industry, agriculture), whereas S3 is a major inland city and metropolitan borough with a high-density of urban and suburban areas notable for its commercial and social impact. They are, therefore, highly distinctive and heterogeneous in both LU and LC configurations and are, thereby, able to be used to test the generalisation ability of the proposed method.

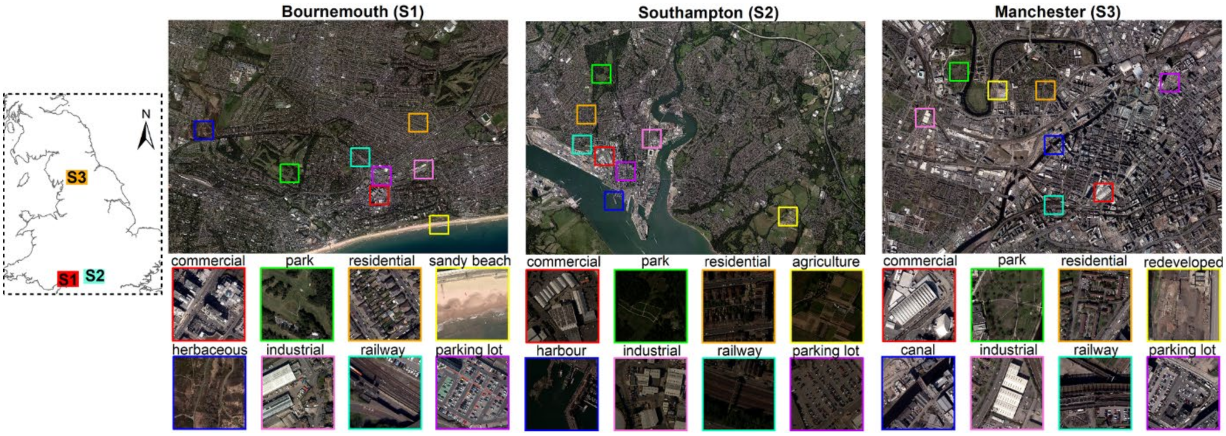

Figure 2. Three study areas: Bournemouth (S1), Southampton (S2) and Manchester (S3) in England, with typical land use categories highlighted for each study site.

Aerial photos of S1 (23,070×18,526 pixels), S2 (23,250×17,500 pixels) and S3 (17,590×14,360 pixels) composed of four spectral bands (R, G, B and NIR) with 50-cm spatial resolution, were captured by Vexcel UltraCam Xp digital aerial cameras on 20 April 2016, 22 July 2012, and 20 April 2016, respectively. Ten, nine and nine LC categories were recognised in S1, S2, and S3, respectively (Table 1). Eight LC classes appear consistently at three study sites: *Concrete Roof*, *Clay Roof*, *Metal Roof*, *Asphalt*, *Bare Soil*, *Rail*, *Grassland,* and *Woodland*. The remaining two LC classes in S1 were *Heath* and *Sand*, the one in S2 was *Crops*, and the one in S3 was *Water*. Those LCs characterise the physical characteristics of the ground surface, whereas the LUs represent functional use induced by human beings. Eleven LU types, including *Commercial*, *Industrial*, *Residential*, *Institutional*, *Highway*, *Railway*, *Parking Lot*, *Park and Recreational Area*, *Redeveloped Area*, *Herbaceous Vegetation*, and *Sandy Beach*, were identified in S1. As for S2, 10 major types of LUs were involved, namely, *Commercial*, *Industrial*, *Medium-density Residential*, *High-density Residential*, *Railway*, *Highway*, *Parking Lot*, *Redeveloped Area*, *Park and Recreational Area,* and *Agricultural Area*. In terms of S3, nine main LU categories were found, including: *Commercial*, *Industrial*, *Residential*, *Railway*, *Highway*, *Parking Lot*, *Redeveloped Area*, *Park and Recreational Area,* and *Canal*. These LU and LC classes were defined based on the European Environment Agency Urban Atlas 2012 and the Land Cover Map 2015 produced by NERC Centre for Ecology & Hydrology, together with the UK national land use system developed by Ministry of Housing, Communities and Local Government. Detailed LU classes and their sub-classes as well as major LC components were listed in Table 1.

Table 1. The land use (LU) classes with their sub-class descriptions, and the associated major land cover (LC) components across the three study sites (S1, S2 and S3).

LU	Study site	Sub-class descriptions	Major LC
(High-density) residential	S1, S2, S3	Residential houses, terraces, green space	Buildings, Grassland, Woodland
(Medium-density) residential	S2	Residential flats, green space, parking lots	Buildings, Grassland, Asphalt
Commercial	S1, S2, S3	Shopping centre, retail parks, commercial services	Buildings, Asphalt
Industrial	S1, S2, S3	Marine transportation, car factories, gas industry	Buildings, Asphalt
Highway	S1, S2, S3	Asphalt road, lane, cars	Asphalt
Railway	S1, S2, S3	Rail tracks, gravel, sometimes covered by trains	Rail, Bare soil, Woodland
Parking lot	S1, S2, S3	Asphalt road, parking line, cars	Asphalt
Park and recreational area	S1, S2, S3	Green space and vegetation, bare soil, lake	Grassland, Woodland
Redeveloped area	S1, S2, S3	Bare soil, scattered vegetation, reconstructions	Bare soil, Grassland
Sandy beach	S1	Costal line, sand, seaside beaches	Asphalt, Bare soil
Herbaceous Vegetation	S1	Grasses and Forbs, shrubs	Grassland, Woodland
Agricultural area	S2	Pastures, arable land, and permanent crops	Crops, Grassland
Canal	S3	Water drainage channels, canal water	Water, Asphalt

Reference polygons for LU and LC are collected by field surveyors and manually digitised by photogrammetrists at Ordnance Survey (Britain's National Mapping Agency). These reference polygons (covering the majority of study sites) were split randomly into 60% for training and 40% for validation. Sample points were chosen by means of stratified random sampling within the training and testing polygons, and the numbers of each LU and LC class were made proportional to the area of the total reference polygons for each class. For classes that were sparsely covered (e.g., railway), their sample sizes were enlarged to achieve a representative distribution. Approximately, 600 and 1000 samples per class for both LU and LC were adopted, allowing the MLP and the CNN networks to be sufficiently trained with a relatively large sample size. These sample points were cross-validated by the Ordnance Survey MasterMap Topographic Layer, Open Street Maps, and the CEH Land Cover® plus: Crops (https://www.ceh.ac.uk/crops2015) to ensure precision and the fidelity of the selected samples.

3.2 Experimental design and parameters

Within the SS-JDL method, the MLP and OCNN classifiers need to predefine parameters to obtain the highest classification accuracy and generalisation for both study sites. These models were parameterised in S1 and directly applied to S2, as recommended by Zhang *et al.* (2018c) and Zhang *et al.* (2019). The structures of the model and parameters are detailed below.

For MLP, the initial input is four-band image at the pixel level, and the initial prediction of each pixel corresponds to the LC category. Two hidden layers were chosen as optimal with 20 nodes in each layer. The activation functions for the hidden layers were set as 'Rectified Linear Unit' to achieve nonlinearity within the MLP network, and the number of epochs was tuned to 1000 to allow full convergence to a stable state through backpropagation.

The OCNN requires pre-processing of the image into homogeneous objects that are representative of specific LCs through object-based image segmentation. Multi-resolution segmentation was implemented using the eCognition 9.2 software to acquire the segmented objects. The scale parameter was varied from 10 to 100 to explore the influence of object size on segmentation performance, and 40 was found to be the optimal parameter to obtain slightly over-segmented results.

For each object, a standard CNN was applied to an image patch located at the object centre to learn the within-object information and its spatial context. Nine hidden layers that alternate with convolution, max-pooling, and batch normalisation, were designed to capture the deep LU feature representations (Figure 3). Small filters (3×3) in convolutional layers were adopted following the common deep network structures (e.g., VGG-16), and the number of filters was tuned as 64 to extract the multi-dimensional deep feature representations. The learning rate and the epoch were set as 0.01 and 800, respectively, to learn the deep features through iteration.

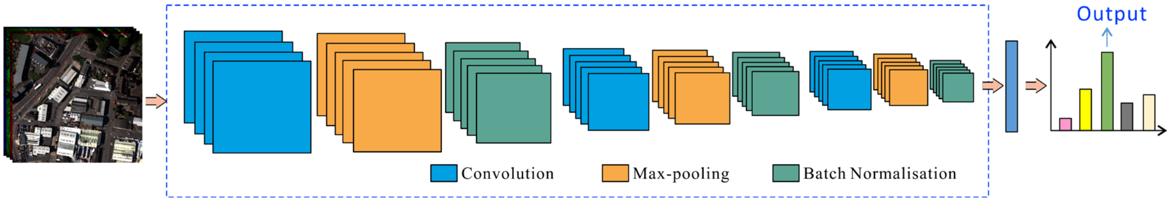

Figure 3. Model structures and architectures of the deep CNN network with nine hidden layers.

3.3 Benchmarks and parameter settings

In this research, five typical methods served as benchmarks for LC classification, including the MLP (spectral only), GLCM-MLP (spectral and textural features), Markov random field (MRF, contextual-based), Multi-scale CNN applied to land cover (MCNN-LC), and the recently proposed Joint Deep Learning method applied to land cover (JDL-LC; as for SS-JDL but without scale sequencing). As for LU classification, five state-of-the-art approaches were benchmarked, including MRF, object-based image analysis (OBIA), the standard pixel-wise CNN, Multi-scale CNN applied to land use (MCNN-LU), and Joint Deep Learning applied to land use (JDL-LU). The classification experiments were implemented using Keras/Tensorflow under a Python environment using a laptop with a NVIDIA 940M GPU and 12.0 GB memory. The parameters of these benchmark comparators are detailed below.

