Adaptive MUlti-scale Superpixel Embedding Convolutional Neural Network (AMUSE-CNN) for Land Use Classification

Huaizhong Zhang, *Member, IEEE,* Callum Altham, Marcello Trovati, *Member, IEEE,* Ce Zhang, Iain Rolland, Lanre Lawal, Dozien Wegbu, and Nemitari Ajienka

***Abstract*—Currently, large quantities of remote sensing images with different resolutions are available for earth observation and land monitoring, which are inevitably demanding intelligent analysis techniques for accurately identifying and classifying land use (LU). This article proposes an adaptive multi-scale superpixel embedding convolutional neural network architecture (AMUSE-CNN) for tackling land use classification. Initially, the images are parsed via the superpixel representation so that the object based analysis (via a superpixel embedding CNN scheme) can be carried out with the pixel context and neighborhood information. Then, a multi-scale convolutional neural network (MS-CNN) is proposed to classify the superpixel based images by identifying object features across a variety of scales simultaneously in which multiple window sizes are used to fit to the various geometries of different LU classes. Furthermore, a proposed adaptive strategy is applied to best exert the classification capability of MS-CNN. Subsequently two modules are developed to fully implement the AMUSE-CNN architecture. More specifically, Module I is to determine the most suitable classes for each window size (scale) by applying majority voting to a series of MS-CNNs. Module II carries out the classification of the classes identified in Module I for the given scale used in MS-CNN and therefore complete the LU classification of the entire classes. The proposed AMUSE-CNN architecture is both quantitatively and qualitatively validated using remote sensing data collected from two cities, Kano and Lagos in Nigeria due to the spatially complex land use distribution. The experimental results show the superior performance of our approach against several state-of-the-art techniques.**

***Index Terms*—Land use classification, Convolutional neural network, Superpixel embedding CNN, VFSR remotely sensed imagery.**

I. Introduction

INDUSTRIAL advancement and a constant need for energy have negatively impacted the environment, which has led to severe degradation to the natural vegetation ecosystem. In turn, this has created substantial effects on the land use (LU) such as loss of prime agricultural lands, destruction of important wetlands, as well as increased flood risk [1]. However, having access to reliable LU maps, categorising scene images into a discrete set of meaningful LU classes, greatly assists these tasks in natural resources monitoring and sustainable land management [2]. As the rate of LU change grows ever faster, particularly in developing countries [3], the importance of having accurate, up-to-date LU information grows too. Although these maps can be produced by hand, this is both time-consuming and expensive to do [4]. As access to high-power computing resources has significantly increased in recent years, it is perhaps unsurprising that various computational techniques have been presented which aim to automate this task. In early days, due to low spatial resolution of satellite images (such as Landsat series), the pixel level classification paradigms dominated the studies in the field. Some popular machine learning methods had been successfully applied for categorising scene images at pixel level, including support vector machines (SVMs) and random forests (RFs) [5]. However, whilst achieving fine spatial resolution of remote sensing images, the performance of these pixel level methods has not improved due to the inherent issues in high-resolution images such as the within-class variability and low between-class separability [6]. As a result, the object level delineation and analysis of remote sensing images have emerged to study semantic entities or scene components rather than individual pixels, as the solution paradigm named object-based image analysis (OBIA) [7]–[9]. A typical method, OBIA-SVM [10], [11], which incorporates this object based paradigm into the traditional SVM solution, has successfully moved from pixel-based techniques towards object-based representation for better land use classification.

In the last decade, however, the successes of deep learning methods in many areas have led those in the remote-sensing community to investigate their potential to become the new state-of-the-art in the field. From convolutional neural networks (CNNs) [12], to recurrent neural networks (RNNs) [13], to autoencoders (AEs) [14], it is now commonplace to find deep learning methods being used to process remote sensing data and they often perform tasks such as LU classification with phenomenal precision. Among these methods, the pixelwise CNN had been widely applied by combining a pixel-based classifier with a context-based classifier, such as the Multilayer-Perceptron (MLP) based CNN [15], [16] providing different feature representations with strong complementarity. Recently, an object-based CNN (OCNN) [17] method has also been broadly accepted in LU classification, which incorporates the OBIA technique into the CNN framework for better learning LU objects via within-object and between-object information.

However, while deep learning methods are capable of producing high-accuracy LU classification maps, they often require large volumes of labelled data to do so, which can

H. Zhang is with the Department of Computer Science Department, Edge Hill University, Ormskirk LP39 4QP, United Kingdom e-mail: zhangh@edgehill.ac.uk.

be nearly as time-consuming and expensive as producing LU classifications by hand [5]. There have, therefore, been various studies which have implemented techniques such as transfer learning [18] and active learning [19] in order to attempt to replicate high-accuracy results with smaller quantities of training data. In this study, a two-stage approach is developed for automatic LU classification, where the multi-scale superpixel CNN (MS-CNN) model is firstly proposed for allowing object features to be learnt across multiple scales simultaneously and then an adaptive strategy based on majority-voting is developed to exert the role of MS-CNN to find most suitable scale for each class in order to obtain best LU classification. To this end, MS-CNN is designed to replicate the ability of deeper CNN models to extract features [20] but does so without using such a deep network in order to avoid requiring the very large quantities of training data necessary to protect against overfitting and to ensure such that the model generalises well. MS-CNN is therefore built upon an initial image segmentation using superpixels [21], where the superpixel segmentation is used to extract image patches in order to inform targeted sampling locations such that fewer locations need CNN classification. Thus, MS-CNN is an object-based CNN with contextual features for LU classification. Nonetheless, in the viewpoint of its model architecture, MS-CNN aims to extract multi-scale features to classify all objects without considering the geometric characteristics of each class, for example residential class may be better identified using a small scale but highway class could be well recognised using a large scale in this study. For exploring this scale adaptability in MS-CNN and improving the performance of LU classification, an Adaptive MUlti-scale Superpixel Embedding Convolutional Neural Network (AMUSE-CNN) architecture is developed to sufficiently exert the classification capabilities of MS-CNN when dealing with complicated geographical situations. To this end, the AMUSE-CNN architecture is designed to consist of two modules that each one performs its own respective function. Module I aims to find the most suitable classes for each window size (scale) by applying the majority voting to the outcomes of a series of MS-CNNs. Module II is used to apply the scale identified in Module I to the associated classes so that these classes are able to be better classified comparing to the use of singular MS-CNN. With the seamless integration of the MS-CNN model and the adaptive strategy of scale selection, the proposed AMUSE-CNN architecture is effective and efficient to perform the LU classification, as an adaptive, multi-scale and object based LU classification system.

We compare our method against the state-of-the-art techniques, OCNN [17], OBIA-SVM [11] and pixelwise CNN [16], on two real world datasets in Nigeria.

The main contributions of this study can be summarized as:

- An adaptive and object based land use classification system is developed in terms of a superpixel embedding representation of remote sensing images.
- For better extracting object features with regard to diverse geometries of LU objects, the multi-scale based CNN (MS-CNN) model is designed by applying the multiple window sizes to perform LU classification.
- An adaptive strategy of scale selection is proposed to improve the performance of MS-CNN using majority voting. The generated LU classification system is named AMUSE-CNN.
- Two urban areas (Lagos and Kano) are used and annotated to validate the proposed AMUSE-CNN architecture.

The remainder of this paper is organised as follows: Section II focuses on the details of the method including the AMUSE-CNN architecture and its key components. Section III presents the experimental results, and in Section IV, a detailed discussion on the results is presented. Finally, Section V concludes the article.

