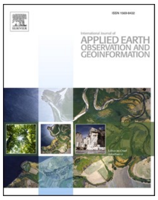
Contents lists available at ScienceDirect

International Journal of Applied Earth Observations and Geoinformation

journal homepage: www.elsevier.com/locate/jag



CapViT: Cross-context capsule vision transformers for land cover classification with airborne multispectral LiDAR data

Yongtao Yu [a,*], Tao Jiang [a], Junyong Gao [a], Haiyan Guan [b], Dilong Li [c], Shangbing Gao [a], E Tang [a], Wenhao Wang [a], Peng Tang [a], Jonathan Li [d]

[a] *Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian, JS 223003, China*
[b] *School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing, JS 210044, China*
[c] *College of Computer Science and Technology, Huaqiao University, Xiamen, FJ 361021, China*
[d] *Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L3G1, Canada*

ARTICLE INFO

Keywords:
Multispectral light detection and ranging (MS-LiDAR)
Land cover
Vision transformer
Feature self-attention
Capsule network
Cross-context feature

ABSTRACT

Equipped with multiple channels of laser scanners, multispectral light detection and ranging (MS-LiDAR) devices possess more advanced prospects in earth observation tasks compared with their single-band counterparts. It also opens up a potential-competitive solution to conducting land cover mapping with MS-LiDAR devices. In this paper, we develop a cross-context capsule vision transformer (CapViT) to serve for land cover classification with MS-LiDAR data. Specifically, the CapViT is structurized with three streams of capsule transformer encoders, which are stacked by capsule transformer (CapFormer) blocks, to exploit long-range global feature interactions at different context scales. These cross-context feature semantics are finally effectively fused to supervise accurate land cover type inferences. In addition, the CapFormer block parallels dual-path multi-head self-attention modules functioning to interpret both spatial token correlations and channel feature interdependencies, which favor significantly to the semantic promotion of feature encodings. Consequently, with the semantic-promoted feature encodings to boost the feature representation distinctiveness and quality, the land cover classification accuracy is effectively improved. The CapViT is elaborately testified on two MS-LiDAR datasets. Both quantitative assessments and comparative analyses demonstrate the competitive capability and advanced performance of the CapViT in tackling land cover classification issues.

1. Introduction

Light detection and ranging (LiDAR) techniques have showcased remarkable progress and success in recent decades as one of the leading earth observation means. An advantageous feature of the LiDAR systems lies in that they can directly portray the three-dimensional (3D) topologies and measure the spectral properties of targets in a fully active way. The exported data are usually stored with the format of 3D point cloud, which records both the real-world 3D coordinates and the backscattered spectral intensities. With the unprecedented increase in data volume and the strict requirement on interpretation efficiency and standard, there have burst a series of automated or semi-automated LiDAR data processing solutions developed for different application purposes (Ma et al., 2018; Mirzaei et al., 2022; Yan et al., 2015). In comparison with single-channel LiDAR devices, the multispectral LiDAR (MS-LiDAR) counterparts possess more powerful potentials in earth observation missions. To be specific, by sensing multiple spectral bands from different laser channels, abundant inherent attributes of targets can be attained, which significantly enrich the feature quantity and discriminability. Accordingly, MS-LiDAR data potentially provide an advanced solution to a variety of land cover mapping and environmental factor analysis tasks, such as land use survey, landmark recognition, precision agriculture, forest inventory, etc. Amongst, precise and regular land cover mapping means vitally to take control of the conditions and changes in environment, society, and economy. Thus, it is absolutely necessary and practically valuable to evaluate the applicability and reliability of MS-LiDAR data in diverse land cover mapping issues.

In light of the state-of-the-art capability of transformer architectures in natural language processing, intent attempts have recently been taken to transfer and improve the transformer philosophies to serve different vision applications, including segmentation, detection, and classification, resulting in an emerging family of vision transformers (ViT) (Liu

* Corresponding author.
E-mail address: allennessy@hyit.edu.cn (Y. Yu).

https://doi.org/10.1016/j.jag.2022.102837
Received 6 April 2022; Received in revised form 6 May 2022; Accepted 21 May 2022
Available online 30 May 2022

et al., 2022). The milestone of the ViT model developed by Dosovitskiy et al. (2021) formulated a self-attention principle to focus on long-range global feature interactions. It performed excellently in coarse-grained classification tasks; however, the memory and computation consumptions increased quadratically for tackling fine-grained detection or segmentation tasks. As optimized solutions to providing versatile and efficient prediction backbones, cross-shaped windows (CSwin) transformer (Dong et al., 2022), shifted windows (Swin) transformer (Liu et al., 2021), focal transformer (Yang et al., 2021), pyramid vision transformer (PVT) (Wang et al., 2021a), cross-scale transformer (CrossFormer) (Wang et al., 2021b), multi-path ViT (MPViT) (Lee et al., 2021), etc. were elaborately proposed to either explore local feature interactions, stripe feature interactions, multiscale feature interactions, or the combination of local and global feature interactions for the purpose of feature boosting and cost reducing. In addition, some hybrid ViT models (Guo et al., 2021; Peng et al., 2021) were also designed by stacking both the convolutional units and transformer blocks aiming at combining the local and global feature representation superiorities. Due to the predominant performance of ViT models in varying vision prediction tasks, they have been positively introduced to resolve remote sensing data interpretation issues, including target detection (Fang et al., 2022), image classification (Lv et al., 2022; Sun et al., 2022), instance segmentation (Chen et al., 2022), and semantic segmentation (Wang et al., 2022).

In this paper, for the objective of improving the map-level land cover mapping accuracy, we design a novel cross-context capsule vision transformer (CapViT) model for land cover classification by taking advantage of the rich geometrical and spectral properties of the MS-LiDAR data. The CapViT is composed of three streams of capsule-based transformer encoders that dedicate to investigate long-range global feature semantics at different context scales. The cross-context feature semantics are finally integrated and comprehensively interpreted to direct land cover type prediction. The CapViT demonstrates competitive classification performances on two MS-LiDAR datasets. Thus, the developed CapViT provides an effective solution to the land cover classification tasks. Moreover, the work in this paper also enlarges the application domains of the MS-LiDAR data and examines the feasibility and effectiveness of the MS-LiDAR data in handling the land cover classification issues. In summary, the contributions mainly involve the following. (1) A capsule-based vision transformer architecture is stacked to obtain high-quality and entity-aware feature encodings. (2) A cross-context vision transformer formulation is proposed to investigate global feature interactions with different context details for supplying strong and distinctive feature semantics. (3) A transformer block with dual-path multi-head self-attention modules is designed to take into consideration both spatial token correlations and channel feature interdependencies for promoting the feature embedding quality.