MLP took pixel-based four spectral bands as input, with two hidden layers inside the network and 20 nodes for each of them as parameterised by Zhang *et al.* (2018a). The output was the LC label for each pixel.

GLCM-MLP used the same structure as the MLP, while grey-level co-occurrence matrix (GLCM) texture variables were added as additional input features. The prediction was the LC class label at the pixel level.

MRF took the support vector machine as its basic spectral classifier for both LU and LC classification, in which the Radial Basis Function was adopted as the kernel function. Following the recommendations of Zhang *et al.* (2018b), the window size of the MRF was tuned as 5×5, and the smoothing parameter was set as 0.7 to achieve smoothed results using contextual information.

MCNN was designed for both land cover (MCNN-LC) and land use (MCNN-LU) classification based on majority voting at three input scales (CNN window sizes) as proposed by Lv *et al.* (2018). Following the recommendation of Lv *et al.* (2018), three CNN window sizes at 15×15, 25×25, and 35×35 were used as the input patch sizes to classify regions produced by multi-resolution segmentation with a scale parameter of 20. The predictions of the triple-scale CNNs were fused through majority voting to obtain LC and LU classification results, respectively.

JDL-LC incorporated an MLP and OCNN to learn iteratively the LU and LC classification probabilities, respectively. The number of iterations was set to 15 to allow full convergence to a stable state. The prediction of the MLP at the final iteration was taken as the JDL-LC classification result (Zhang *et al.*, 2019).

OBIA was implemented on objects derived from multi-resolution segmentation. Various features were then extracted from the objects, including spectral features (mean and standard deviation), GLCM texture variables and geometry. An SVM was used for object-based classification using these hand-coded features.

CNN was a trained deep network to predict pixel-wise densely overlapping patches across entire image. The input patch size was parameterised as 48×48 as recommended by Längkvist *et al.* (2016), and the number of layers was set as six (alternating between convolution and max-pooling). Softmax regression was adopted to predict the final LU classification results.

JDL-LU was performed by a pixel-based MLP to predict LC probabilities which were used as input features for LU prediction using an object-based CNN. This system can jointly learn the LU and LC classes through iteration. The JDL-LU classification result was achieved at 15 iterations with a steady state (Zhang *et al.*, 2019).

3.4 Classification Results and Analysis

3.4.1 Results and analysis of the scale sequence

The minimum scale for the SS-JDL was set as 28×28 to capture the within-object information, given that the main axis of the smallest object size was found to be less than 14 metres in S1, S2 and S3. The maximum scale was parameterised as 140×140 by considering the largest object within the three scenes to cover the wider spatial context while leveraging the representation capability of the CNN network. Between the minimum and maximum scales, a range of scales were interpolated into the network to obtain a sequence of scales (i.e., CNN window sizes). The smallest number of iterations for the SS-JDL was two representing the minimum and maximum scales only. The number of iterations increases as more scales are introduced. Figure 4 demonstrates the influence of the number of iterations on the overall accuracy, and the SS-JDL method is compared with the recently proposed JDL method on both the S1 and S2 images through iteration. The SS-JDL method consistently shows rapid convergence, with the optimal accuracy achieved in just 5 iterations (red dashed line), significantly faster than the JDL method for both LU and LC classification at 10 iterations (green dashed line). Specifically, for S1, the SS-JDL accuracy started at around 82% and 79% for the LC and LU classifications at iteration 2, and rapidly increased to approximately 91% (LC) and 88.5% (LU) at iteration 5. In contrast, the JDL accuracy was slightly higher than that of the SS-JDL at iteration 2, with around 82.5% (LC) and 80% (LU), and increased slowly towards the optimum accuracy of ~90% (LC) and ~87% (LU) at iteration 10.

A similar trend was found in S2 and S3 (Figure 4), where the SS-JDL accuracy began at around 80% for LC and 79% for LU, and reached 90% (LC) and 88% (LU) at iteration 5. The accuracy of the JDL-LC classifier was slightly higher at iteration 2 (81%), and gradually increased to around 89% at iteration 10, which is still lower than that of the LC classification of SS-JDL (90%). The accuracy of the JDL-LU, in contrast, started lower than that of the SS-JDL, at around 78.5% at iteration 2, and slowly increased with iteration. The optimal accuracy was found at iteration 10 with around 86% accuracy (2% lower than for the LU classification of SS-JDL).

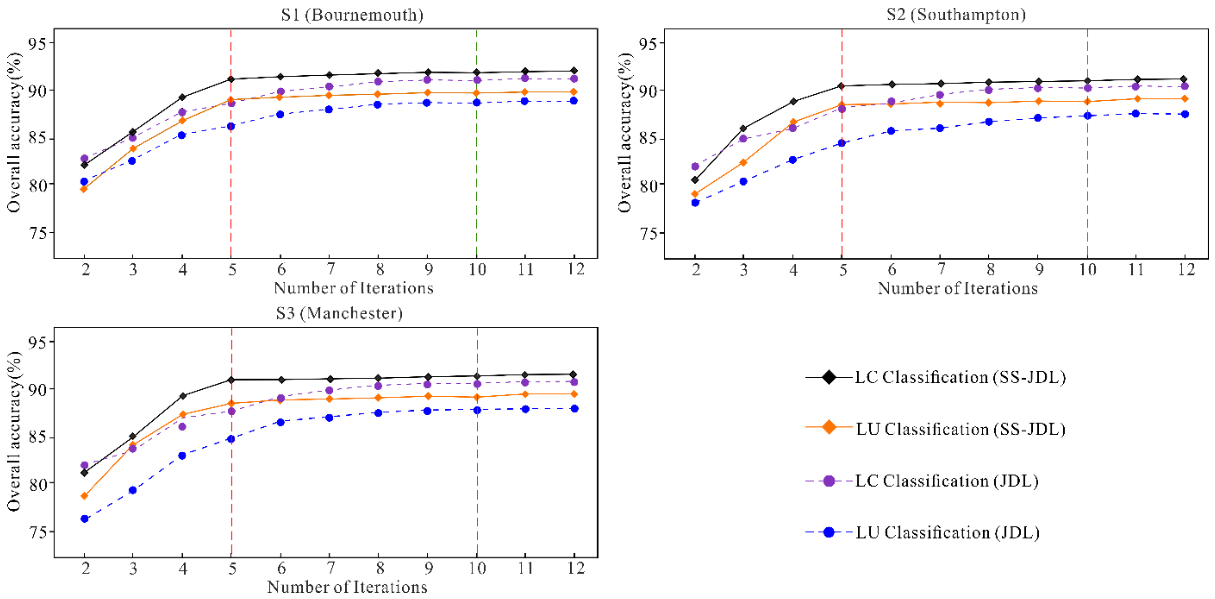

Figure 4. The influence of iteration upon overall accuracy for the LU and LC classifications using the proposed SS-JDL and the JDL method.

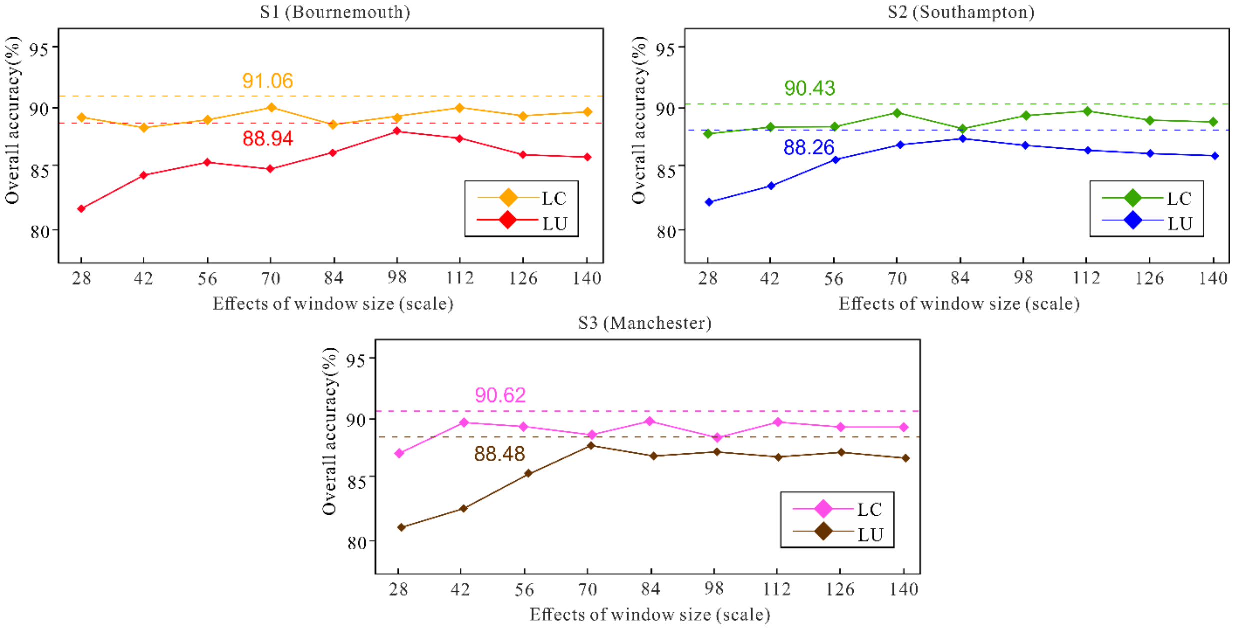

Figure 5. The effects of window size (scale) on overall accuracy of the LU and LC classifications using the SS-JDL (dashed lines) and the JDL method (solid lines).