II. Proposed Method

The proposed method is composed of two parts. First, a Multi-scale Superpixel based Convolutional Neural Network (MS-CNN) is designed to initially implement the LU classification capability with the multiple window sizes. Based on this MS-CNN model, an Adaptive Multi-scale Superpixel Embedding Convolutional Neural Network (AMUSE-CNN) is proposed to provide an overall adaptive and accurate schematic solution for LU classification, by considering the fact that the MS-CNN capability is limited due to being reliant on the input of all data into pre-determined window sizes.

A. AMUSE-CNN Architecture

As mentioned above, MS-CNN is reliant on window sizes that therefore results in a flawed LU classification as not all individual classes perform best when just using a given window size for a given class. This concern has driven the design and implementation of the AMUSE-CNN architecture with an adaptive strategy for enhancing the scale capability of MS-CNN in LU classification. In fact, the AMUSE-CNN architecture consists of two modules whose block diagrams are shown in Fig. 1. Module I aims to automatically assign each LU class to its most suitable window size through majority voting that can then result in the training of further MS-CNN models in Module II, where each class is trained to its highest potential using its most suitable window size obtained from Module I. The relevant functions of each module are described below.

1) Module I: In this module, three specified window sizes are used to separately conduct the best selection of window size for each class. In this study, the window sizes include 48, 96 and 120. As shown in Fig. 1, two functions are designed to carry out the selection implementation for each specified window size.

- **Initial Model Training:** A series of four separate MS-CNN architectures are trained with a given window size in order to generate a series of initial accuracies for each class, resulting in $4 \times N$ votes for each class where N is the number of window sizes, being 3 in this study.
- **Majority Voting:** Majority voting is applied to determine the most suitable window size for each class by comparing the accuracies obtained during initial training, with the highest voted window size for each class being assigned to said class.

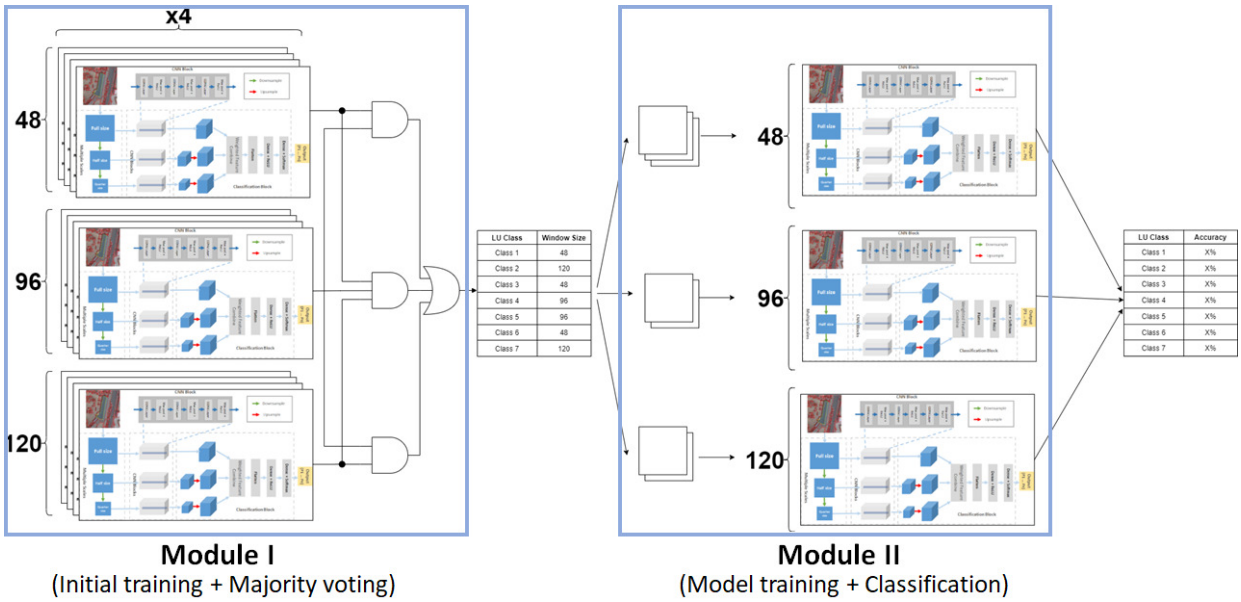

Fig. 1: AMUSE-CNN architecture consists of two modules. Module I is designed to conduct the initial training that aims to find most suitable classes for the given window size with majority voting. Module II is designed for performing the LU classification using the classifiers obtained from Module I for relevant classes.

2) Module II: This module is created to conduct the new MS-CNN model training so that the accurate LU classification can be achieved. To this end, a MS-CNN model is trained for each window size in which only the previously assigned classes in Module I are involved in order to gain the highest possible accuracies for these classes, for example in this study, Classes 1, 3 and 6 get involved in the model training for the case of the windows size 48 as shown in Fig. 1. Afterwards, the models obtained are used to carry out the classification of the respective classes so that all LU classes are well finally classified and the classification results are then output into a single table as shown in Fig. 1.

B. MS-CNN Model

This MS-CNN model is proposed to implement a multi-scale LU classification CNN, in which input data is rescaled in order to produce data at various sizes. In this study, three window sizes are used, namely full, half and quarter resolutions that means that when any image is entered into the MS-CNN model as input data, the image dimensions will be halved and quartered in order to be provided as input to each individual CNN block. The block diagram of the MS-CNN model can be seen in Fig. 2.

1) LU classification problem descriptor: The set X of input data consists of $A \times B$ vectors, where A and B are the height and width of the data image respectively. This can be expressed as

$$X = \{\mathbf{x}_1, ..., \mathbf{x}_{A\times B}\}, \mathbf{x}_i \in \mathbb{R}^3. \tag{1}$$

Each $\mathbf{x}_i$ also has a corresponding u_i and v_i corresponding to the pixel's vertical and lateral position in the image data respectively, where

$$0 \leq u_i < A, \quad 0 \leq v_i < B. \tag{2}$$

The set Y of output data contains each pixel corresponding land use, expressed as

$$Y = \{y_1, ..., y_{A\times B}\}, \quad y_i \in \mathbb{N}, \quad 1 \leq y_i \leq K, \tag{3}$$

where K is the number of LU classes considered.

The training set, $\mathcal{L} \subset X$, is composed of a subset of labelled pixels for which target labels $Y_{\mathcal{L}} \subset Y$ are given. The task of LU classification is to estimate the unknown labels.

2) Superpixel-based LU classification: By grouping pixels which have sufficient similarity to one another (in terms of their spatial proximity and spectral homogeneity) it can be expected that such groups of pixels will tend to belong to a common output class. It is upon this assumption which superpixel-based segmentation methods are based. Exactly what metric is used to measure similarity and what is deemed 'sufficient' similarity will vary with the choice of segmentation algorithm and its parameters, but it suffices for now to say that algorithms which attempt to perform such a task exist.