2. Related works

2.1. Feature image-based strategies

A common strategy for processing MS-LiDAR data is to convert them into top-view feature images based on the data attributes like intensities and elevations. Such a strategy can represent the discrete, unstructured 3D points as gridded image formulations, which can well improve the interpretation efficiency. Specifically, it is suitable for map-level land cover analyses. However, the localization accuracy might be slightly affected caused by the image rasterization operations. Matikainen et al. (2017) trained a random forest (RF) classifier cooperated with histogram analysis to conduct land cover classification. The features fed into the RF involved mainly the segment-based intensity-derived and elevation-derived properties. Likewise, Morsy et al. (2017) designed a maximum likelihood classifier (MLC) by combining the intensity map with the height map. Except for intensity and elevation features, Huo et al. (2018) constructed a support vector machine (SVM) model by including also vegetation indices and morphological profiles for urban area mapping. The vegetation indices were computed by using the intensity properties in different laser channels. Ghaseminik et al. (2021) proposed a segment-directed RF (SRF) inference formula by slicing the feature images into semantic segments to investigate spectral and geometrical characteristics. To abstract deep feature encodings, Pan et al. (2019) formulated a deep Boltzmann machine (DBM) classifier, which was stacked as a multi-layer perception (MLP) architecture. In their following work (Pan et al., 2020), a convolutional neural network (CNN) structure was designed to classify land covers. This structure involved convolutional blocks for local feature extraction and linear connections for category prediction. Yu et al. (2020) developed a hybrid capsule network (HCapsNet) structure that comprised a fully-connected stream to extract global features and a convolutional stream to exploit local features. The feature semantics were significantly promoted to supervise more accurate predictions. As an improvement, Yu et al. (2022) embedded an efficient self-attention (ESA) unit and an adversarial learning scheme into the capsule network, namely ESA-CapsNet, to further enhance the feature representation capability. To be specific, the ESA module realized feature attention by considering the channel and spatial feature saliencies. Karila et al. (2017) evaluated the feasibility of road extraction by using MS-LiDAR data. In their framework, image segments were first generated and a set of features were obtained accordingly. The localization of roads was finalized using an RF classifier. Lindberg et al. (2021) applied MS-LiDAR data to conduct tree species categorization missions. Similarly, statistical properties were computed based on a cell rasterization pattern and further processed through linear discriminant analysis (LDA) for species type determination. Chen et al. (2018) also quantified the carbon storage by analyzing the tree distributions in urban areas. In addition, some other studies dedicated to fuse the MS-LiDAR data with other data types, such as hyperspectral images, optical images, single-band LiDAR, and waveform LiDAR, to well enhance the land cover mapping accuracy (Hänsch and Hellwich, 2021; Hong et al., 2020; Jin and Mountrakis, 2022; Matikainen et al., 2020).

2.2. LiDAR point-based strategies

As another land cover mapping strategy, LiDAR points are directly interpreted with individual semantic labels. Such a strategy can nicely preserve the spatial topologies of targets and conduct mapping at a real-world scale, especially beneficial to the delineation of the lower-storey or shielded targets. Thus, it is suitable for the semantic understanding of the 3D scenes. However, remarkable computation overhead might be generated by directly processing 3D LiDAR points. Shi et al. (2021) proposed a multiscale selection scheme to characterize performance-effective spatial and spectral properties of MS-LiDAR points. The optimized feature embedding was differentiated via an SVM classifier for cover type inference. Wang and Gu (2020) suggested to encode the spectral and geometric attributes of MS-LiDAR points through second-order tensor embedding. The tensor representation with two modes behaved promisingly in distinguishing the intraclass and interclass structure distributions. Given the knowledge of the local geometry relationships among points, Ekhtari et al. (2018) formulated a rule-based classifier for multi-return points labelling. The rules used as evidences in the classifier were derived according to height, distance, and distribution priors. In the work of Luo et al. (2022), non-ground points were first separated from ground points to compute different feature attributes like intensity and height. Then, after feature merging, a decision tree model was leveraged to finalize point categorization. Jing et al. (2021) stacked an encoder-decoder architecture with stage-wise skip connections on the basis of the PointNet++. As for the encoder, channel feature boosting mechanism was appended at each stage for feature semantic optimization. Zhao et al. (2021) designed a graph convolution network (GCN) for point-level prediction based on local graph representations. Worth mentioning, feature reasoning units were included in the GCN to learn global contextual and local edge features. By making use of multiscale feature semantics, Li et al. (2022a) developed a pyramidal network composed of attentive graph geometric moments convolutions. The input feature to the classification network was concatenated with spatial coordinates, spectral attributes, and geometrical properties, which were further encoded into moments embedding by the convolution operations. Shaker et al. (2019) examined two processing pipelines to separate land and water regions in MS-LiDAR data. The first one leveraged a Gaussian mixture model based on the intensity/elevation histograms, whereas the second one employed a scan line analysis approach by considering the intensity-elevation ratios. Li et al. (2020) explored the feasibility of building instance segmentation by using MS-LiDAR data. The segmentation network followed a GCN architecture. Dai et al. (2018) adopted a mean shift segmentation workflow for delineating individual trees. The separation of tree crowns was achieved using the features extracted from both spatial and multispectral domains. Besides, MS-LiDAR data were also applied to geological examination (Hartzell et al., 2014), heritage preservation (Shao et al., 2020), forest inventory (Kukkonen et al., 2019), land nutrient quantification (Sankey et al., 2021), and leaf biochemical constituent estimation (Sun et al., 2019).

3. Materials and data preparation

3.1. MS-LiDAR data

The MS-LiDAR data used in our experiments were acquired by an airborne Titan multispectral laser scanning device from the Teledyne Optech. This device was mounted with three channels of laser scanners, which operated independently under different laser spectrum bands with different deflection angles. To be specific, Channel 1 worked under intermediate infrared spectrum (SWIR) with a wavelength of 1550 nm and a deflection angle of 3.5° (forward); Channel 2 worked under near infrared spectrum (NIR) with a wavelength of 1064 nm and a deflection angle of 0° (vertically downward); Channel 3 worked under visual spectrum (GREEN) with a wavelength of 532 nm and a deflection angle of 7° (forward). During data acquisition, an individual set of point cloud was generated from each channel, resulting in three separated sets of point clouds with different mapping details. Based on the backscattered laser intensities, these three channels can measure the spectral properties of targets from different perspectives (e.g., vegetation shows strong reflectance in Channel 2), which shows more advantages to the single-band LiDAR counterparts.

As shown in Fig. 1(a), the study areas for collecting the MS-LiDAR data are located in Ontario, Canada. Two sites with different land cover conditions were surveyed, including an inland area of Whitchurch-Stouffville (Fig. 1(b)) and a coastal area of Tobermory (Fig. 1(c)). The survey in Whitchurch-Stouffville involved 19 intersecting flying strips occupying an area of about 3.21 km^2. The survey in Tobermory comprised ten intersecting flying strips occupying an area of about 1.99 km^2. The specific areas of these two test sites are marked in Fig. 1(b) and (c). We named the collected MS-LiDAR data in these two areas as the WS (for Whitchurch-Stouffville) and TM (for Tobermory) datasets, respectively. Each of the two datasets contained three sets of point clouds acquired by the Titan system. To be specific, for the WS dataset, the total number of points is 414,090,351. The minimum, maximum, and average point densities are 12, 53, and 43 points/m^2, respectively, in each channel. The point intensity variations in the SWIR, NIR, and GREEN channels are in the ranges of [1, 452], [1, 289], and [1, 315], respectively. For the TM dataset, the total number of points is 268,650,373. The minimum, maximum, and average point densities are 16, 58, and 45 points/m^2, respectively, in each channel. The point intensity variations in the SWIR, NIR, and GREEN channels are in the ranges of [1, 431], [1, 274], and [1, 297], respectively.