The SS-JDL involves multiple scales across the scale sequence and, thus, does not require optimal scale selection. Figure 5 shows the scale selection processes for JDL in comparison to the SS-JDL method with 5 iterations (scales). A range of CNN window sizes were considered, including 28×28, 42×42, 56×56, 70×70, 84×84, 98×98, 112×112, 126×126, and 140×140, and the classifier at each window size was run 20 times to achieve the converged LU and LC classification results. As shown in Figure 5, the SS-JDL method (dashed lines) always outperforms the JDL (solid lines) for both LC classification (OA of 91.06%, 90.43% and 90.62%) and LU classification (OA of 88.94%, 88.26% and 88.48%) for S1, S2 and S3, respectively. For JDL, both LU and LC classifications demonstrate variation along the changing window size, and it is hard to judge the optimal scale. In S1, 28×28, 70×70 and 112×112 are potentially the "optimal" LC window size, whereas the optimal scale for LU classification might be 98×98. Likewise, for S2 multiple accuracy peaks are produced for LC (70×70, 112×112, 140×140), while a single optimum scale (84×84) is found for LU. Similar trends are found in S3, with three accuracy peaks for LC (42×42, 84×84, 112×112) and one optimum scale (70×70) for LU. Clearly, the LU classification is much more sensitive to scale effects with larger accuracy differences (around 81% to 88%), whereas the LC classification does not have as clear a correlation to the CNN window size. In addition, the “optimal” scales for LU and LC are completely different. For example, the optimal scale for LU in S1 is found at 98×98, but this does not coincide with the optimal scales for LC (28×28, 70×70 and 112×112). The SS-JDL, therefore, demonstrates greater classification accuracy for all study sites (S1, S2 and S3) without requiring an optimal scale selection process.

In this paper, a forward scale sequence (FSS) derived by the minimum and the maximum sizes of the segmented objects in the imagery was adopted for JDL classification. The potential sampling space for the scale sequences, however, is enormous (from completely random to sequential scales), and it is extremely hard to examine exhaustively the entire set of possible scale choices. To better explore the space, four typical sampling schemes were considered, including the forward scale sequence (FSS) from small to large scale, the backward scale sequence (BSS) from large to small scale, the random scale sequence (RSS) with scales in a completely random order generated by a Monte Carlo method, as well as the iterative greedy scale sequence (IGSS) that chooses the scale with the best accuracy increase at each iteration. Table 2 demonstrates the superiority of FSS in OA and computational efficiency compared with IGSS, RSS, and BSS. The high OA is achieved by gradually enlarging the observational scales from the minimum to the maximum, while retaining the precise information achieved initially at the smaller scales through subsequent scales. In the meantime, exhaustive search (e.g., IGSS) was not required by the FSS, thereby significantly reducing the computational time through fast implementation.

Table 2. The overall accuracy and the computational time of four sampling schemes, including forward scale sequence (FSS), backward scale sequence (BSS), random scale sequence (RSS), and iterative greedy scale sequence (IGSS).

Sampling scheme	Overall Accuracy (%)			Computational time (h)
	S1 (LC, LU)	S2 (LC, LU)	S3 (LC, LU)	S1, S2, S3
FSS	91.06, 88.94	90.43, 88.26	90.62, 88.48	7.52, 7.86, 7.32
BSS	86.73, 83.84	86.68, 83.05	87.04, 84.26	7.52, 7.86, 7.64
RSS	87.24, 84.32	87.59, 84.13	87.74, 83.85	8.95, 9.37, 9.28
IGSS	90.35, 87.69	89.76, 87.14	89.43, 87.25	35.58, 37.94, 36.65

3.4.2 Classification results and analysis for all study sites

To gain a better spatial visualisation of how the classification accuracy increases with iteration, the converged five iterations of the SS-JDL for both LC (Figure 6, 7 and 8) and LU (Figure 9, 10 and 11) are demonstrated at iteration 1 (28 × 28) to iteration 5 (140 × 140) using three subsets of S1 and S2 as well as one subset of S3, respectively (Figures 6 to 11).

The LC classification result at iteration 1 (28 × 28) contained severe salt-and-pepper effects, as shown in Figure 6 (a, b and c), Figure 7 (a, b and c), and Figure 8(b). Such problems were tackled through iteration by incorporating spatial context from the LU probabilities and increasing the scale at each iteration. Iteration 2 significantly smoothed the classification results while keeping the fidelity in the representations, thereby enhancing the classification accuracy, accordingly. Figure 6(b) illustrates the clear increase in accuracy achieved by reducing the noise (salt-and-pepper effects) in the Asphalt road and the Rail classes as well as the Concrete roof class. Both iterations 1 and 2, however, failed to differentiate Concrete roof and Asphalt (e.g., the red circles in Figure 6(a) and 6(c) as well as Figure 7(b)), given the extremely similar spectral reflectance between them. Those pixels misclassified as Concrete roof were rectified to Asphalt after iteration 3 and remained the same throughout further iterations (e.g. Figure 8(c)). Another remarkable improvement demonstrated through iteration was the elimination of Bare soil within the classification maps. For example, the falsely classified Bare soil pixels at iterations 1 to 4 of Figure 6(a) and iterations 1 to 3 of Figure 6(c) were corrected as Asphalt and Sand, respectively.

More impressively, the shadow effects cast by the woodland and buildings shown in Figure 7(b) and Figure 8(a) were falsely classified as Rail and Concrete roof at iterations 1 and 2, but were gradually rectified to Asphalt or partial Woodland at iterations 3 and 4, and the shadow adjacent to the trees was completely replaced as entirely Woodland at iteration 5. In terms of agricultural land, the Crop and Grassland classes were more clearly differentiated through further iteration. Figure 7(c) demonstrates the misclassified Grassland at iterations 1, 2 and 3, which was partially rectified to Crops at iteration 4, and completely identified as Crops with high accuracy at iteration 5.

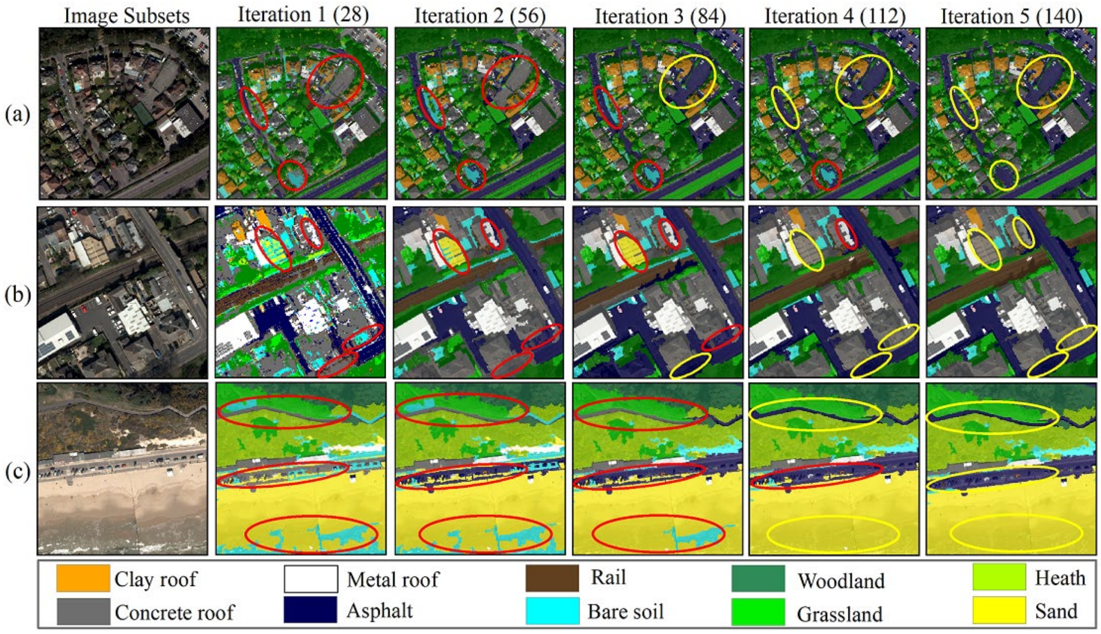

Figure 6. Three subset (i.e., a, b, c) of LC classification in S1 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

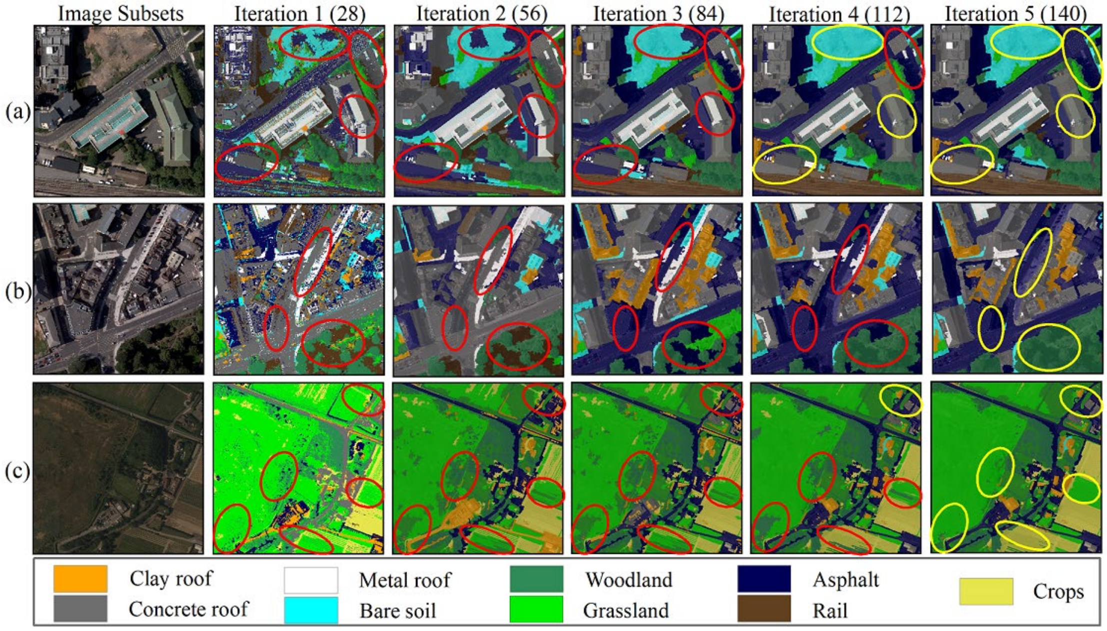

Figure 7. Three subset (i.e., a, b, c) of LC classification in S2 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