Using the term superpixel to refer to these segmented pixel groups, each superpixel set, S is a subset of X. The set X is divided into non-overlapping superpixel sets whose collective union equals X, i.e.

$$S_\alpha \cap S_\beta = \emptyset, \quad \forall \beta \neq \alpha \tag{4}$$

and

$$S_\alpha \cup S_\beta \cup ... = X. \tag{5}$$

The shorthand notation $i \in S$ will be used to refer to the set $\{i \mid \mathbf{x}_i \in S\}$. The common output class assumption supposes that

$$y_i = y_j, \quad \forall i \in S, \forall j \in S. \tag{6}$$

3) LU classification of superpixels using data windows: In order to harness the power of convolutional layers to perform

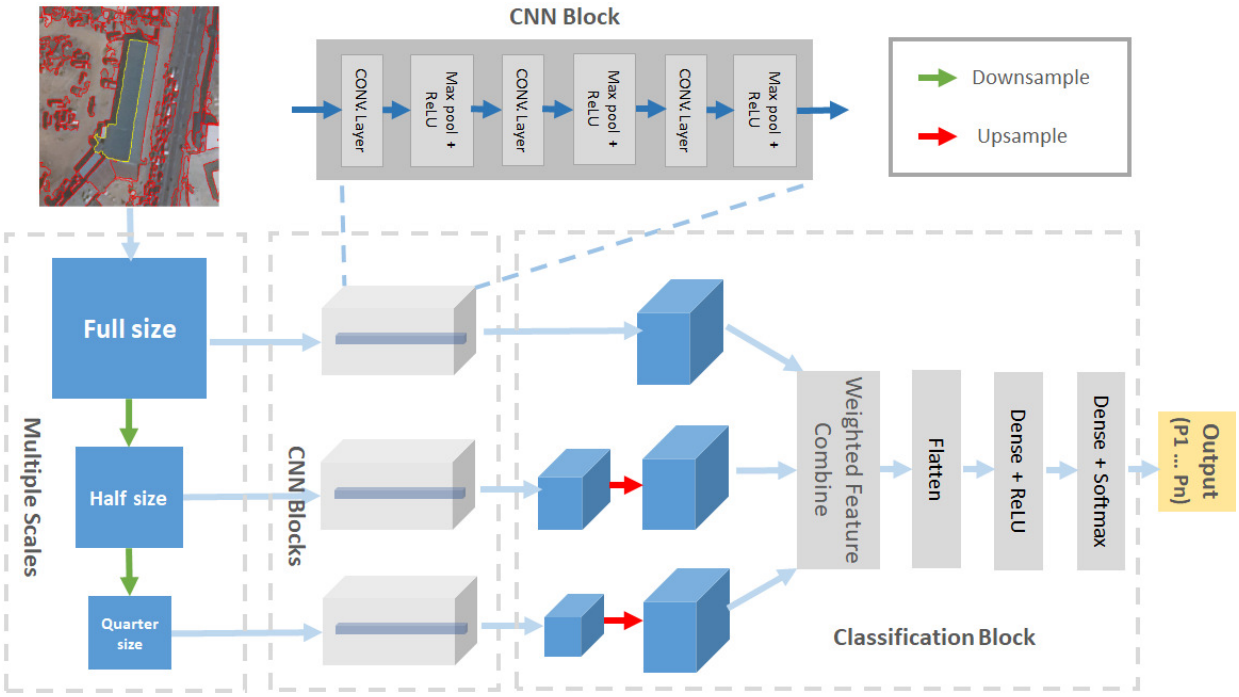

Fig. 2: Schematic of the MS-CNN model. Blue blocks represent data, other coloured blocks represent one or more neural layers.

data-driven feature extraction, a second assumption is made which considers that the LU label of the pixels within a superpixel depends only on a local neighbourhood of the input data, a square window, S_w, centred on the segment's centroid. This allows any superpixel of arbitrary geometry to be classified using a square of the input data in its vicinity. Put probabilistically,

$$P(y_i \in S|X) = P(y_i \in S|\mathbf{x}_i \in S_w). \tag{7}$$

The superpixel centroid vertical and horizontal position, denoted u_S and v_S respectively, can be found according to

$$u_S = \frac{\sum_{i\in S} u_i}{|S|}, \quad v_S = \frac{\sum_{i\in S} v_i}{|S|}, \tag{8}$$

where $|S|$ refers to the cardinality of the superpixel set or equivalently the number of pixels contained within the superpixel. The shorthand $i \in S_w$, used to index pixels contained within the superpixel local neighbourhood, refers to the set

$$\{i : -W/2 \leq u_i - u_S < W/2, -W/2 \leq v_i - v_S < W/2\}, \tag{9}$$

where W is the width and height in pixels of the data neighbourhood considered relevant for LU classification.

4) Data-driven feature extraction using CNNs: Convolutional neural networks (CNNs) are designed such that successive convolutional layers are capable of learning increasingly abstract and translationally-invariant features using relatively few parameters [22]. This makes them very powerful in many computer vision tasks, particularly performing LU classification like OCNN [17] and pixelwise CNN [16]. While these methods are capable of producing state of the art performance, their ability to learn multi-scale features is limited by how many filters they contain and the depth of their architecture which are in turn limited by the amount of training data available. With insufficient training data, deeper CNNs are liable to overfit training data and perform poorly on unseen data regions [15].

5) MS-CNN architecture: In order to more easily allow a model to identify features at various scales, we propose the MS-CNN architecture as shown in Fig. 2, which directly operates at multiple scales simultaneously. For a given scale, if features that would distinguish one class from another exist at that scale, the CNN block (see Fig. 2) which has filters of size similar to those features ought to more readily learn to identify them.

The design of the MS-CNN model is based upon the notion that, when working with data resolution as high as $50cm$ per pixel, some of the features useful for performing LU classification might be comparatively large in terms of the number of pixels they make up. While a deep CNN would likely be capable of identifying these features, the MS-CNN model should be able to learn these features with comparatively fewer layers used in succession.

Fig. 2 illustrates that the proposed MS-CNN architecture has three main components, Multiple Scales, CNN Blocks and Classification Block. Multiple Scales provide multi-scale routes for MS-CNN to learn features, parallel pathways are used, where each pathway operates at a different scale (e.g. half or quarter resolution). The component of CNN Blocks goes to extract the features for each pathway with the respective scale. Each convolutional block is made up of repeated units, consisting of a convolutional layer, a maximum pooling and a rectified linear unit (ReLU) activation function. In

Classification Block, the yielded feature maps of the parallel pathways are then combined using a simple linear combination, which is encompassed within the 'Weighted Feature Combine' layer which has one scalar parameter per resolution pathway such that its output, $\tilde{\mathbf{X}}$, is given by

$$\tilde{\mathbf{X}} = \sum_i w_i \mathbf{X}_i, \tag{10}$$

where $\mathbf{X}_i$ refers to one of its inputs. This combines the features identified at each of the network's scales into one feature map according to a 'pathway importance' weight. Subsequently, as shown in Fig. 2, the resulting feature map is flattened and classified using fully connected layers and finally applying a softmax activation function to the model output.

III. Results and Analysis

A. Area and data of study

Using very fine spatial resolution (VFSR) remotely sensed optical images which were released as part of Digital Globe's Open Data Program, two cities of Lagos and Kano, Nigeria, were selected for LU classification analysis. The data, seen in Fig. 3, has three bands (red, green and blue) and a spatial resolution of $50cm$ per pixel.