3.2. Feature image rasterization

In this paper, aiming at providing map-level analyses of land covers, we employ a feature image-based interpretation scheme to accomplish land cover classification by using MS-LiDAR data, which can also improve the processing efficiency. To this end, the three clusters of raw MS-LiDAR point clouds are rasterized to form a group of top-view feature images according to the data attributes in different channels. As a matter of fact, the three clusters of MS-LiDAR points are not geographically the same due to the deflection angle differences of the three channels of laser scanners. However, the three clusters of MS-LiDAR points are acquired based on the same global navigation satellite system (GNSS) and each point has a geographical coordinate under the same coordinate system. Thus, first, a data registration operation is performed to merge the three clusters of MS-LiDAR points together simply based on their geographical coordinates, resulting in a single point cloud set. Specifically, the outlier points in each channel are removed before performing data registration by using the CloudCompare software (http://www.cloudcompare.org). Then, the merged point cloud is vertically gridded along the *Z*-axis direction to form a cell representation, where each cell contains several points from different channels. The cell size (or the spatial resolution) is set as 0.5 m by default by considering the minimum point density in a channel and the mapping accuracy. Finally, each cell is rasterized into an individual pixel in the corresponding feature image. The pixel values associated with the cells are interpolated based on the point attributes within the cells through inverse distance weighted (IDW) interpolation (Yu et al., 2014). The IDW interpolation method is proven to show an excellent rasterization quality by assigning different degrees of contributions to different points in a cell, which performs better than the strategies selecting the maximum point attribute or averaging the point attributes. To be specific, the empty cells with no data points are simply assigned with zero values. The feature image rasterization operations are

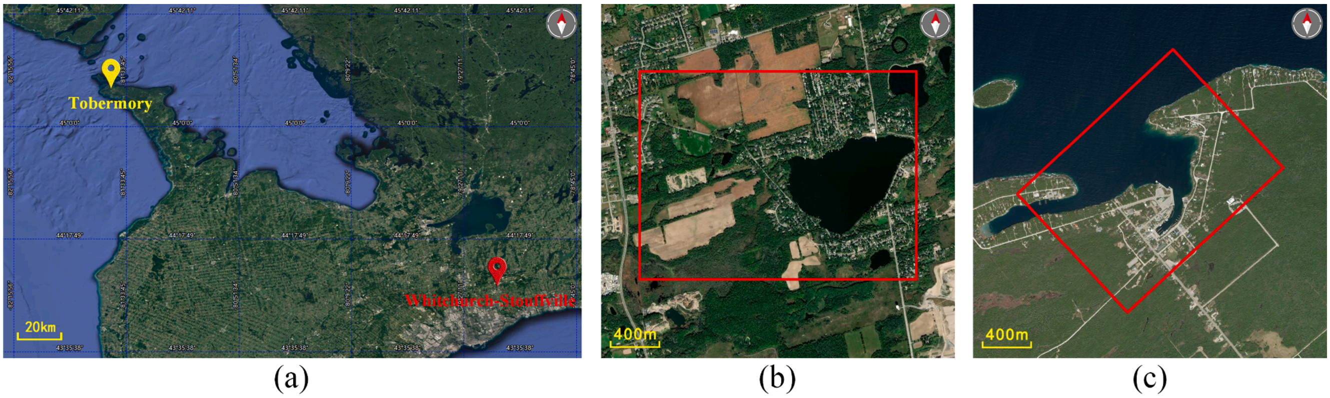


Fig. 1. (a) Overview of the study areas, (b) study area of Whitchurch-Stouffville, and (c) study area of Tobermory.

implemented using C++ on Microsoft Visual Studio 2019. As shown in Fig. 2, we obtain five kinds of feature images based on the data attributes with regard to elevation (Fig. 2(a)), number of returns (Fig. 2(b)), and the three channels of intensities (Fig. 2(c)-(e)). Fig. 2(f) also presents a false-color image synthesized by combining all the channels of intensity images. The red, green, and blue channels of the false-color image are composed of the intensity images from Channel 1, Channel 2, and Channel 3, respectively. Note that, except the three intensity images that are rasterized independently based on only the LiDAR points from their corresponding channels, the other feature images are all rasterized based on the merged point cloud by considering the entire information.

4. Methodology

By combining the feature semantic encoding superiority of capsule primitives and the long-range self-attention capability of transformer formulations, we design a cross-context capsule vision transformer (CapViT) architecture to serve land cover classification with MS-LiDAR feature images. As depicted by Fig. 3, the CapViT parallels three streams to investigate patch features under different contexts, which are finally integrated together to conduct category prediction. To be specific, given a query pixel in the feature images, three patches with different context details are retrieved and, respectively, fed into the corresponding stream to obtain capsule feature encodings. For each stream, the patch is first processed to form the capsule representations through a capsule embedding procedure, followed by a token embedding procedure to convert the capsule representations into the token representations. Then, a transformer encoder is connected to extract feature semantics. Eventually, the cross-context features are fused with an MLP to determine the category label of the query pixel.

4.1. Cross-context patch embedding

As preprocessing, the five kinds of feature images rasterized from the MS-LiDAR data are first aligned and stacked together to structurize into a multispectral image form, in which a pixel contains five-channel values provided by the corresponding feature images. This multispectral image representation is leveraged as the input to the CapViT to predict pixel-level category information. As shown in Fig. 3, for a query pixel, three patches with different sizes of $s_1 \times s_1$, $s_2 \times s_2$, and $s_3 \times s_3$ ($s_3 > s_2 > s_1$) are generated with the query pixel as the patch center. This set of patches can reflect the contextual properties of the query pixel at different scales, which perform better in distinctive feature exploitation than that of using a single fixed-size patch. Then, the three patches are taken as the input to the corresponding streams of the CapViT to extract capsule feature semantics.