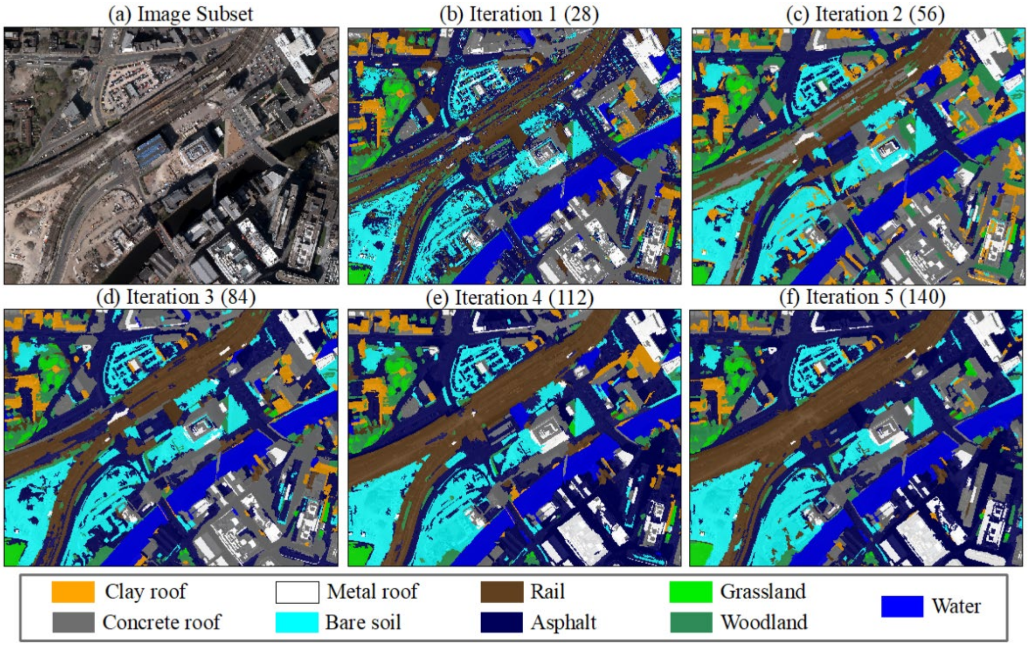

Figure 8. The land cover classification in S3 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140).

In terms of LU classification, the most significant increase in accuracy was obtained for the Parking lot class, which was correctly differentiated after iteration. For example, the confusion between Parking lot and Highway is shown in Figure 9(b) at iterations 1 to 4 and Figure 10(b) at iterations 1 to 3 (red circles) and Figure 11(b) and 11(c), which was resolved and clearly identified as Highway at iteration 5 (yellow circles). Those pixels misclassified as Commercial at iterations 1 to 3 (Figure 10(a)) were correctly modified to Parking lot at iterations 4 and 5. Furthermore, the misclassification between Highway and Railway was rectified throughout the iterative process. For example, Figure 9(b) and 11(d) show that some Railways were affected by shadows and wrongly identified as Highway at iterations 1 to 3. Likewise, some of the Highways in Figure 9(c) were falsely classified as Railway at iterations 1 to 4 when adjacent to sandy beaches. These problems were addressed and differentiated accurately at iteration 5 in all cases. Moreover, the mutual confusion between Agricultural area and Redeveloped area is shown in Figure 10(c) with red circles, which was precisely distinguished with sharp boundaries at the 5th iteration (in yellow circles).

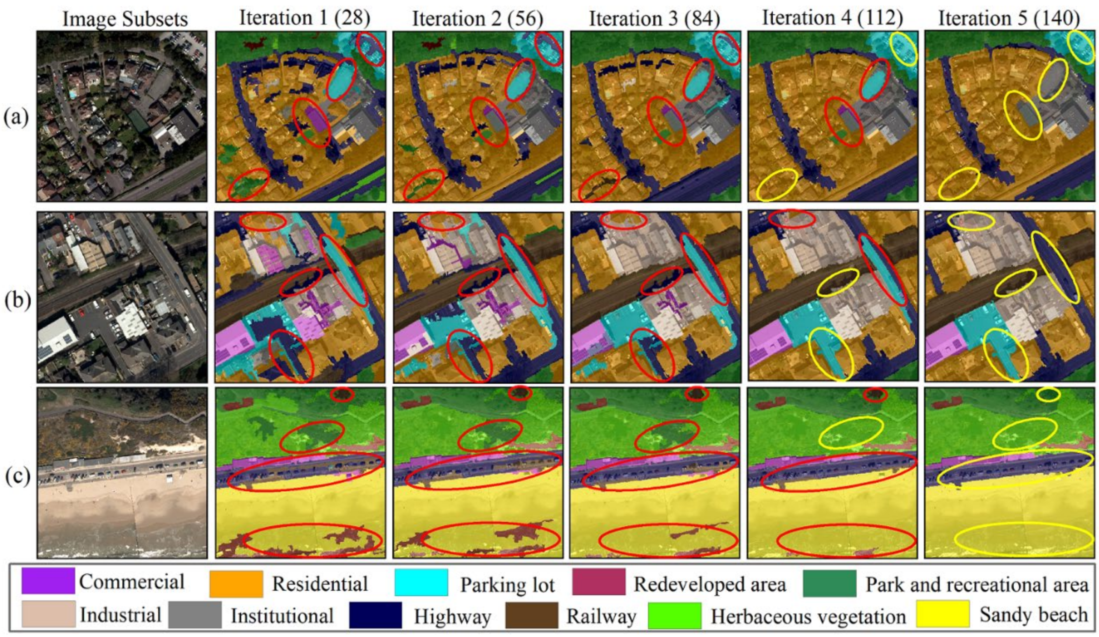

Figure 9. Three subset (i.e., a, b, c) of LU classification in S1 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

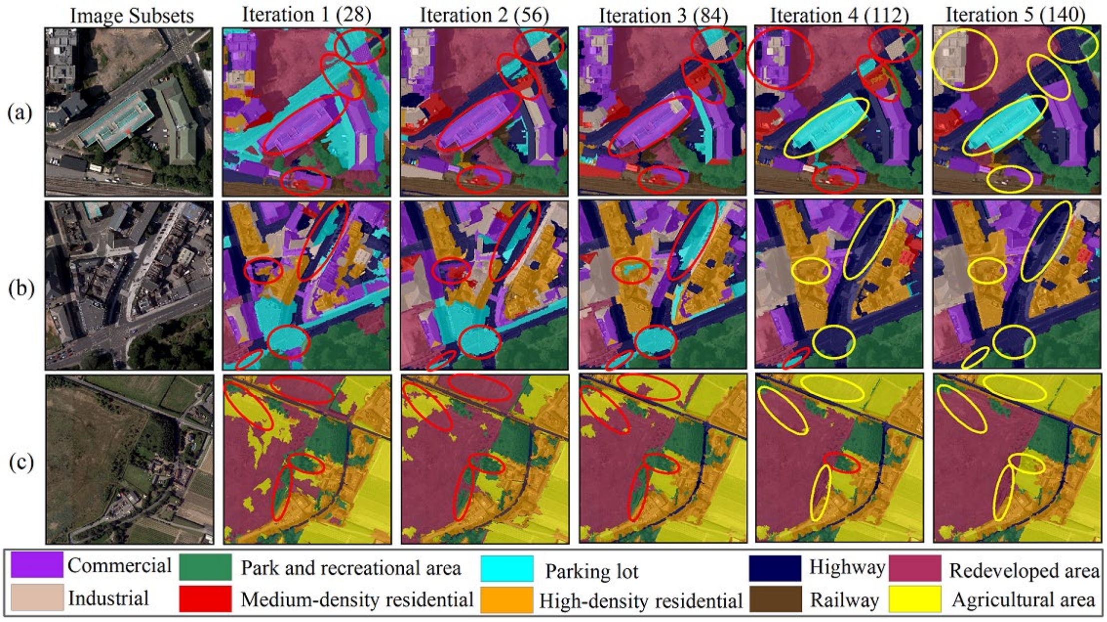

Figure 10. Three subset (i.e., a, b, c) of LU classification in S2 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

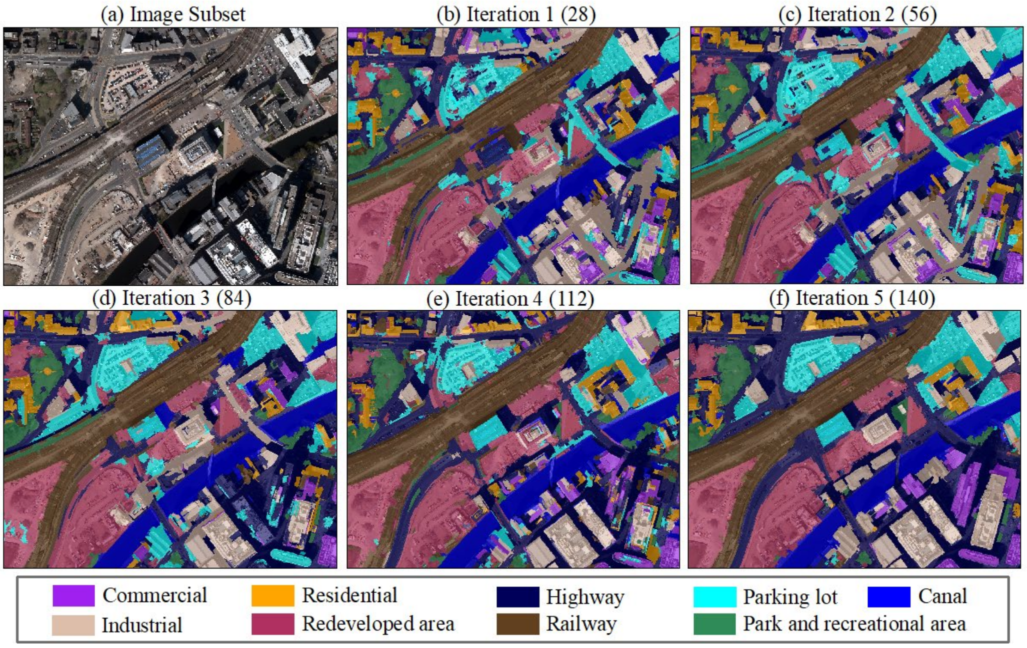

Figure 11. The land use classification in S3 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140).