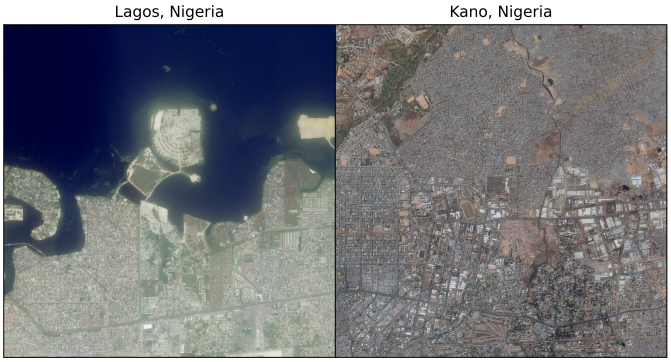

Fig. 3: Visualisation of the data for two cities of Lagos and Kano, Nigeria. Each image covers an area of 16384 × 16384 pixels.

B. Data annotation

The OpenStreetMap () definition of LU ("what the area is used for by humans") will be adopted as the definition of LU used in this study. Following a consideration of the LU classes recognised by OpenStreetMap and a visual analysis of the Lagos and Kano data, the set of LU classes described by Table I and Table II was adopted.

By labelling polygons within the raster which were accurate representations of each of these LU classes, a set of labels were obtained which could be used to represent both training and validation sets. The data annotation was not exhaustive, with the total annotated area making up 36.8% of the total area.

The labelled polygons were randomly assigned into one of five sets, with each acting as a validation set for one experiment in a 5-fold cross-validation analysis, with the others acting as that particular k-fold experiment's training data.

TABLE I: Identified LU classes within the Lagos dataset and brief descriptions of what they encompass.

LU Class	Description
Brownfield or construction	Exposed soil/dirt or dusty land
Highway	Any kind of street or way
Industrial	Larger buildings typically representing warehouses/shared business sites
Lagoon	A body of shallow sea water
Natural conservation	Land in a natural state
Residential	Areas containing private houses or accommodation
Retail	Low/medium density self-contained shops and/or market activity areas

TABLE II: Identified LU classes within the Kano dataset and brief descriptions of what they encompass.

LU Class	Description
Bare/open land	Exposed soil/dirt or dusty land
Commercial	High density, built-up region comprised of businesses
Industrial	Larger buildings typically representing warehouses/shared business sites
Residential	Areas containing private houses or accommodation
Retail	Low/medium density self-contained shops and/or market activity areas
Road	Regions designated for vehicular use
Vegetation	Shrubs, grassland or agricultural lands

C. Initial image segmentation

Different segmentation algorithms place different importances on various measures of pixel homogeneity (e.g. texture, colour, spatial proximity). In order to segment the data into superpixels which would represent locally coherent subregions with a shared LU class (as per the assumption of Eq. 6), the mean-shift algorithm [23], implemented using the open-source software Orfeo Toolbox, was selected. The parameters of the algorithm were adjusted in order to obtain superpixels which were small enough to sufficiently separate individual LU classes but not so small that the quantity of superpixels which made up the image was unnecessarily high which would increase inference times (the model inference time increases approximately linearly with the number of superpixels in the data). The mean-shift algorithm has three main hyperparameters: the spatial radius, the range radius and the minimum region size. Following a trial-and-error approach, these were set to 30, 8 and 196 respectively.

D. Parameter settings

1) MS-CNN model: For the purposes of this investigation, MS-CNN models were trained using three scales (full, half and quarter resolution). Various models with various window sizes have been trained, in order to determine the optimal window size for performance. The CNN block shown in Fig. 2 for the models trained consisted of three repeated blocks, each consisting of a convolutional layer (32 3×3 kernels), a maximum pooling layer (2×2 window) and a ReLU activation function. The first dense layer had 32 neurons and the second had 7 neurons (one for each LU class).

2) AMUSE-CNN architecture: For effectively implementing the adaptive strategy, four separate MS-CNN models are used to conduct the initial model training and thus $4 \times 3 = 12$ outcomes participate in majority voting (here 3 window sizes used in this study).

E. LU Classification Results and Analysis

For well evaluating the AMUSE-CNN model, three state-of-the-art LU classification models were used as benchmarks to conduct the performance comparison: OBIA-SVM [11], Pixelwise CNN [15], and OCNN [17]. In addition, the performance of MS-CNN at each scale was also presented for showing the role of the adaptive strategy used in AMUSE-CNN. In the following subsections, a quantitative analysis is first carried out, followed by a qualitative comparison.

1) Quantitative Results: Tables III and IV show the classification accuracies achieved for each of the models assessed. The classification accuracy is measured as the percentage of the area in the validation set representing a given class which is correctly classified. Average accuracy (AA) is calculated by averaging the classification accuracies over all classes. Equivalently, the overall accuracy (OA) is measured as the percentage of the area of all polygons in the validation set which is correctly classified. The accuracies presented are the averages achieved over 5-fold cross-validation.

It is clearly shown in Tables III and IV that the AA 93.25% of AMUSE-CNN in Kano is significantly better than that 85.44% in Lagos while its OA 90.16% in Kano is obviously inferior to that 96.71% in Lagos. In fact, these accuracy differences are determined by the geographical complexities of two cities and the nature of the relevant LU classes. In Lagos, some classes are very similar in color and geometries such as Residential and Retail, which leads to ineffective differentiation of the relevant classes, resulting in lower AA in Lagos comparing to Kano. Fig. 8 presented in Section III-E2 visually shows how this issue affects the performance of the relevant methods. On the other hand, the lagoon class takes most of the area in Lagos so its high detection accuracy dominates the OA of Lagos, which therefore greatly helps Lagos achieve better OA than Kano.

To demonstrate the benefits of the adaptive strategy applied in AMUSE-CNN, the results for the MS-CNN model at each single scale are presented in Tables III and IV. AMUSE-CNN clearly outperforms MS-CNN in both Lagos and Kano. For example, for Residential in Lagos, MS-CNN with the scale 120 achieves the best accuracy 60.99% while AMUSE-CNN reaches 78.34%; for Industrial in Kano, 72.81% is the best accuracy obtained by MS-CNN with the scale 96 while 91.11% is achieved by AMUSE-CNN.

Comparing to the state-of-the-art methods, the traditional handcrafted method, OBIA-SVM, is clearly ineffective to deal with LU classification in these two datasets due to its AAs for both datasets being 42.67% and 46.60% respectively. In particular, Retail (0.17%) in Lagos is almost all misclassified due to being similar to Residential in colour. As for Pixelwise CNN and OCNN, their performance is much better than OBIA-SVM. However, our AMUSE-CNN significantly outperforms these two methods in both AA and OA such as AA: 85.44% vs. 78.59% vs. 82.20% in Lagos and 93.25% vs. 81.70% vs. 83.70% in Kano. Tables III and IV show that AMUSE-CNN is able to well classify some difficult classes however Pixelwise CNN and OCNN perform poorly, for example, Residential and Retail in Lagos, Industrial and Residential in Kano.

On a class-by-class basis, AMUSE-CNN optimally classifies six of the seven LU classes for Lagos except for Brownfield Construction (Pixelwise CNN being best 85.43%). The nearest AA accuracy benchmark, OCNN can classify the classes averagely better than any of the other methods presented for both Lagos and Kano. For the most part, the Highway and Lagoon LU classes in Lagos, and the Road and Vegetation in Kano are easiest to classify (with all models achieving $> 75\%$ accuracy). The classes which have proved most difficult to classify (Industrial and Residential) are both classes where AMUSE-CNN has greatly excelled relative to the benchmarks particularly in Kano.