As for each stream, the scalar intensity-valued patch is first converted into a vectorial capsule-encoded representation through a capsule embedding module. The resultant capsule representation has the identical size to the input patch. Noteworthily, different from traditional scalar primitives commonly used in CNN architectures, the capsule primitive employs a tensor formulation composed of a set of parameters (Sabour et al., 2017). The superior uniqueness about the capsule primitive is embodied in that both feature presence saliency and entity-aware intrinsic properties can be simultaneously encoded by using the capsule length and the parameters, respectively. The capsule embedding module is built by a convolutional layer with a kernel size of 3 × 3, a padding of 1, and a stride of 1. Specifically, the generated feature channels at each position are further partitioned sequentially into G sets, each of which encapsulates D components, thereby resulting in a D-dimensional capsule representation with G feature channels. In our architecture, D and G are set as 12 and 64 by default. Then, the squashing function (Sabour et al., 2017) is applied to the capsules for normalizing their lengths. The squashing function takes the following form:

$$\boldsymbol{C} = \frac{\|\boldsymbol{T}\|^2}{\|\boldsymbol{T}\|^2 + 1} \cdot \frac{\boldsymbol{T}}{\|\boldsymbol{T}\|} \tag{1}$$

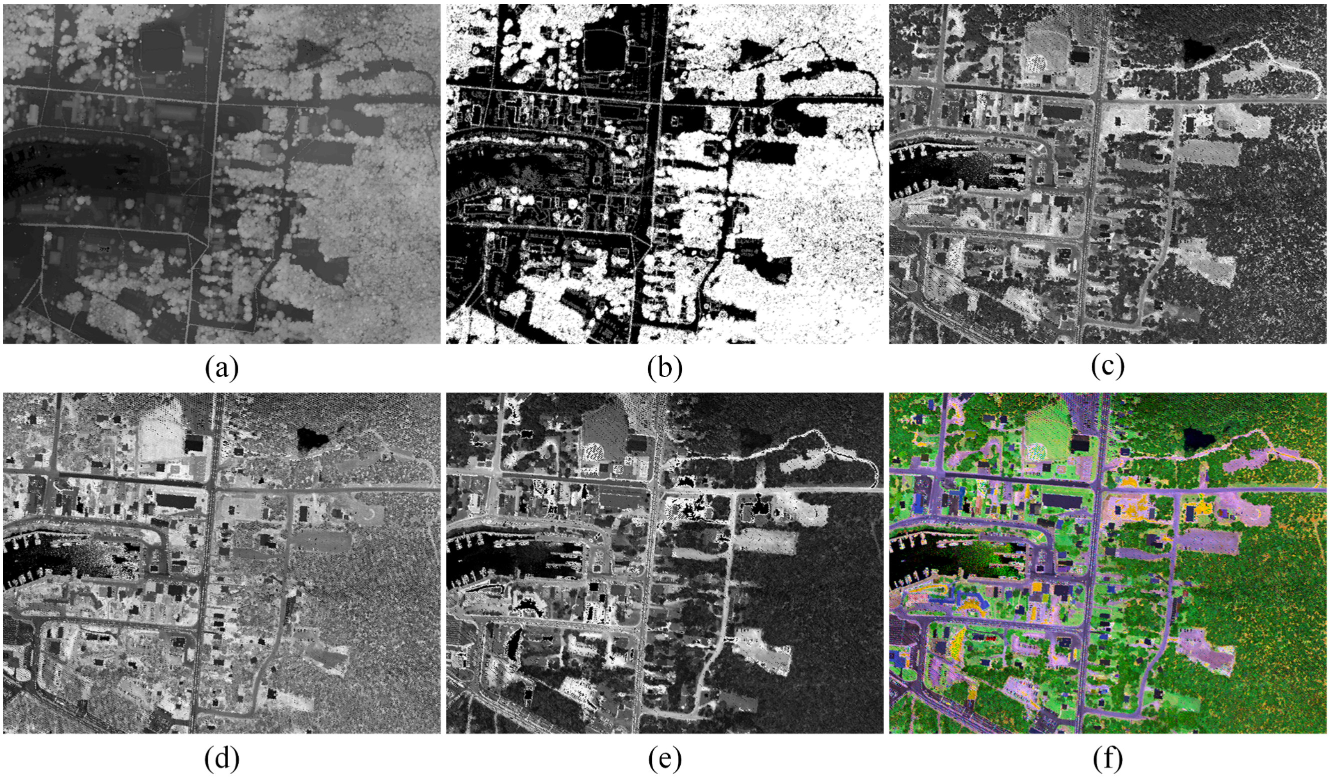


Fig. 2. Illustration of the feature images rasterized with (a) elevation, (b) number of returns, (c)-(e) three channels of intensities, and (f) the false-color image synthesized with all the channels of intensities.

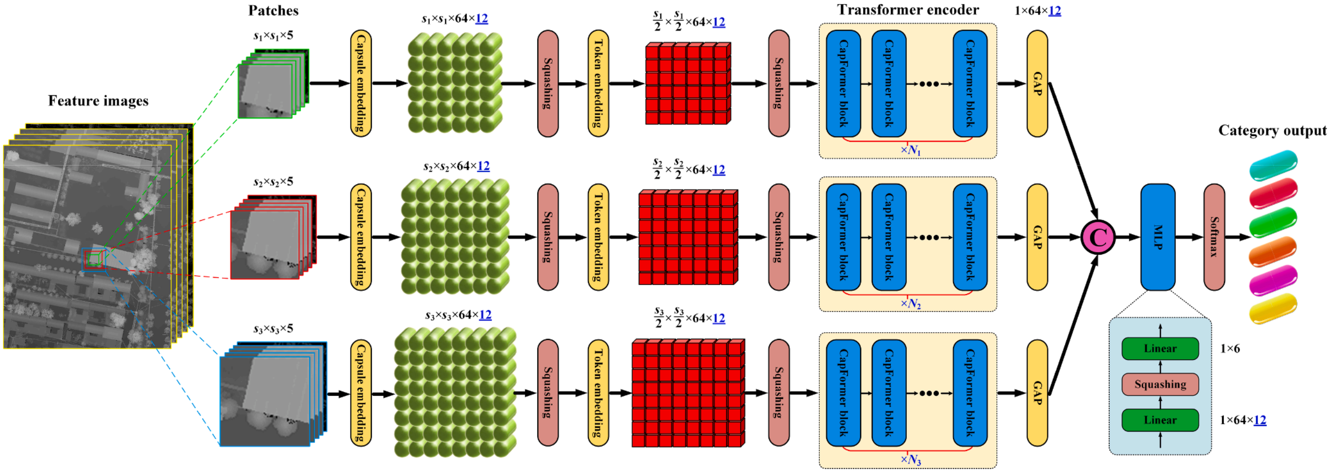

Feature images
Patches
$s_1 \times s_1 \times 5$
$s_2 \times s_2 \times 5$
$s_3 \times s_3 \times 5$
Capsule embedding
$s_1 \times s_1 \times 64 \times 12$
$s_2 \times s_2 \times 64 \times 12$
$s_3 \times s_3 \times 64 \times 12$
Squashing
Token embedding
$\frac{s_1}{2} \times \frac{s_1}{2} \times 64 \times 12$
$\frac{s_2}{2} \times \frac{s_2}{2} \times 64 \times 12$
$\frac{s_3}{2} \times \frac{s_3}{2} \times 64 \times 12$
Transformer encoder
CapFormer block
$\times N_1$
$\times N_2$
$\times N_3$
$1 \times 64 \times 12$
GAP
C
MLP
Softmax
Category output
Linear
Squashing
Linear
1×6
$1 \times 64 \times 12$

Fig. 3. Overview of the cross-context capsule vision transformer architecture.

where $\boldsymbol{T}$ denotes the raw capsule and $\boldsymbol{C}$ represents the normalized capsule.