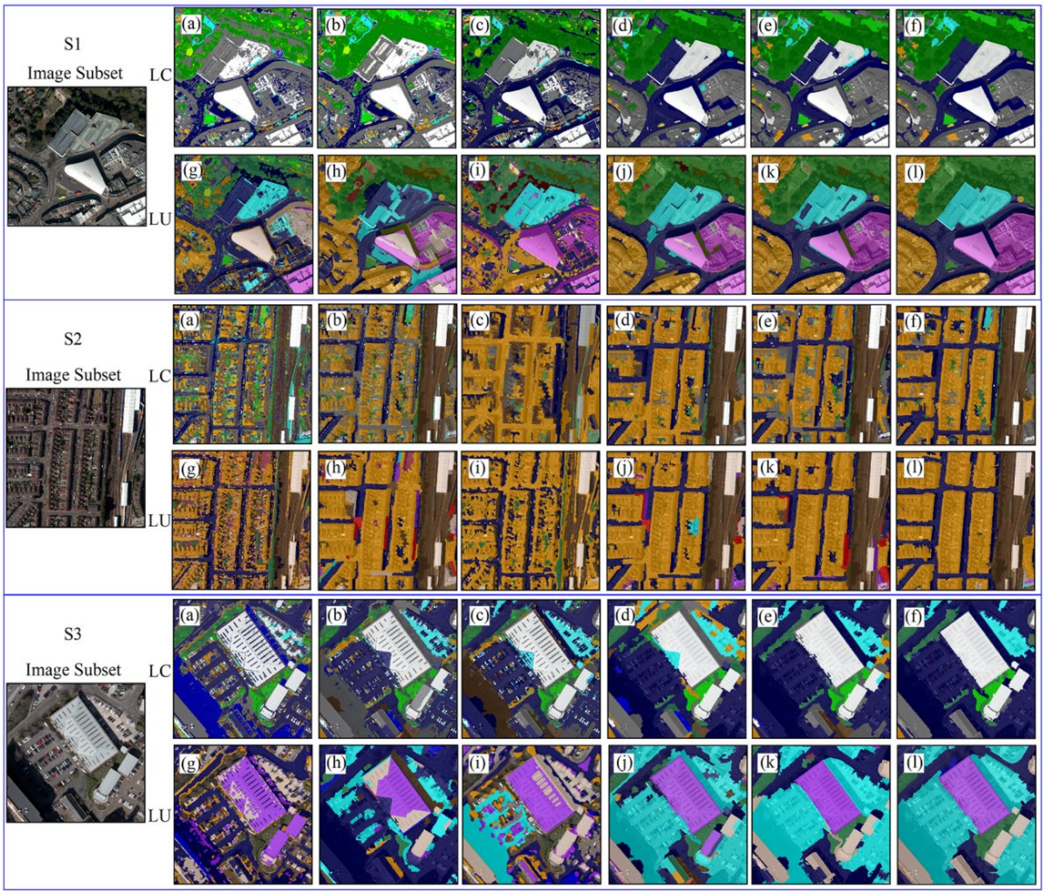

Figure 12. Image subset benchmark comparison among various methods for S1, S2 and S3. The LC classifications include (a) MLP, (b) GLCM-MLP, (c) MRF, (d) MCNN-LC, (e) JDL-LC, and the (f) SS-JDL-LC. The LU classifications include (g) MRF, (h) OBIA, (i) CNN, (j) MCNN -LU, (k) JDL-LU, and the (l) SS-JDL-LU. Refer to Figures 6 to 11 for details of the corresponding classification legends.

The classification accuracy of the proposed SS-JDL was further compared with a range of benchmark approaches for S1, S2 and S3, respectively. The LC results (SS-JDL-LC) were benchmarked with other comparators, including the MLP, the GLCM-MLP, the MRF, the MCNN-LC, and the JDL-LC; whereas, the LU results (SS-JDL-LU), were compared with MRF, OBIA, CNN, MCNN-LU, and the JDL-LU. Visual inspections and accuracy assessment, based on the overall accuracy (OA), Kappa coefficient (κ) and the per-class mapping accuracy, were used to test the classification results.

Figure 12 demonstrates visually the classification results of S1, S2 and S3 amongst the various benchmark methods. For LC, the pixel-based MLP showed the lowest classification accuracy due to the severe salt-and-pepper effects in all study sites (Figure 12(a)). Confusion was found between the Asphalt and Concrete roof classes together with the severe issues of shadow cast by buildings and woodlands. The GLCM-MLP incorporated spatial texture within the image, which increased the capability to capture ground objects with distinctive textures. For example, the woodlands with course textures were identified accurately in Figure 12(b). Such GLCM-MLP based classification results, however, still suffered from difficulties in differentiating those LC classes with similar spectra and textures (e.g., the Asphalt and Concrete roof classes). The MRF significantly increased the ability to characterise the Asphalt road class by borrowing adjacent neighbourhood information, but suffered from some issues with respect to other classes (e.g., Concrete roof and Clay roof) as illustrated in Figure 12(c). The MCNN-LC clearly showed increased accuracy in differentiating Asphalt and Concrete roof, but some edges along the roads and bare soils were misclassified as Clay roof (Figure 12(d)). The JDL-LC significantly increased the classification accuracy using LU and LC characteristics iteratively (Figure 12(e)). It, however, failed to resolve some problems along the object boundaries (e.g., for the Asphalt class). The proposed SS-JDL-LC method solved all these problems achieving a high accuracy overall (Figure 12(f)).

In terms of LU classification, the MRF demonstrated serious deficiencies in identifying residential and commercial areas with noisy results (Figure 12(g)). OBIA smoothed the LU classification to a large extent, but failed to differentiate complex objects such as the Parking lot and Industrial classes, and lost some fine-grained details (e.g., Highway) (Figure 12(h)). The pixel-wise CNN showed some advantages in capturing complex LU classes (e.g., Parking lot, Commercial). It, however, produced severe geometric distortions (e.g., the enlarged commercial buildings) and poorly defined boundaries (e.g., the edge between the Residential and Highway classes) (Figure 12(i)). The MCNN-LU achieved increased accuracy in classifying the Parking lot class, but failed to capture continuous linear features such as Highway or Railway (Figure

12(j)). JDL-LU (Figure 12(k)) and the proposed SS-JDL-LU (Figure 12(l)) share similar classification results with high precision and fidelity. The SS-JDL-LU, surprisingly, demonstrated some further improvements in identifying detailed objects and their boundaries (e.g., Highway).

The quantitative accuracy assessment for LC classification is reported in Tables 3, 4 and 5 for S1, S2 and S3, respectively. The SS-JDL-LC consistently achieved the highest OA of 91.06%, 90.43% and 90.62% (κ = 0.90, 0.89 and 0.89) for S1, S2 and S3, respectively, greater than the JDL-LC of 89.68%, 88.29% and 88.48% (κ = 0.88, 0.87 and 0.87), the MCNN-LC of 87.54%, 86.95% and 86.57% (κ = 0.86, 0.86 and 0.85), the MRF of 84.32%, 84.78% and 84.54% (κ = 0.84, 0.84 and 0.83), the GLCM-MLP of 83.24%, 82.85% and 83.06% (κ = 0.82, 0.82 and 0.82), and the MLP of 82.06%, 81.29% and 82.22% (κ = 0.81, 0.80 and 0.81), respectively. In terms of LU classification, the SS-JDL-LU yielded the greatest OA (88.94%, 88.26% and 88.48%) for S1, S2 and S3 with the highest κ (0.89, 0.88 and 0.88), consistently higher than the JDL-LU (OA = 87.68%, 87.58% and 86.26%, κ = 0.88, 0.87 and 0.86), the MCNN-LU (OA = 85.94%, 85.29% and 85.08%, κ = 0.86, 0.85 and 0.84), the CNN (84.32%, 84.08% and 83.32%, κ = 0.84, 0.83 and 0.82), the OBIA of (82.17%, 80.26% and 80.42%, κ = 0.82, 0.80 and 0.80), and the MRF (81.06%, 79.38% and 79.29%, κ = 0.80, 0.79 and 0.79).