For further demonstrating the overall performance and strength of AMUSE-CNN, the Receiver Operating Characteristic (ROC) Curve is plotted for each model used in this study, which Figs. 4 and 5 are presented for Lagos and Kano respectively. These figures obviously illustrate that AMUSE-CNN gets higher TPR (Sensitivity) and lower FPR (1-Specificity) at any threshold setting in comparison to the benchmarks. Thus, the classification performance of AMUSE-CNN in both Lagos and Kano appears markedly superior to those of the benchmarks.

2) Qualitative Results: In order to visually compare the performance, Fig. 6 presents one randomly selected region of Lagos (2000×2000 pixel area) alongside the ground truth (GT) annotations for that region and the LU classification output for each of the models trained. Where used, the subscripts refer to the window size of the model. This visual results show that the overall performance of AMUSE-CNN is much better than the benchmarks and also MS-CNN although the classification of some classes is still not best conducted due to the similarity between classes such as a small portion of 'Retail' being misclassified as 'Industrial'. Fig. 7 shows another selected region of the Kano data for visually showing the performance of our method and the benchmarks. It is clearly illustrated that AMUSE-CNN well classified most classes except for a small portion of 'Retail' being misclassified as 'Residential'. Promisingly, referring to the GT as shown in Fig. 7, AMUSE-CNN can perform the classification much more accurate than the benchmarks and MS-CNN. For noticeably looking into the classification details and issues of the given classes, we zoom on the selected parts of the images and present the relevant classification results of these zooming-in portions as follows.

'Residential' and 'Retail' are the classes with similar geographical information in Lagos and the benchmarks thus often wrongly classify these two classes. Fig. 8 illustrates the classification results of a selected region in which AMUSE-CNN can well classify these two classes however the benchmarks perform poorly.

In Lagos, 'Industrial' is the class with the lowest performance of AMUSE-CNN. Fig. 9 illustrates that the indistinguishable boundary of 'Highway' and 'Retail' with 'Industrial'

TABLE III: Classification accuracy (%) comparison among OBIA-SVM, Pixelwise CNN, OCNN, MS-CNN, AMUSE-CNN for Lagos. The presented accuracies (AA and OA) are the averages achieved over 5-fold cross-validation.

LU Class	OBIA-SVM	Pixelwise CNN	OCNN	MS-CNN$_{48}$	MS-CNN$_{96}$	MS-CNN$_{120}$	AMUSE-CNN$_{[48, 96, 120]}$
Brownfield/construction	34.01	85.43	73.55	73.88	72.05	75.77	77.37
Highway	80.62	80.33	78.14	75.34	77.43	64.41	85.39
Industrial	16.28	73.65	61.07	56.70	75.70	60.22	76.85
Lagoon	99.31	99.84	99.79	99.87	99.84	99.86	99.95
Natural Conservation	11.70	84.65	92.63	91.42	88.13	90.98	97.62
Residential	56.66	51.56	46.09	51.26	57.35	60.99	78.34
Retail	0.17	74.69	48.25	66.24	68.41	79.60	82.56
AA	42.67	78.59	82.20	73.53	76.99	75.98	85.44
OA	81.80	93.71	92.89	93.73	93.49	94.43	96.71

TABLE IV: Classification accuracy (%) comparison among OBIA-SVM, Pixelwise CNN, OCNN, MS-CNN, AMUSE-CNN for Kano. The presented accuracies (AA and OA) are the averages achieved over 5-fold cross-validation.

LU Class	OBIA-SVM	Pixelwise CNN	OCNN	MS-CNN$_{48}$	MS-CNN$_{96}$	MS-CNN$_{120}$	AMUSE-CNN$_{[48, 96, 120]}$
Bare/open land	29.2	85.9	87.2	76.3	81.0	82.57	98.41
Commercial	43.8	77.5	82.0	72.4	78.9	78.73	92.62
Industrial	15.5	62.8	71.1	62.6	72.8	60.59	91.11
Residential	29.8	67.4	72.9	59.2	75.5	61.85	85.07
Retail	50.7	89.7	85.8	80.6	88.1	78.80	91.95
Road	77.9	92.9	94.9	91.4	94.3	81.77	96.35
Vegetation	78.8	95.4	91.9	91.2	94.2	91.59	97.29
AA	46.6	81.7	83.7	76.3	83.5	77.84	93.25
OA	32.9	73.3	77.7	68.19	76.52	69.38	90.16

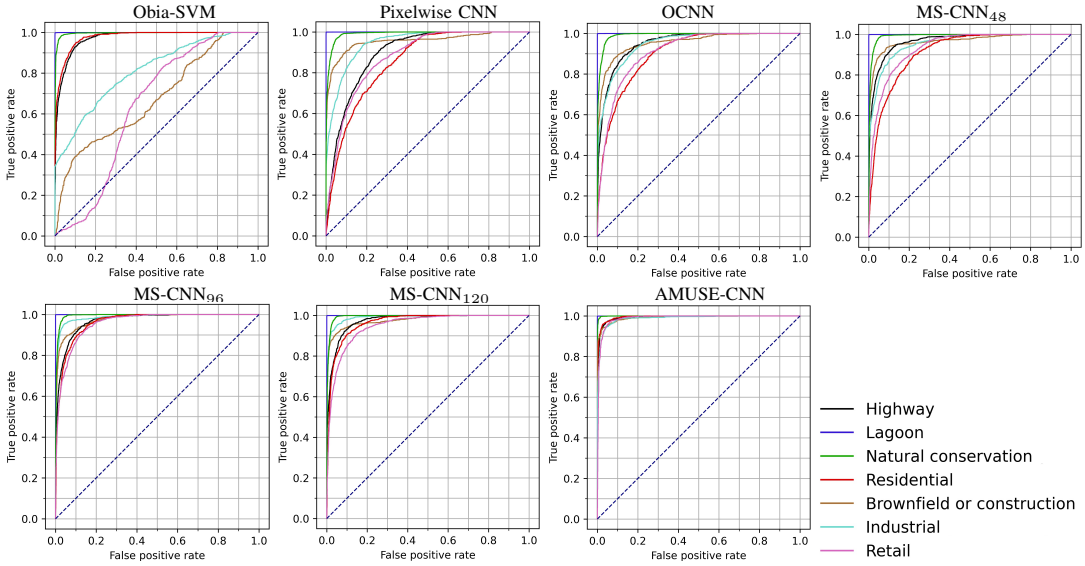

Fig. 4: ROC curves showing the performance of all methods: AMUSE-CNN produces excellent classification performance for all classes in comparison with other methods in Lagos.

actually intervene in the classification thus significantly reducing the performance of AMUSE-CNN in 'Industrial'. Even so, AMUSE-CNN obviously outperforms the benchmarks as visually presented in Fig. 9.

Fig. 10 presents the classification results of the selected region in Kano that the performance of AMUSE-CNN in 'Industrial' and 'Residential' is with the lowest accuracy in comparison with other classes. It illustrates that the benchmarks is prone to classifying 'Residential' as 'Industrial' while a small portion of 'Industrial' is wrongly classified as 'Residential' with AMUSE-CNN. However, our proposed method is clearly better at conducting the classification than the benchmarks for these two classes.

F. Computational environment and complexity

To compare the computational costs of AMUSE-CNN with those of the benchmarks, an analysis which examines the amount of time to train each model and for each model to perform inference on the complete data is presented in Tables V and VI. The experiments have been carried out on a machine with Intel i5-9400F CPU, 32 GB RAM and a NVIDIA Quadro M400 GPU.