Next, the capsule representation is processed by a token embedding module to generate a token representation, which will act as the operable semantic units for the subsequent feature encoding procedure. The token representation has a quarter size of the capsule representation. That is, both the width and height are reduced to the half sizes of the original ones after token embedding. It is equivalent to a feature downsampling manipulation by a sampling factor of 0.5. To be specific, the token embedding module is built by a capsule convolutional layer with a kernel size of 2 × 2 and a stride of 2, and maintains the same capsule dimension of D and the same number of feature channels of G. The capsule convolution (Sabour et al., 2017) is operated as follows:

$$\boldsymbol{T} = \sum_i a_i \mathbf{W}_i \boldsymbol{C}_i \tag{2}$$

where $\boldsymbol{C}_i$ is a capsule within the kernel, $\boldsymbol{T}$ is the resultant unnormalized capsule, $\mathbf{W}_i$ is a feature mapping matrix associated with capsule $\boldsymbol{C}_i$, and a_i denotes a coupling coefficient indicating the relevance of capsule $\boldsymbol{C}_i$, which is dynamically computed with the improved dynamic routing process (Rajasegaran et al., 2019). The improved dynamic routing process takes into consideration both the capsule lengths and orientations to iteratively determine the significance of a capsule. Compared with the original dynamic routing process (Sabour et al., 2017), it is more stable and easy to converge for constructing deep network architectures. Finally, a squashing function layer is appended to the token representation to conduct token normalization. As illustrated by Fig. 3, the token representation in each stream will be taken as the operable components to the transformer encoder for further feature semantic investigation.

4.2. Transformer encoder

The functionality of the transformer encoder is to exploit long-range global feature interactions of the input tokens with a self-attention mechanism. It is stacked by a set of capsule transformer (CapFormer) blocks and exports the overall feature representation of the input tokens. As detailed in Fig. 3, the transformer encoders of the three streams contain, respectively, N_1, N_2, and N_3 CapFormer blocks. In our architecture, we deploy the same configuration on the transformer encoders, which involve the same number of CapFormer blocks, i.e., $N_1 = 8$, $N_2 = 8$, and $N_3 = 8$ by default. Note that, the transformer encoder maintains the same capsule dimension of D, the same feature channel number of G and the same token number through all the CapFormer blocks. That is, the dimensions of the input tokens and the output representations of each CapFormer block are identical.

As illustrated by Fig. 4(a), the CapFormer block is constructed by two parallel multi-head self-attention (MSA) modules for global feature recalibrations and an MLP for local feature exploitation. Importantly, a LayerNorm layer is linked before the MLP and the MSA modules to perform layer normalization, and a residual connection is deployed after each of them to perform model augmentation. To be specific, given the input tokens, they are first normalized through the LayerNorm layer. Then, the normalized tokens are duplicated into two groups: one is directly fed into the token-wise MSA (T-MSA) for performing global feature self-attention from the token's perspective (i.e., performing spatial feature semantic attention) and one is transposed and fed into the channel-wise MSA (C-MSA) for performing global feature self-attention from the channel's perspective (i.e., performing channel feature

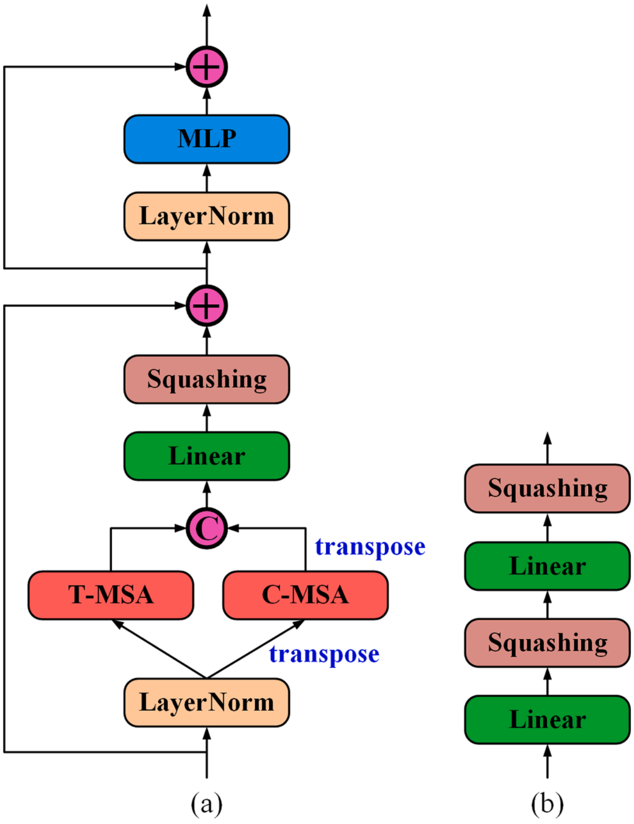


Fig. 4. Architectures of (a) the CapFormer block and (b) the MLP.

semantic attention). Here, the transpose operation functions to transform the tokens from the spatial domain into the channel domain. That is, given the input token representation $R^{t\times G\times D}$, where t, G, and D are, respectively, the token number, the feature channel number, and the capsule dimension, the input to the T-MSA module has the form of $R^{t\times G\times D}$ and the input to the C-MSA module has the form of $R^{G\times t\times D}$. Finally, the output from the C-MSA module is transposed and concatenated with the output from the T-MSA module, which are further fused through a linear layer, followed by a squashing nonlinearity. As illustrated by Fig. 4(b), the MLP consists of two linear layers and two squashing function layers. Specifically, a linear layer is followed by a squashing function layer. With such a dual-path self-attention design pattern of the CapFormer block, the spatial feature interactions and channel feature interactions can be simultaneously characterized in a global way, which is significantly beneficial to obtain high-quality, strongly-distinctive feature semantics.

As illustrated by Fig. 5(a), the T-MSA and C-MSA modules have the same architecture that follows a capsule-based MSA formulation. Take the T-MSA module for detailed description. For the input embedded tokens $R^{t\times G\times D}$, they are first preprocessed by three different sets of linear layers, each of which contains n separate parallel linear layers with different parameters, resulting in a query set Q={$\mathbf{Q}_1$, $\mathbf{Q}_2$, …, $\mathbf{Q}_n$}, where $\mathbf{Q}_i \in R^{t\times g}$, $i = 1, 2, \ldots, n$, a key set K={$\mathbf{K}_1$, $\mathbf{K}_2$, …, $\mathbf{K}_n$}, where $\mathbf{K}_i \in R^{t\times g}$, $i = 1, 2, \ldots, n$, and a value set V={$\mathbf{V}_1$, $\mathbf{V}_2$, …, $\mathbf{V}_n$}, where $\mathbf{V}_i \in R^{t\times g\times D}$, $i = 1, 2, \ldots, n$. Here, g denotes the number of feature channels. Then, each triad {$\mathbf{Q}_i$, $\mathbf{K}_i$, $\mathbf{V}_i$}, i = 1, 2, …, n, is dispatched to a single-head self-attention module to exploit global feature interactions. Finally, the outputs from all the n heads are sequentially concatenated and further fused through a linear layer, followed by a squashing nonlinearity. In our architecture, the number of heads is configured to be $n = 5$ and the number of feature channels is configured to be $g = 16$ by default.