The per-class mapping accuracy further demonstrated the superiority of the SS-JDL method, with the most accurate results shown in bold font in Tables 3 to 8. Specifically, for LC classification, the Clay roof, Metal roof, Woodland, Grassland, Asphalt classes were accurately classified in S1, S2 and S3 using the SS-JDL-LC (accuracy > 90%) by incorporating spatial and spectral feature representations across different scales. Such high accuracies were also achieved for Heath (90.07%) and Sand (92.62%) in S1, Crops (90.74%) in S2 and Water (98.27%) in S3. The accuracies of these LC classes, in particular Woodland and Grassland (90.99% and 91.62% on average), were significantly higher than for the benchmarks, with average accuracies for the

MLP (69.36% and 71.74%), GLCM-MLP (72.78% and 71.63%), MRF (76.15% and 75.47%), MCNN-LC (85.95% and 86.05%), and JDL-LC (88.75% and 90.35%), respectively. The Concrete roof class was the most challenging LC class to be classified, producing the lowest accuracy of 83.07% on average for the SS-JDL-LC, which was nevertheless significantly higher than for the MLP (70.19%), GLCM-MLP (72.62%), MRF (73.89%), MCNN-LC (77.56%), and JDL-LC (79.56%), respectively. Accuracies for other classes, such as Rail and Bare soil (88.57% and 87.16%) were less significantly increased using the SS-JDL-LC compared with the benchmark methods, in which less than 5% accuracy differences were found among them.

Tables 6, 7 and 8 show the quantitative accuracy assessment for LU classification for S1, S2 and S3, respectively. The greatest accuracy increases were shown for the Commercial, Industrial, Parking lot and Highway classes, with average accuracies of 85.10%, 85.58%, 91.71%, and 84.74%, respectively, for the proposed SS-JDL-LU, much higher than for the MRF (70.75%, 70.78%, 79.12%, 78.37%), OBIA (71.24%, 71.06%, 81.42%, 78.85%), CNN (73.85%, 73.64%, 84.16%, 79.10%), MCNN-LU (78.19%, 80.14%, 86.20%, 80.44%), and JDL-LU (82.61%, 83.74%, 88.08%, 81.57%). For the Residential, Redeveloped area, and Park and recreational area classes, moderately increased accuracies were obtained by the SS-JDL-LU (88.49%, 91.59%, and 95.02%), greater than for the MRF (80.27%, 81.74%, 88.29%), OBIA (81.39%, 83.79%, 90.06%), CNN (82.06%, 86.79%, 91.29%), MCNN-LU (83.86%, 88.24%, 91.42%), and JDL-LU (86.63%, 89.70%, 93.69%), respectively. Other LU classes, including Railway, Herbaceous vegetation, Sandy beach, Canal, and Agricultural area, did not show significant increases in accuracy in comparison with the benchmarks, with similar accuracies being achieved by the JDL-LU and SS-JDL-LU classifiers.

Table 3. LC accuracy comparison for each class and overall between MLP, GLCM-MLP, MRF, MCNN-LC, JDL-LC, and the proposed SS-JDL-LC method in S1. The largest classification accuracies and Kappa coefficients are shown in bold font.

LC Class (S1)	MLP	GLCM-MLP	MRF	MCNN-LC	JDL-LC	SS-JDL-LC
Clay roof	90.12%	88.62%	89.58%	88.27%	91.87%	**92.16%**
Concrete roof	70.54%	73.95%	74.23%	77.59%	80.25%	**84.05%**
Metal roof	90.17%	90.28%	90.16%	90.82%	91.34%	**91.64%**
Woodland	69.45%	73.02%	76.28%	85.43%	88.24%	**90.82%**
Grassland	72.36%	72.94%	75.53%	86.32%	90.65%	**92.43%**
Asphalt	89.42%	88.57%	89.42%	88.29%	90.22%	**90.68%**
Rail	83.21%	83.26%	83.56%	86.37%	88.54%	**88.95%**
Bare soil	80.23%	81.05%	82.44%	83.52%	85.59%	**86.78%**
Heath	82.63%	83.84%	86.18%	87.24%	89.74%	**90.07%**
Sand	88.39%	88.98%	89.54%	89.43%	91.42%	**92.62%**
Overall Accuracy (OA)	82.06%	83.24%	84.32%	87.54%	89.68%	**91.06%**
Kappa Coefficient (*κ*)	0.81	0.82	0.84	0.86	0.88	**0.90**

Table 4. LC accuracy comparison for each class and overall between MLP, GLCM-MLP, MRF, MCNN-LC, JDL-LC, and the proposed SS-JDL-LC method in S2. The largest classification accuracies and Kappa coefficients are shown in bold font.

LC Class (S2)	MLP	GLCM-MLP	MRF	MCNN-LC	JDL-LC	SS-JDL-LC
Clay roof	89.57%	88.27%	89.17%	90.05%	91.36%	**91.92%**
Concrete roof	69.45%	71.82%	73.24%	77.56%	79.48%	**82.43%**
Metal roof	89.36%	89.43%	90.18%	90.74%	91.56%	**91.86%**
Woodland	69.03%	72.18%	76.84%	86.39%	88.54%	**90.74%**
Grassland	70.64%	71.36%	75.42%	84.28%	90.06%	**91.87%**
Asphalt	88.42%	88.75%	89.43%	88.62%	87.64%	**90.22%**
Rail	82.06%	82.64%	83.57%	85.34%	87.25%	**88.16%**
Bare soil	80.12%	80.92%	82.45%	83.27%	85.74%	**87.23%**
Crops	84.15%	85.28%	86.58%	88.21%	89.63%	**90.74%**
Overall Accuracy (OA)	81.29%	82.85%	84.78%	86.95%	88.29%	**90.43%**
Kappa Coefficient (κ)	0.80	0.82	0.84	0.86	0.87	**0.89**

Table 5. LC accuracy comparison for each class and overall between MLP, GLCM-MLP, MRF, MCNN-LC, JDL-LC, and the proposed SS-JDL-LC method in S3. The largest classification accuracies and Kappa coefficients are shown in bold font.

LC Class (S3)	MLP	GLCM-MLP	MRF	MCNN-LC	JDL-LC	SS-JDL-LC
Clay roof	90.06%	87.45%	89.55%	90.05%	90.82%	**91.35%**
Concrete roof	70.58%	72.08%	74.21%	77.53%	78.96%	**82.74%**
Metal roof	90.12%	88.36%	90.09%	90.19%	90.88%	**91.28%**
Woodland	69.59%	73.14%	75.32%	86.02%	89.47%	**91.42%**
Grassland	72.22%	70.59%	75.45%	87.54%	90.35%	**90.56%**
Asphalt	89.46%	88.62%	89.42%	88.57%	88.24%	**90.73%**
Rail	83.18%	83.42%	84.36%	85.42%	87.89%	**88.59%**
Bare soil	80.21%	80.75%	82.25%	82.76%	84.92%	**87.46%**
Water	97.54%	96.28%	97.43%	96.53%	98.06%	**98.27%**
Overall Accuracy (OA)	82.22%	83.06%	84.54%	86.57%	88.48%	**90.62%**
Kappa Coefficient (κ)	0.81	0.82	0.83	0.85	0.87	**0.89**

Table 6. LU accuracy comparison for each class and overall between MRF, OBIA, Pixel-wise CNN, MCNN-LU, JDL-LU, and the proposed SS-JDL-LU method in S1. The largest classification accuracies and Kappa coefficients are shown in bold font.

LU Class (S1)	MRF	OBIA	CNN	MCNN-LU	JDL-LU	SS-JDL-LU
Commercial	71.11%	68.47%	74.16%	78.52%	82.72%	**85.95%**
Industrial	72.52%	72.05%	74.84%	79.68%	83.26%	**85.73%**
Residential	78.41%	80.38%	82.45%	84.02%	86.56%	**88.26%**
Redeveloped area	82.57%	84.15%	87.04%	88.96%	90.75%	**92.84%**
Park and recreational area	88.42%	89.54%	90.76%	90.47%	94.59%	**96.59%**
Parking lot	79.63%	82.06%	84.37%	86.58%	88.02%	**92.58%**
Highway	81.43%	79.26%	80.59%	83.04%	84.37%	**88.29%**
Railway	85.94%	88.14%	88.32%	89.54%	91.48%	**91.89%**
Herbaceous vegetation	82.71%	84.37%	85.24%	86.82%	88.57%	**89.02%**

Sandy beach	85.63%	88.28%	87.18%	88.25%	90.74%	**91.45%**
Overall Accuracy (OA)	82.06%	82.17%	84.32%	85.94%	87.68%	**88.94%**
Kappa Coefficient (κ)	0.80	0.81	0.84	0.86	0.88	**0.89**

Table 7. LU accuracy comparison for each class and overall between MRF, OBIA, Pixel-wise CNN, MCNN-LU, JDL-LU, and the proposed SS-JDL-LU method in S2. The largest classification accuracies and Kappa coefficients are shown in bold font.

LU Class (S2)	MRF	OBIA	CNN	MCNN-LU	JDL-LU	SS-JDL-LU
Commercial	70.07%	72.83%	73.25%	77.62%	82.43%	**84.76%**
Industrial	67.26%	69.04%	71.22%	80.14%	84.74%	**85.28%**
High-density residential	81.55%	80.37%	80.04%	82.32%	86.46%	**88.32%**
Medium-density residential	82.72%	84.38%	85.23%	86.75%	88.58%	**88.62%**
Park and recreational area	88.02%	91.12%	92.34%	92.74%	93.06%	**94.02%**
Parking lot	78.04%	80.12%	83.75%	85.29%	88.14%	**91.78%**
Highway	77.24%	78.06%	76.15%	77.84%	79.65%	**82.37%**
Railway	88.05%	90.63%	86.53%	89.02%	91.89%	**91.92%**
Agricultural area	85.08%	88.55%	87.43%	88.36%	90.94%	**91.85%**
Redeveloped area	80.08%	83.07%	86.24%	87.82%	88.62%	**90.69%**
Overall Accuracy (OA)	79.38%	80.26%	84.08%	85.29%	87.58%	**88.26%**
Kappa Coefficient (κ)	0.79	0.80	0.83	0.85	0.87	**0.88**

Table 8. LU accuracy comparison for each class and overall between MRF, OBIA, Pixel-wise CNN, MCNN-LU, JDL-LU, and the proposed SS-JDL-LU method in S3. The largest classification accuracies and Kappa coefficients are shown in bold font.