Overall, the amount of time needed to train/perform inference is at a similar level for MS-CNN and OCNN. Among the benchmarks, Obia-SVM is quite time-consuming while the pixelwise CNN stands out as the fastest training but the slowest inference model. For the AMUSE-CNN, due to the multiple

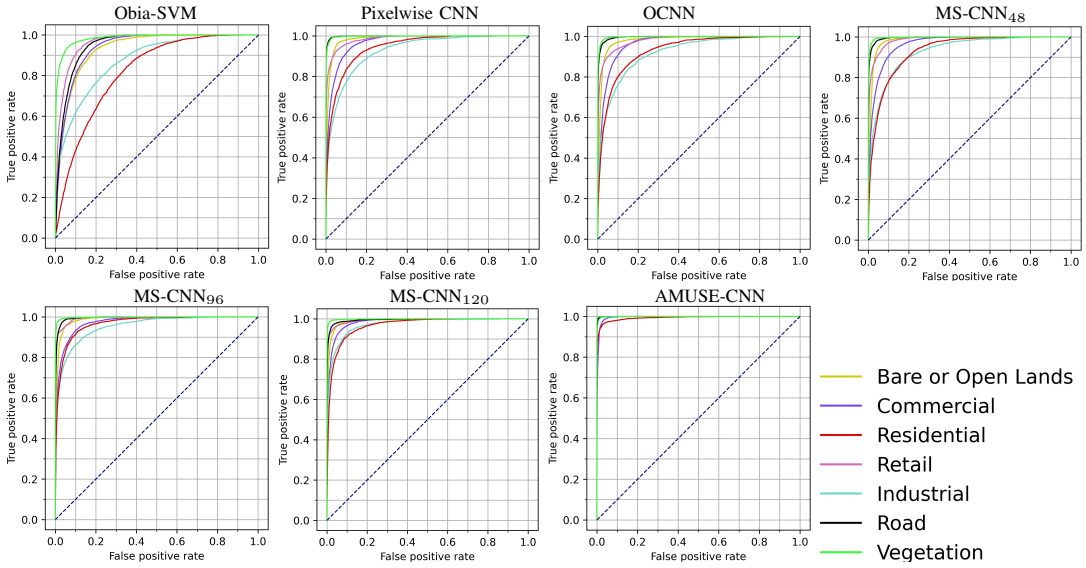

Fig. 5: ROC curves showing the performance of all methods: AMUSE-CNN produces excellent classification performance for all classes in comparison with other methods in Kano.

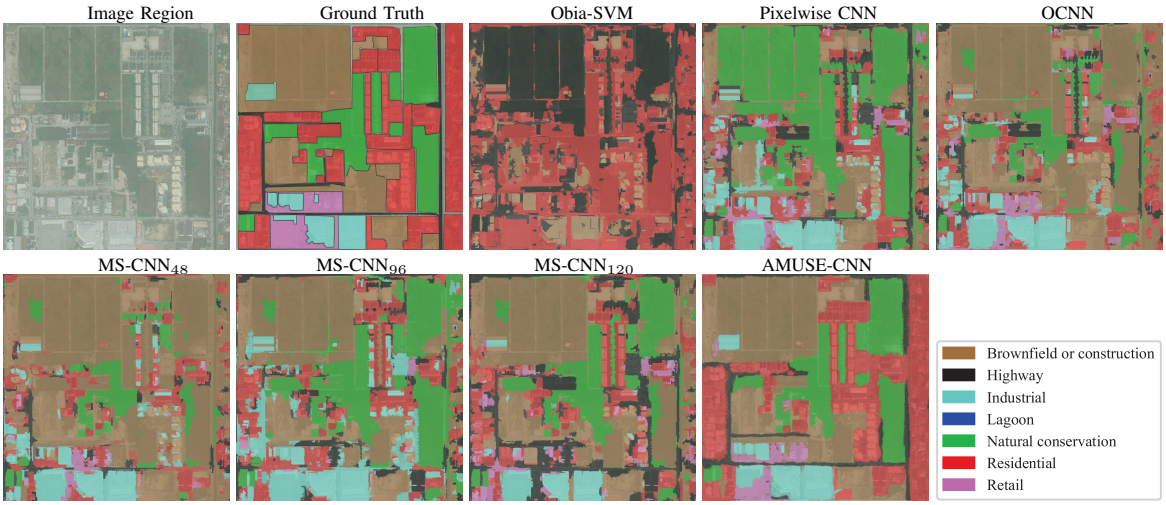

Fig. 6: A 2000 $\times$ 2000 pixel Lagos image region and the corresponding ground truth as well as LU classification for all methods.

MS-CNN models being involved in the adaptive strategy, its computation is costly and pretty time-consuming, however in compensation its accuracy of LU classification is highly improved in comparison with the benchmarks.

IV. Discussion

In the proposed solution scheme, the MS-CNN model is first proposed to improve upon the benchmark methods by offering direct routes for multi-scale feature identification through the use of scale-specific CNN blocks. Furthermore, the AMUSE-CNN architecture is designed to explore the capability of LU classification with the use of majority voting to implement an adaptive strategy. The experimental results shown in Sections III-E1and III-E2 demonstrate that AMUSE-CNN does accurately outperform the state-of-the-art methods. However, for a better LU classification, there is much room to improve AMUSE-CNN whatever the classification accuracy or the computational efficiency is.

It is clear that the overall performance of AMUSE-CNN is greatly dependent on that of MS-CNN. In Tables III and IV, the improved AA and OA offered by MS-CNN with different scales suggests that there may be cause for further investigation into multi-scale architectures for LU classification. That being said, the improvements observed are not seen across every class, and so it is proved that some LU classes are better identified using multi-scale feature learning than others by applying the adaptive strategy, which is the essence of the AMUSE-CNN architecture proposed. The further work could consider to use a different model selection scheme instead of majority voting.

With regard to the window sizes used, although MS-CNN models presented have used half and quarter resolution as their additional scales, this choice was somewhat arbitrary

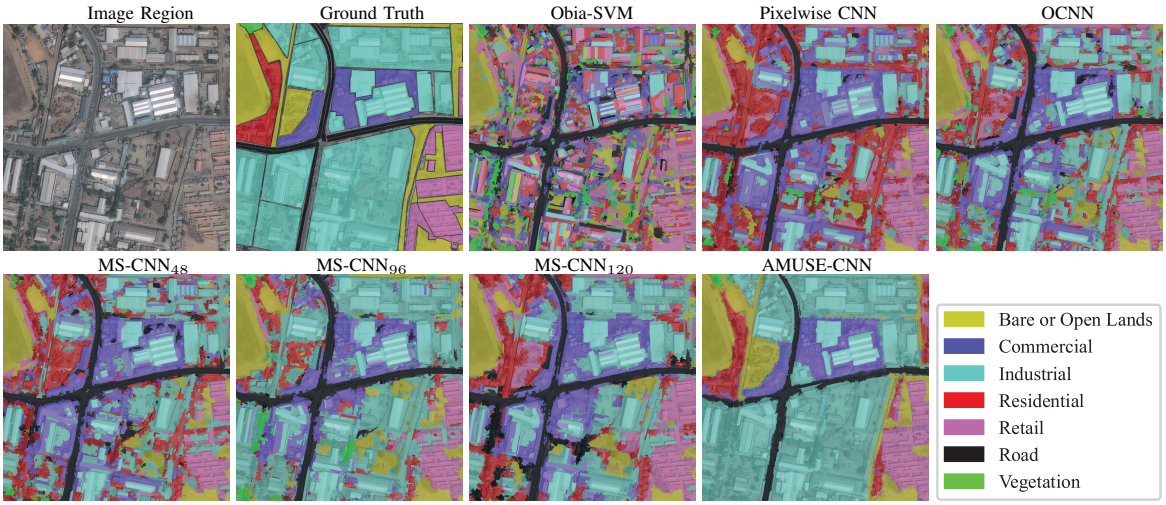

Fig. 7: A 2000 $\times$ 2000 pixel Kano image region and the corresponding ground truth as well as LU classification for all methods.