The detailed architecture of the single-head self-attention module is illustrated in Fig. 5(b). Its mathematical formulation is as follows:

$$\text{Attention}(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) = Softmax\left(\frac{\boldsymbol{Q}_i\boldsymbol{K}_i^{\mathrm{T}}}{\sqrt{g}} + \boldsymbol{B}\right)\boldsymbol{V}_i \tag{3}$$

where $\mathbf{B} \in R^{t\times t}$ is a relative position bias between each pair of tokens (Liu et al., 2021), which is shared by all the n heads. To be specific, first, matrix multiplication is performed on $\mathbf{Q}_i$ and $\boldsymbol{K}_i^{\mathrm{T}}$, followed by a scaling operation on the product matrix by dividing $\sqrt{g}$. Then, the relative position bias is added to the product matrix to include position embeddings, followed by a softmax function. Here, we regard the resultant matrix as the position-embedded attention matrix. Finally, the position-embedded attention matrix is multiplied with $\mathbf{V}_i$ to generate the self-attention-recalibrated feature semantics as the output.

4.3. Prediction head

The predication head of the CapViT functions to gather the cross-context feature semantics from the three streams of transformer encoders to make a decision on the category label of the query pixel in the input. To this end, as illustrated by Fig. 3, the cross-context feature semantics from the three streams of transformer encoders are first downsampled with a global average pooling (GAP) layer, resulting in three capsule feature vectors representing the overall feature encodings under the corresponding contexts. Then, these three feature vectors are sequentially concatenated and further comprehensively interpreted by an MLP. Note that, the MLP comprises two linear layers that are linked by a squashing function layer. The second linear layer transforms the capsule representation into a scalar representation, whose nodes correspond to the land cover categories. Finally, the output of the MLP is adjusted with a softmax function for providing the category-oriented predictions with a “one-hot” encoding pattern. Here, the “one-hot” encoding pattern means that one entry in the output has the largest value and the other entries in the output have smaller values. To be specific, the entry with the largest value in the output determines the category of the input. The “one-hot” encoding pattern is formed by the supervision of the loss function at the training stage by comparing the predictions and the ground-truths and the adjustment of the softmax function applied to the output.

4.4. Loss function

In fact, there is a nonnegligible issue regarding the imbalance of the training samples from different classes in the land cover classification task, which might influence the capability of the constructed model. Hence, to well supervise the optimization of the CapViT, the loss func-

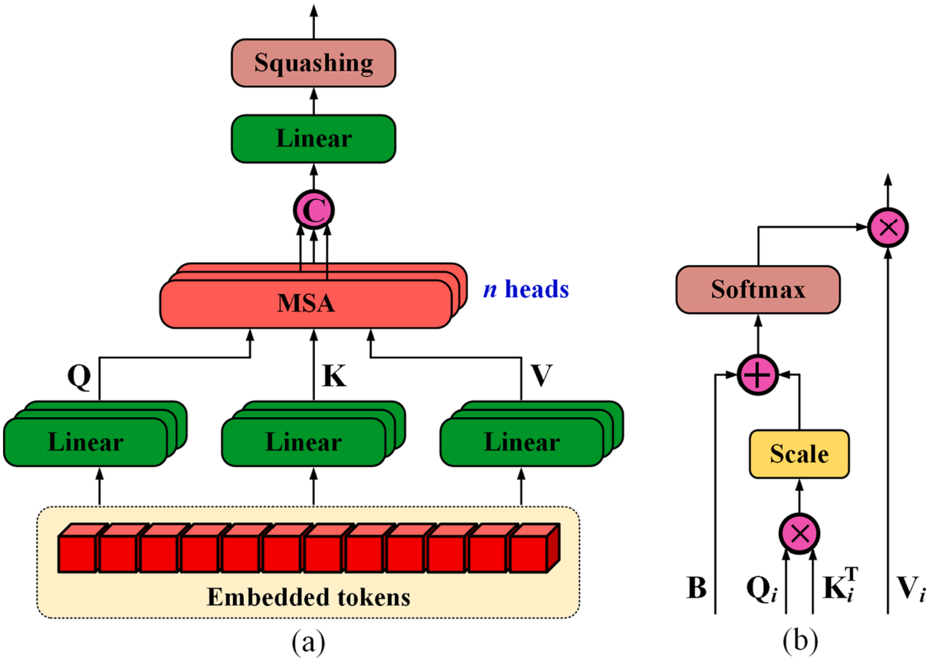


Fig. 5. Architectures of (a) the MSA module and (b) the single-head self-attention module.

tion is formulated as follows:

$$L = \sum_{i=1}^{P} L_{\mathrm{EFL}} \tag{4}$$

where P denotes the number of training samples; L_{EFL} is computed as the equalized focal loss (Li et al., 2022b) of the ground-truth prediction entry. The equalized focal loss can well alleviate the imbalance issue of training samples by using a class-related focusing factor to balance the contribution of each class to the total loss.

5. Results and discussions

5.1. Network training

At the network optimization stage, the CapViT was fine-tuned with the AdamW optimizer (Kingma and Ba, 2014) supervised by the loss function defined in Eq. (4) in a cloud computing environment. This platform is packed with a 128-GB memory, a 16-core CPU, and ten 16-GB GPUs. We trained the CapViT for 400 epochs with a batch size of 640, split on the ten GPUs. The weight decay of 0.05 and the initial learning rate of 0.001 were configured. The learning rate was scaled by a cosine decay learning rate scheduler. As in Yu et al. (2022), for each of the two test datasets, 60% of the annotated samples were randomly selected from each class to form the training set, where 3% of the samples were treated for validation. Such amount of training samples was sufficient to construct and optimize the CapViT. The remaining 40% data were left as the test set for model performance examination. Specifically, at either the training or the test stages, three patches with sizes of 24 × 24, 32 × 32, and 40 × 40 pixels containing different contextual details were cropped centered at a sample, which were, respectively, dispatched to the three streams of the CapViT.

5.2. Land cover classification

The classification performance of the CapViT was quantitatively evaluated with the following three metrics: overall accuracy (OA), average accuracy (AA), and Kappa coefficient (κ). These three metrics estimate the classification quality from different perspectives. Specifically, OA cares about the overall classification performance on all categories, while AA concerns the individual classification performance on each category. κ evaluates the model performance by taking into account both the cross-category and intra-category classification accuracies. Generally, the higher the values of the evaluation metrics, the better the classification accuracies of the model.

For each of the two test datasets of the surveyed areas, the land covers were annotated into six different types including water (T1), vegetation (T2), road (T3), soil (T4), building (T5), and other impervious surface (T6). Specifically, the land cover type of vegetation involved the trees and grasses. The ground-truths of the land covers were pixel-wisely labelled assisted by the high-resolution remote sensing images in the surveyed areas. The number of samples associated with each land cover type in the two test datasets was reported in detail in Table 1. The land cover classification results on the WS and TM datasets are reported in Tables 2 and 3, respectively, where the classification accuracy of each land cover type, as well as the overall classification accuracies measured by the OA, AA, and κ metrics, are recorded in detail. The visual exhibitions of the land cover classification results on the two datasets are also presented in Fig. 6, where different land cover types are marked with different colors. For providing clear visual inspections, the close-up views of two regions from the two datasets are also illustrated in Fig. 6.