LU Class (S3)	MRF	OBIA	CNN	MCNN-LU	JDL-LU	SS-JDL-LU
Commercial	71.08%	72.43%	74.13%	78.44%	82.67%	**84.58%**
Industrial	72.57%	72.08%	74.85%	80.59%	83.22%	**85.73%**
Residential	78.39%	80.42%	80.52%	82.36%	84.91%	**88.76%**
Park and recreational area	88.43%	89.52%	90.78%	91.05%	93.43%	**94.47%**

Parking lot	79.68%	82.05%	84.36%	86.74%	88.09%	**90.92%**
Highway	76.43%	79.22%	80.57%	80.43%	82.02%	**83.59%**
Railway	85.96%	88.17%	88.31%	89.15%	90.39%	**91.65%**
Redeveloped area	82.57%	84.14%	87.09%	87.95%	89.72%	**91.24%**
Canal	90.68%	92.27%	94.16%	95.48%	96.58%	**96.84%**
Overall Accuracy (OA)	79.29%	80.42%	83.32%	85.08%	86.26%	**88.48%**
Kappa Coefficient (κ)	0.79	0.80	0.82	0.84	0.86	**0.88**

4 Discussion

Spatial scale is a fundamental concern in remotely sensed feature representations, as real-world features are often manifested over a range of scales (e.g., small football pitch and large-scale shopping centres). The importance of scale is well recognised in the remote sensing community through hand-coded and learnt features (e.g., Chen and Tian, 2015; Zhao *et al.*, 2016). However, the current need for scale selection and multi-scale representations are cumbersome and extremely inefficient, and often fail to capture the scale variations of objects and their local and global stationary characteristics. Such issues are crucial for deep learning methods that require a large amount of effort for parameterisation, such as choosing the optimal scale or multiple scales as CNN input window sizes for feature representations. These hyper-parameters within the deep networks are extremely difficult to tune effectively, which severely restricts their practical utility in remotely sensed image classification. To overcome these issues, a scale sequence joint deep learning (SS-JDL) method was developed to solve the complex LU and LC classification problem in an efficient and effective manner.

Scale sequence joint deep learning (SS-JDL) provides a novel paradigm that embeds multiple scales explicitly within joint deep learning across different classification hierarchies (e.g., LU and LC). Two major characteristics of SS-JDL include (1) information pathways from small to large scales by mimicking the human visual cognition system, and (2) integrated hierarchical learning between a pixel-based MLP and patch-based CNN across multiple scales.

Regarding the former, a forward scale sequence (FSS) was autonomously derived based on the minimum and maximum sizes of objects found within the remotely sensed images to be classified. The FSS represents a sequential observation and identification process from small scale features to large scale contexts and from LC states to LU representations, which is consistent with human visual cognition from simple parts and components towards more generalised and complex concepts as well as higher-level characteristics (Lappe *et al.*, 2013). With the scale sequence, the SS-JDL intrinsically involves multi-scale representations, where input patch sizes for the CNNs change from small to large along the iteration sequence to capture the scale effects manifest in high-order LU features. In contrast, the recently proposed JDL requires a pre-defined CNN window size to be found. This may require experimenting with a wide range of window sizes, to find the potentially "optimal" scale for both LU and LC representations. The entire process of scale selection takes potentially an extremely long time (20 JDL iterations at each scale), and it is impossible to fit a single "optimal" scale for LU and LC simultaneously as shown in Figure 5. Whereas the SS-JDL does not aim to find such an "optimal" scale, but integrates multiple scales through an iterative classification process to represent the scale effects across the scene. For the three study sites, the SS-JDL converged to the optimal solution rapidly (just five iterations or input scales; Figure 4), Thus, five scales are recommended as the default settings for the scale sequence depending on the complexity of the landscape. Within each iteration, the CNN networks learn the LU representations in deep and abstract levels (nine layers in the experiments), which captures the spatial pattern successively in a hierarchy at a specific scale, and continuously learns along the sequence of scales through the iterative process. Such a scale sequence needs only the minimum and the maximum scales, and autonomously interpolates the scale at each iteration, which is simple to implement for practitioners and end-users. Therefore, the proposed SS-JDL is highly suitable for remotely sensed image classification due to its simplicity and effectiveness.

For the latter hierarchical learning issue, the complex LU and LC classification problems were addressed jointly through iteration, where the pixel-based MLP and patch-based CNN were integrated through a hierarchy in a way that is mutually beneficial (Zhang *et al.*, 2019). Specifically, at each iteration, the spectral-based MLP was fitted to predict the LC at the pixel level, and based on this, the CNN was applied at the patch level to predict the LU of objects through spatial feature representations. Such joint learning was able to model the hierarchical relations between LU and LC iteratively while retaining the precise pixel-level spectral information. When the MLP is used alone for iteration, the process will lead to model overfitting towards training samples and failure to capture the spatial context relevant to LU (e.g., commercial areas involve large buildings and retail together with parking lots). Using the CNN only through iteration will result in blurred object boundaries within the classification results caused by the densely overlapping patches and spatial convolution, thereby missing fine-scale detail and degrading the classification accuracy (Zhang *et al.*, 2018c). By combining the MLP and CNN in a hierarchy, the blurred boundaries in the LU obtained by the patch-based CNN can be pulled back to the pixel-level detail in the LC by employing the MLP classifier. Similarly, the spatial context of the neighbourhood information in the LU is utilised by the MLP to support the production of a less noisy and more accurate LC classification. Such joint classification formulates a cyclic process of information as: "neighbourhood – pixel – neighbourhood – pixel", where the precise LU and LC are characterised through the appropriate hierarchical representation and in a joint fashion.

Together with the scale sequence and integrated hierarchical learning, the proposed SS-JDL is, therefore, parsimonious with high computational efficiency, and effective in that it delivers superior classification accuracy relative to benchmarks, some of which can be considered to be state-of-the-art. Both efficiency due to simplicity and effectiveness in accuracy were supported by the experimental results, in which the SS-JDL constantly achieved the highest classification accuracies for LU and LC with the least computational time in both study areas.

From an artificial intelligence perspective, the SS-JDL mimics the human visual system, combining the information across multiple scales to increase semantic meanings through joint reinforcement processes. Within the SS-JDL, the information learnt from lower scales passes forward to the higher scales, and high-level semantic information is learned gradually through continuously increasing window sizes of the CNN. Likewise, the human visual system can capture high level semantic representations (e.g., LU feature representations) without conscious effort, and such that the spatial outlines and the fine grained detail are integrated for vision and image understanding. Human brains are not required to exhaustively search for the so-called "optimal" scales, but rather are able to identify and label objects with both low and higher-order semantic meaning, drawing from labels that exist in a changing hierarchical ontological relationship, with great ability for generalisation and practical utility. The joint reinforcement in SS-JDL across scales, therefore, has great potential to catalyse a step change in the future of machine learning and AI, as well as applications in remote sensing and machine vision.

5 Conclusion

Scale effects are a fundamental concern in remotely sensed image classification and are manifested in the landscapes to be classified. For land use (LU) classification and land cover (LC), it has been demonstrated that *greatly* increased classification accuracy for both can be achieved by predicting LU using an object-based CNN, predicting LC via an MLP, and modelling explicitly the relationship between the predicted LU and LC variables as a joint distribution (Zhang *et al.*, 2019), thus, representing the obvious hierarchical relationship between LU and LC in both the scale and the ontological sense. However, its implementation requires the selection of an optimal patch size for the OCNN, which requires extensive searching and is, thus, computationally expensive. In this paper, an innovative scale sequence joint deep learning (SS-JDL) framework, that involves the same MLP and OCNN classification models, was proposed for joint LU and LC classification. Based on the minimum and the maximum sizes of image objects, the SS-JDL method autonomously incorporates multiple scales within its iterative process, such that it removes the requirement for tedious optimal scale selection. The experimental results demonstrate excellent classification accuracy and computational efficiency in comparison with the benchmark methods, including the recently proposed joint deep learning (JDL) method. The proposed method is simple to implement, and has great generalisation capability and practical utility with the default parameter settings. The SS-JDL, therefore, has the potential to transform image classification in the field of remote sensing, and machine learning generally, by creating a fast and effective implementation of the unifying joint deep learning (JDL) framework for classifying higher order feature representations, including LU in the context of remote sensing.

Acknowledgements

This research was funded by the Centre of Excellence in Environmental Data Science (CEEDS), jointly sponsored by Lancaster University and UKRI Centre for Ecology & Hydrology. The research was supported by the National Key R&D Program of China (Grant No. 2016YFB0502300) and partially funded by the National Natural Science Foundation of China (41871236). The authors are grateful to the Ordnance Survey for providing the aerial imagery and ground data.