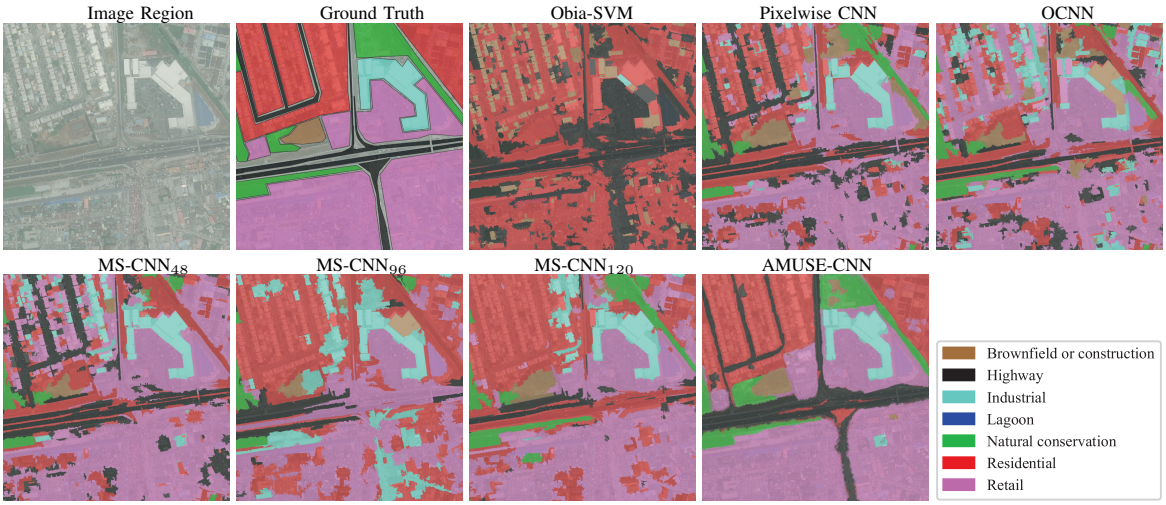

Fig. 8: Results of the selected region showing the interference between 'Residential' and 'Retail' in Lagos.

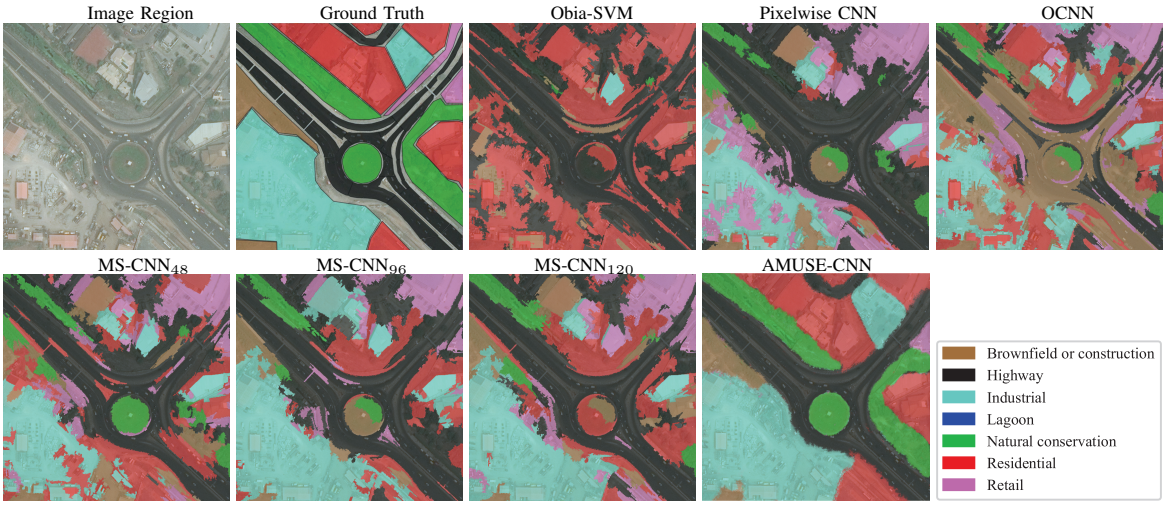

Fig. 9: Results of the selected region showing the interference of 'Highway' and 'Retail' on the classification of 'Industrial' in Lagos.

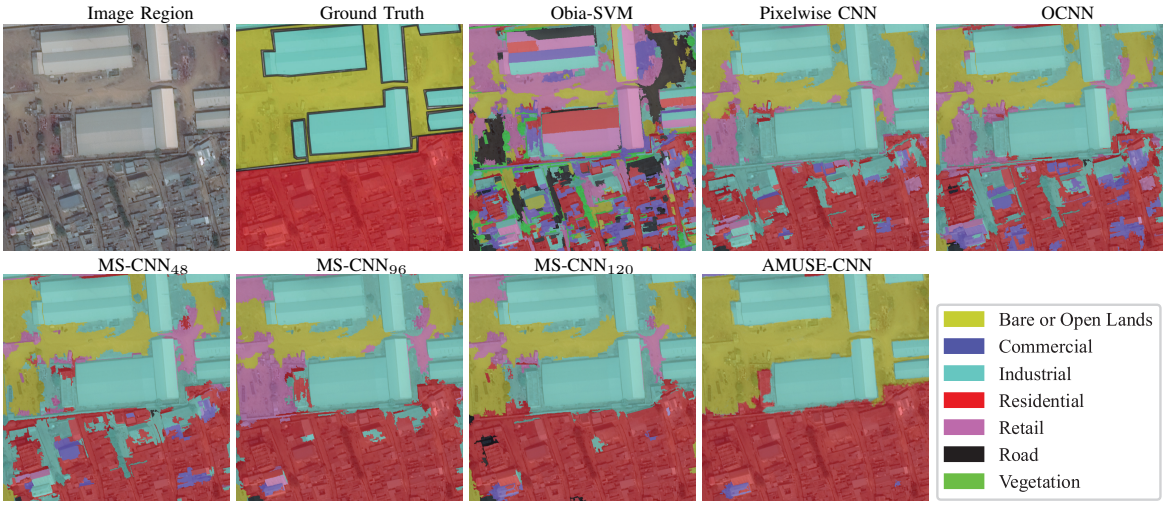

Fig. 10: Results of the selected region showing the classification interference between 'Industrial' and 'Residential' in Kano.

TABLE V: Computation times to train each of the classification algorithms presented for the Lagos dataset and make LU classification predictions for the complete data.

	OBIA-SVM	Pixelwise CNN	OCNN	MS-CNN$_{48}$	MS-CNN$_{96}$	MS-CNN$_{120}$	AMUSE-CNN$_{[48, 96, 120]}$
Train time (h)	1.03	0.07	0.19	0.10	0.23	0.32	3.25
Predict time (h)	1.13	6.27	0.22	0.13	0.25	0.55	4.60

TABLE VI: Computation times to train each of the classification algorithms presented for the Kano dataset and make LU classification predictions for the complete data.