As indicated in Table 2, the six types of land covers in the WS dataset were excellently distinguished from each other with a quite promising classification accuracy on each land cover type. Specifically, the best classification performance appeared on the land cover water with a single-class classification accuracy of 99.57%. In contrast, compared with the classification accuracies of the other land cover types, relatively lower classification performances fell on the land covers soil and building with single-class classification accuracies of 94.16% and 94.02%, respectively. Quantitatively, the classification accuracy difference between land covers water and building was about 5.55%. Moreover, similar classification performances were attained in identifying land covers other impervious surface and vegetation. All in all, for the WS dataset, the CapViT performed satisfactorily with high overall classification accuracies of 98.95%, 95.93%, and 0.9834 with regard to the OA, AA, and κ metrics, respectively. As reflected in Table 3, the CapViT behaved better on the TM dataset with higher classification accuracies on all the land cover types compared with those of the WS dataset. Similarly, among the six land cover types, water regions were successfully located with the highest single-class classification accuracy of 99.64%. The land cover vegetation was also effectively recognized and separated from the other land covers with a single-class classification accuracy of 96.85%. For the land covers road and soil, equally matched results were achieved in identifying them with single-class classification accuracies of 95.26% and 95.03%, respectively. Comparatively, less promising classification results also appeared on the land cover building with a single-class classification accuracy of 94.61%. To be specific, the classification accuracy of land cover building was degraded by about 5.03% in comparison with that of the land cover water. On the whole, competitive overall classification accuracies of 99.42%, 96.30%, and 0.9883 with regard to the OA, AA, and κ metrics, respectively, were also obtained by the CapViT on the TM dataset.

The classification results on the two datasets indicated that the land cover types with small elevation fluctuations, low topology complexities, and homogeneous reflectivity properties can be effectively recognized and differentiated with high classification accuracies. For instance, the water bodies and road regions showed quite homogeneous textural attributes in the feature images either with respect to the intensities or the elevations. Thus, they were easy to tell apart from the other land covers due to their feature uniqueness. On the contrary, the classification performance was degraded on the land cover types with more complex spatial structures and varying spectral characteristics. For instance, the buildings in the surveyed areas exhibited different color appearances, diverse geometric structures, and various heights. As a result, the misclassification error was increased in handling the building regions due to their feature heterogeneities in the feature images. However, on both datasets, the classification accuracy of land cover building was still acceptable and promising. In addition, some classification errors were generated in the adjacent regions of different land cover types. For example, a part of soil elements was wrongly classified as the road type. Nevertheless, the misclassification rate was quite low on every land cover type in both of the two datasets. It convincingly demonstrated the high performance of the CapViT on land cover classification with MS-LiDAR data. The classification performance gains of the CapViT mainly benefitted from the following three aspects of architecture designs. First, adopted with capsule primitives, more powerful and distinctive entity-aware feature semantics can be

Table 1
Number of samples associated with each land cover type in the two test datasets.

Dataset	T1	T2	T3	T4	T5	T6
WS	1,935,836	6,315,571	641,277	3,534,682	351,372	92,162
TM	3,164,296	2,922,096	213, 212	362,264	151,044	1,145,232

Table 2
Land cover classification results on the WS dataset.

Type	CapViT	ESA-CapsNet	HCapsNet	ViT	CNN	DBM	SRF	SVM	MLC	RF
T1 (%)	**99.57**	**99.35**	**99.11**	**99.27**	**98.44**	**98.10**	**94.64**	**95.17**	**94.76**	**92.77**
T2 (%)	96.18	95.27	94.53	95.06	93.17	92.42	87.63	88.54	87.99	86.61
T3 (%)	95.53	94.56	93.72	94.22	91.86	83.63	81.87	83.36	82.92	82.53
T4 (%)	94.16	93.31	92.91	93.17	91.43	88.71	86.12	87.02	85.84	84.97
T5 (%)	94.02	93.17	92.23	92.95	91.15	88.95	86.35	88.79	88.63	82.38
T6 (%)	96.14	95.24	94.76	95.11	93.91	93.54	89.14	91.65	90.25	81.22
OA (%)	**98.95**	98.42	97.89	98.27	95.91	94.36	90.52	92.17	91.23	90.64
AA (%)	**95.93**	95.15	94.54	94.96	93.33	90.89	87.63	89.09	88.40	85.08
$\kappa \times 100$	**98.34**	97.76	97.13	97.51	95.34	93.78	88.96	91.75	89.67	84.96

Table 3
Land cover classification results on the TM dataset.

Type	CapViT	ESA-CapsNet	HCapsNet	ViT	CNN	DBM	SRF	SVM	MLC	RF
T1 (%)	**99.64**	**99.52**	**99.34**	**99.47**	**98.76**	**98.35**	**95.28**	**96.38**	**95.34**	**93.42**
T2 (%)	96.85	96.13	95.25	95.84	94.45	93.51	88.94	89.77	88.92	86.88
T3 (%)	95.26	94.82	94.17	94.68	92.56	88.26	82.73	85.34	83.17	83.35
T4 (%)	95.03	94.47	93.52	94.21	91.77	90.13	87.86	89.12	88.05	85.24
T5 (%)	94.61	93.55	92.41	93.09	91.51	89.92	88.07	88.81	88.21	82.57
T6 (%)	96.42	95.89	95.02	95.52	94.22	93.81	91.82	92.43	91.74	82.26
OA (%)	**99.42**	98.91	98.34	98.75	96.68	95.13	91.97	93.22	92.15	90.96
AA (%)	**96.30**	95.73	94.95	95.47	93.88	92.33	89.12	90.31	89.24	85.62
$\kappa \times 100$	**98.83**	98.37	97.76	98.13	95.85	94.24	91.18	92.25	91.27	85.31

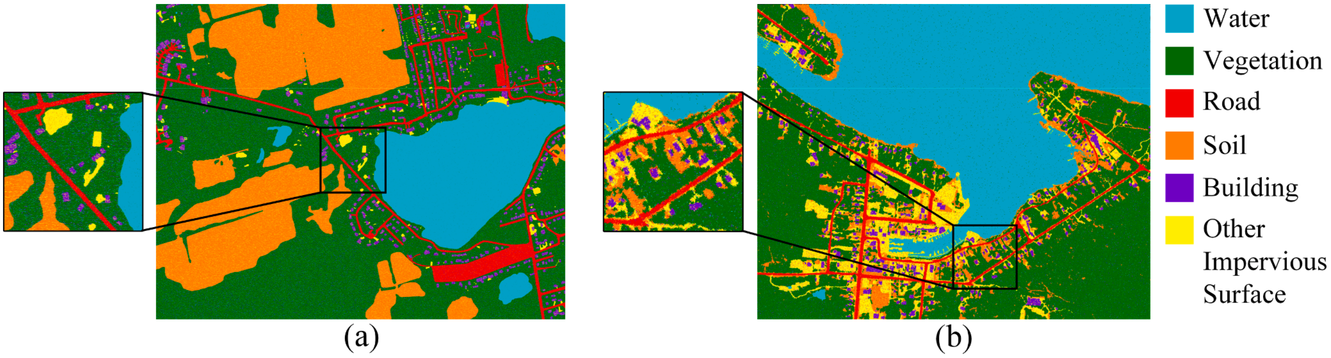


Fig. 6. Land cover classification results on (a) the WS dataset and (b) the TM dataset.