References

Arel, I., Rose, D.C., Karnowski, T.P., 2010. Deep machine learning - A new frontier in artificial intelligence research. IEEE Comput. Intell. Mag. 5, 13–18.

https://doi.org/10.1109/MCI.2010.938364

Atkinson, P.M., Tatnall, A.R.L., 1997. Introduction Neural networks in remote sensing. Int. J. Remote Sens. 18, 699–709. https://doi.org/10.1080/014311697218700

Chen, S., Tian, Y., 2015. Pyramid of spatial relatons for scene-level land use classification. IEEE Trans. Geosci. Remote Sens. 53, 1947–1957. https://doi.org/10.1109/TGRS.2014.2351395

Cheng, G., Wang, Y., Xu, S., Wang, H., Xiang, S., Pan, C., 2017. Automatic road detection and centerline extraction via cascaded end-to-end Convolutional Neural Network. IEEE Trans. Geosci. Remote Sens. 55, 3322–3337. https://doi.org/10.1109/TGRS.2017.2669341

Del Frate, F., Pacifici, F., Schiavon, G., Solimini, C., 2007. Use of neural networks for automatic classification from high-resolution images. IEEE Trans. Geosci. Remote Sens. 45, 800–809. https://doi.org/10.1109/TGRS.2007.892009

Dong, Y., Zhang, Liangpei, Zhang, Lefei, Du, B., 2015. Maximum margin metric learning based target detection for hyperspectral images. ISPRS J. Photogramm. Remote Sens. 108, 138–150. https://doi.org/10.1016/j.isprsjprs.2015.07.003

He, N., Paoletti, M.E., Haut, J.M., Fang, L., Li, S., Plaza, A., Plaza, J., 2019. Feature extraction with multiscale covariance maps for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57, 755–769. https://doi.org/10.1109/TGRS.2018.2860464

Herold, M., Liu, X., Clarke, K.C., 2003. Spatial Metrics and Image Texture for Mapping Urban Land Use. Photogramm. Eng. Remote Sens. 69, 991–1001. https://doi.org/10.14358/PERS.69.9.991

Hu, F., Xia, G.-S., Hu, J., Zhang, L., 2015. Transferring deep Convolutional Neural Networks for the scene classification of high-resolution remote sensing imagery. Remote Sens. 7, 14680–14707. https://doi.org/10.3390/rs71114680

Hu, S., Wang, L., 2013. Automated urban land-use classification with remote sensing. Int. J. Remote Sens. 34, 790–803. https://doi.org/10.1080/01431161.2012.714510

Kim, M., Warner, T.A., Madden, M., Atkinson, D.S., 2011. Multi-scale GEOBIA with very high spatial resolution digital aerial imagery: scale, texture and image objects. Int. J. Remote Sens. 32, 2825–2850. https://doi.org/10.1080/01431161003745608

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet classification with deep Convolutional Neural Networks, in: NIPS2012: Neural Information Processing Systems. Lake Tahoe, Nevada, pp. 1–9.

Längkvist, M., Kiselev, A., Alirezaie, M., Loutfi, A., 2016. Classification and segmentation of satellite orthoimagery using Convolutional Neural Networks. Remote Sens. 8, 1–21. https://doi.org/10.3390/rs8040329

Lappe, M., Kruger, N., Leonardis, A., Janssen, P., Piater, J., Wiskott, L., Rodriguez-Sanchez, A.J., Kalkan, S., 2013. Deep Hierarchies in the Primate Visual Cortex: What Can We Learn for Computer Vision? IEEE Trans. Pattern Anal. Mach. Intell. 35, 1847–1871. https://doi.org/10.1109/tpami.2012.272

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444. https://doi.org/10.1038/nature14539

Li, Q., Mou, L., Liu, Q., Wang, Y., Zhu, X.X., 2018. HSF-Net: Multiscale deep feature embedding for ship detection in optical remote sensing imagery. IEEE Trans. Geosci. Remote Sens. 56, 7147–7161. https://doi.org/10.1109/TGRS.2018.2848901

Li, Y., Wang, N., Shi, J., Hou, X., Liu, J., 2018. Adaptive Batch Normalization for practical domain adaptation. Pattern Recognit. 80, 109–117. https://doi.org/10.1016/j.patcog.2018.03.005

Liu, X., He, J., Yao, Y., Zhang, J., Liang, H., Wang, H., Hong, Y., 2017. Classifying urban land use by integrating remote sensing and social media data. Int. J. Geogr. Inf. Sci. 31, 1675–1696. https://doi.org/10.1080/13658816.2017.1324976

Liu, Y., Guan, Q., Zhao, X., Cao, Y., 2018. Scene Classification Based on Multiscale Convolutional Neural Network. IEEE Trans. Geosci. Remote Sens. 56, 7109–7121.

Lv, X., Ming, D., Lu, T., Zhou, K., Wang, M., Bao, H., 2018. A new method for region-based majority voting CNNs for very high resolution image classification. Remote Sens. 10, 1–24. https://doi.org/10.3390/rs10121946

Ming, D., Li, J., Wang, J., Zhang, M., 2015. Scale parameter selection by spatial statistics for GeOBIA: Using mean-shift based multi-scale segmentation as an example. ISPRS J. Photogramm. Remote Sens. 106, 28–41. https://doi.org/10.1016/j.isprsjprs.2015.04.010

Nogueira, K., Penatti, O.A.B., dos Santos, J.A., 2017. Towards better exploiting convolutional neural networks for remote sensing scene classification. Pattern Recognit. 61, 539–556. https://doi.org/10.1016/j.patcog.2016.07.001

Pan, X., Zhao, J., 2018. High-Resolution Remote Sensing Image Classification Method Based on Convolutional Neural Network and Restricted Conditional Random Field. Remote Sens. 10, 1–20. https://doi.org/10.3390/rs10060920

Romero, A., Gatta, C., Camps-valls, G., Member, S., 2016. Unsupervised deep feature extraction for remote sensing image classification. IEEE Trans. Geosci. Remote Sens. 54, 1349–1362. https://doi.org/10.1109/TGRS.2015.2478379.

Stürck, J., Schulp, C.J.E., Verburg, P.H., 2015. Spatio-temporal dynamics of regulating ecosystem services in Europe- The role of past and future land use change. Appl. Geogr. 63, 121–135. https://doi.org/10.1016/j.apgeog.2015.06.009

Wang, H., Wang, Y., Zhang, Q., Xiang, S., Pan, C., 2017. Gated convolutional neural network for semantic segmentation in high-resolution images. Remote Sens. 9, 1–15. https://doi.org/10.3390/rs9050446

Wu, S.S., Qiu, X., Usery, E.L., Wang, L., 2009. Using geometrical, textural, and contextual information of land parcels for classification of detailed urban land use. Ann. Assoc. Am. Geogr. 99, 76–98. https://doi.org/10.1080/00045600802459028

Yang, Z., Mu, X. dong, Zhao, F. an, 2018. Scene classification of remote sensing image based on deep network and multi-scale features fusion. Optik (Stuttg). 171, 287–293. https://doi.org/10.1016/j.ijleo.2018.06.024

Zhang, C., Pan, X., Li, H., Gardiner, A., Sargent, I., Hare, J., Atkinson, P.M., 2018a. A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification. ISPRS J. Photogramm. Remote Sens. 140, 133–144. https://doi.org/10.1016/j.isprsjprs.2017.07.014

Zhang, C., Sargent, I., Pan, X., Gardiner, A., Hare, J., Atkinson, P.M., 2018b. VPRS-based regional decision fusion of CNN and MRF classifications for very fine resolution remotely sensed images. IEEE Trans. Geosci. Remote Sens. 56, 4507–4521. https://doi.org/10.1109/TGRS.2018.2822783

Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., Atkinson, P.M., 2019. Joint Deep Learning for land cover and land use classification. Remote Sens. Environ. 221, 173–187. https://doi.org/10.1016/j.rse.2018.11.014

Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., Atkinson, P.M., 2018c. An object-based convolutional neural network (OCNN) for urban land use classification.

Remote Sens. Environ. 216, 57–70. https://doi.org/10.1016/j.rse.2018.06.034

Zhao, B., Zhong, Y., Zhang, L., 2016. A spectral-structural bag-of-features scene classifier for very high spatial resolution remote sensing imagery. ISPRS J. Photogramm. Remote Sens. 116, 73–85. https://doi.org/10.1016/j.isprsjprs.2016.03.004

List of Figure Captions

Figure 1. The general workflow of scale sequence joint deep learning (SS-JDL) for land cover and land use classification.

Figure 2. Three study areas: Bournemouth (S1), Southampton (S2) and Manchester (S3) in England, with typical land use categories highlighted for each study site.

Figure 3. Model structures and architectures of the deep CNN network with nine hidden layers.

Figure 4. The influence of iteration upon overall accuracy for the LU and LC classifications using the proposed SS-JDL and the JDL method.

Figure 5. The effects of window size (scale) on overall accuracy of the LU and LC classifications using the SS-JDL (dashed lines) and the JDL method (solid lines).

Figure 6. Three subset (i.e., a, b, c) of LC classification in S1 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

Figure 7. Three subset (i.e., a, b, c) of LC classification in S2 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

Figure 8. The land cover classification in S3 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140).

Figure 9. Three subset (i.e., a, b, c) of LU classification in S1 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

Figure 10. Three subset (i.e., a, b, c) of LU classification in S2 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140). The correct and incorrect classifications are highlighted by circles in yellow and red, respectively.

Figure 11. The land use classification in S3 using Scale Sequence Joint Deep Learning (SS-JDL) from iteration 1 (28 × 28) to 5 (140 × 140).

Figure 12. Image subset benchmark comparison among various methods for S1, S2 and S3. The LC classifications include (a) MLP, (b) GLCM-MLP, (c) MRF, (d) MCNN-LC, (e) JDL-LC, and the (f) SS-JDL-LC. The LU classifications include (g) MRF, (h) OBIA, (i) CNN, (j) MCNN -LU, (k) JDL-LU, and the (l) SS-JDL-LU. Refer to Figures 6 to 11 for details of the corresponding classification legends.