	OBIA-SVM	Pixelwise CNN	OCNN	MS-CNN$_{48}$	MS-CNN$_{96}$	MS-CNN$_{120}$	AMUSE-CNN$_{[48, 96, 120]}$
Train time (h)	1.33	0.09	0.31	0.16	0.43	0.65	5.93
Predict time (h)	2.0	12.34	0.35	0.35	0.45	0.63	7.15

and further investigation might analyse how the choice of scales in MS-CNN affects results. This should therefore result in improving the performance of AMUSE-CNN. Similarly, the choice of operating with three scales was also somewhat heuristic and an extension of this work might assess the balance between performance and increased computational costs by comparing MS-CNNs which operate with more or fewer scales. In addition, four MS-CNN models are used to conduct majority voting in the current study. We could give a try to taking different number of models involved in the scale selection process such as three or five in further studies.

To better validate the AMUSE-CNN method, although two different urban datasets are used in this study, it is anticipated that the improved results could be replicated for more datasets. The next step would be to produce similar improvements using datasets collected from other continents. This should not only offer some challenges unique to new datasets, but also present some of the same complexities faced with classifying the Lagos and Kano datasets.

Another big challenge to our proposed method is how to reduce the computational complexity and make the solution scheme more efficient. The inherent drawback of majority voting is actually seriously bringing down the calculation efficiency of AMUSE-CNN thus some more efficient method of scale choice should be used in future. In addition, we could consider to improve the computational capability with better hardware such as TPU for more efficient calculation.

V. Conclusion

This article presents an adaptive LU classification CNN architecture, namely AMUSE-CNN, based on a multi-scale feature learning model (MS-CNN) proposed, as a means for better identifying the subtle visual cues that allow a region's LU to be determined. Using superpixels obtained via a segmentation of the data to represent regions of locally homogenous LU, MS-CNN learns to identify features at multiple scales simultaneously in order to classify superpixels and obtain a semantic segmentation of the data. Moreover, based on MS-CNN, AMUSE-CNN is designed and put forward to find the most suitable scale for classes and thus obtain incomparable classification accuracy by resorting to a scale selection process of majority voting. The AMUSE-CNN performance is well examined with two urban datasets, Lagos and Kano in Nigeria, by comparing to three state-of-the-art benchmarks. A barrier to AMUSE-CNN is the time-consuming step of performing the scale selection, two ways of improving work for reducing the computational complexity could be valuably attempted in future. One could use another efficient scale section scheme

instead of majority voting. Another one may optimise the calculation process with more efficient code along with the use of high-performance hardware units such as TPU.

REFERENCES

[1] U. Aid and USGS, "West africa: Land-use and land-cover dynamics," *U.S. Aid and USGS*. [Online]. Available: ⟨https://eros.usgs.gov/westafrica/land-cover/land-use-land-cover-and-trends-nigeria⟩

[2] A. Rahman, S. Kumar, S. Fazal, and A. Siddiqui, "Assessment of land use/land cover change in the north-west district of delhi using remote sensing and gis techniques," *Journal of the Indian Society of Remote Sensing*, vol. 40, no. 4, pp. 689–697, 2012.

[3] A. Rahman, S. Aggarwal, M. Netzband, and S. Fazal, "Monitoring urban sprawl using remote sensing and gis techniques of a fast growing urban centre, india," *IEEE J. Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, pp. 56 – 64, 2011.

[4] P. Langat, L. Kumar, R. Koech, and M. Ghosh, "Monitoring of land use/land -cover dynamics using remote sensing: A case of tana river basin, kenya," *Geocarto International*, vol. 8, no. 1–20, pp. 1–20, 2019.

[5] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. Johnson, "Deep learning in remote sensing applications: A meta-analysis and review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 152, no. 4, pp. 166–177, 2019.

[6] G. Cheng, X. Xie, J. Han, L. Guo, and G. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE J. Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, no. 1, pp. 3735–55, 2020.

[7] T. Blaschke, "Object based image analysis for remote sensing," *J. Photogram. Remote Sens.*, vol. 65, pp. 2–16, 2010.

[8] H. Li, H. Gu, Y. Han, , and J. Yang, "Object oriented classificaiton of high-resolution reote sensing imagery based on an improved colour structure code and a support vector machine," *Int. J. Remote Sens.*, vol. 31, no. 29, pp. 1453–1470, 2010.

[9] T. Blaschke, G. Hay, M. Kelly, S. Lang, P. Hofmann, E. Addink, R. Feitosa, and D. Tiede, "Geographic object-based image analysis - towards a new paradigm," *J. Photogram. Remote Sens.*, vol. 87, pp. 180–191, 2014.

[10] A. Tzotsos, "A support vector machine approach for object based image analysis," in *Proceedings of 1st International Conference on Object-based Image Analysis*, 2006, pp. 1–6.

[11] S. Zylshal, S. Sulma, F. Yulianto, J. Nugroho, , and T. Sofan, "A support vector machine object based image analysis approachon urban green space extraction using pleiades-1a imagery," *Model. Earth Syst. Environ.*, vol. 2, no. 54, pp. 1–12, 2016.

[12] Z. Shao and J. Cai, "Remote sensing image fusion with deep convolutional neural network," *IEEE J. Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 5, pp. 1656–1669, 2018.

[13] D. Ho, D. Ienco, R. Gaetano, N. Lalande, E. Ndikumana, F. Osman, and P. Maurel, "Deep recurrent neural networks for winter vegetation quality mapping via multitemporal sar sentinel-1," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 3, pp. 464–468, 2018.

[14] S. Hao, W. Wang, Y. Ye, T. Nie, and L. Bruzzone, "Two-stream deep architecture for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 4, pp. 2349–2361, 20108.

[15] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6232–6251, 2016.

[16] C. Zhang, X. Pan, H. Li, A. Gardiner, J. Hare, and P. M. Atkinson, "A hybrid mlp-cnn classifier for very fine resolution remotely sensed image classification," *Photogrammetry and Remote Sensing*, vol. 140, no. 3, pp. 133–144, 2017.

[17] C. Zhang, I. Sargent, X. Pan, H. Li, A. Gardiner, J. Hare, and P. M. Atkinson, "An object-based convolutional neural network (OCNN) for urban land use classification," *Remote Sensing of Environment*, vol. 216, no. 5, pp. 57–70, 2018.

[18] H. Lyu, H. Lu, L. Mou, W. Li, J. Wright, X. Li, X. Li, X. Zhu, J. Wang, L. Yu, and P. Gong, "Long-term annual mapping of four cities on different continents by applying a deep information learning method to landsat data," *J. Remote Sensing*, vol. 10, no. 3, pp. 2072–4292, 2018.

[19] P. Liu, H. Zhang, and K. B. Eom, "Active deep learning for classification of hyperspectral images," *IEEE J. Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 2, pp. 712–724, 2017.

[20] M. Jogin, Mohana, M. Madhulika, G. Divya, R. Meghana, and S. Apoorva, "Feature extraction using convolution neural networks (cnn) and deep learning," in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, 2018, pp. 2319–2323.

[21] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-artsuperpixel methods," *IEEE Trans. PAMI*, vol. 11, no. 11, pp. 2274 – 2282, 2012.

[22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[23] D. Comaniciu and M. P., "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.