extracted. Second, formulated with a cross-context transformer architecture, multiscale context properties can be effectively exploited to provide high-quality feature evidence for prediction. Third, designed with dual-path MSA modules, the spatial and channel interactions of tokens can be simultaneously analyzed to obtain strong global feature encodings. Note that, the processing efficiency of the CapViT was slightly lower than that of using a pure CNN-based architecture due to the dynamic routing process used in capsule convolutions. Nevertheless, the efficiency degradation was not significant.

5.3. Comparative analyses

As for verification experiments to further convincingly examine the practical feasibility and competitive superiority of the CapViT in MS-LiDAR-based land cover classification missions, we carried out intensive comparisons and analyses with some state-of-the-art land cover classification models, which were based on MS-LiDAR feature images, including ESA-CapsNet (Yu et al., 2022), HCapsNet (Yu et al., 2020), CNN (Pan et al., 2020), DBM (Pan et al., 2019), SRF (Ghaseminik et al., 2021), SVM (Huo et al., 2018), MLC (Morsy et al., 2017), and RF (Matikainen et al., 2017). Besides, the ViT model (Dosovitskiy et al., 2021) was also included as a baseline for performance comparison. To be specific, the ESA-CapsNet and HCapsNet were constructed with capsule primitives and followed the convolutional architectures. The CNN and DBM employed the traditional deep learning design principles with scalar neuron primitives. The SRF, SVM, MLC, and RF relied on machine learning techniques to train classifiers with handcrafted features. Note that, feature attention mechanism was considered in the ESA-CapsNet to promote the representation quality of the output feature semantics. Local and global feature semantics were reasonably combined in the HCapsNet for investigating different contexts of details. For fair comparisons, the same evaluation metrics of OA, AA, and κ were leveraged for providing quantitative classification performance assessments on these models. The detailed land cover classification results obtained by these models are reported in Tables 2 and 3.

As reflected by the statistical results in Tables 2 and 3, the ESA-CapsNet, ViT, and HCapsNet behaved more superiorly than the other models with higher overall classification accuracies on both of the two datasets. Specifically, the ESA-CapsNet achieved the best performance among the nine models. Likewise, for all these models, a better performance also appeared on the TM dataset. It means that the scene condition of the TM dataset was less complicated than that of the WS dataset, thereby achieving more accurate land cover predictions by these models. In contrast, the SRF, MLC, and RF introduced more misclassification errors almost on each land cover type caused by either omissions or commissions. As a result, the single-class classification accuracy on each of the datasets was degraded apparently, thereby leading to the decline of the overall classification accuracies. Note that, the classification accuracy differences between the ESA-CapsNet and RF were about 0.1280 and 0.1306, respectively, with regard to the κ metric on the two datasets. Furthermore, the CNN and DBM performed equally matched with moderate classification accuracies among the nine models. The performance superiorities of the ESA-CapsNet and HCapsNet benefitted from the capsule feature encoding philosophy and the self-attention or multi-context feature embedding strategies for semantic-strong feature abstraction. The performance gains on the ViT well proved the competitive capability of the long-range global feature exploitation scheme of the transformer architectures. On the other hand, the classification performances of the SRF, MLC, and RF were greatly impeded owing to the use of handcrafted low-level features. Noteworthily, the SRF leveraged a segment-based processing pipeline based on the pre-segmentation of semantic regions. Thus, the segmentation quality affected significantly on the final land cover prediction results.

However, through comparative analyses on the land cover classification results recorded in Tables 2 and 3, we convinced that the proposed cross-context CapViT demonstrated significant improvements over the other compared models with either higher single-class classification accuracies or overall classification accuracies. For instance, the CapViT improved by about 0.78% and 0.57% with regard to the AA metric in comparison with the ESA-CapNet and by about 0.1338 and 0.1352 with regard to the κ metric compared with the RF on the two datasets. In conclusion, the CapViT worked suitably and competitively in the land cover classification task.

6. Conclusion

In order to improve the map-level land cover mapping accuracy, this paper has formulated a cross-context vision transformer architecture stacked by capsule primitives, termed as CapViT, for conducting land cover classification with MS-LiDAR data. The CapViT parallels three streams of transformer encoders functioning to exploit long-range global feature interactions under different context details, which are finally effectively combined to provide high-quality and strong feature semantics for accurate land cover prediction. Specifically, the CapFormer block constituting the transformer encoder involves dual-path capsule-based MSA modules serving for interpreting both the spatial feature correlations and channel feature interdependencies of the token representations, which significantly promotes the feature encoding semantics to a large extent. The proposed CapViT has been elaborately testified on two MS-LiDAR datasets towards land cover classification. Quantitative assessments demonstrated that the CapViT performed excellently with high single-class and overall classification accuracies on the two datasets. The OA, AA, and κ values are 98.95%, 95.93%, and 0.9834 on the WS dataset, and 99.42%, 96.30%, and 0.9883 on the TM dataset. Intensive comparative analyses also convinced the practical feasibility and competitive superiority of the CapViT in handling MS-LiDAR-based land cover mapping missions. Thus, the work in this paper provides an effective and high-performance solution to the improvement of the map-level land cover classification accuracies and positively enlarges the application domains of the MS-LiDAR data. In the future, we will investigate pretraining techniques and more advanced network architectures to further promote the land cover classification performance.

Funding

This work was supported by the National Natural Science Foundation of China [grant numbers 62076107, 51975239, 41971414]; the Natural Science Foundation of Jiangsu Province [grant numbers BK20211365, BK20191214]; and the Six Talent Peaks Project in Jiangsu Province [grant number XYDXX-098].

CRediT authorship contribution statement

Yongtao Yu: Conceptualization, Funding acquisition, Methodology, Writing – original draft. **Tao Jiang:** Conceptualization, Methodology, Writing – original draft. **Junyong Gao:** Data curation, Methodology, Writing – original draft. **Haiyan Guan:** Funding acquisition, Formal analysis, Writing – review & editing. **Dilong Li:** Formal analysis, Supervision. **Shangbing Gao:** Investigation, Validation. **E Tang:** Investigation, Validation. **Wenhao Wang:** Software, Visualization. **Peng Tang:** Software, Visualization. **Jonathan Li:** Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Chen, X., Qiu, C., Guo, W., Yu, A., Tong, X., Schmitt, M., 2022. Multiscale feature learning by transformer for building extraction from satellite images. IEEE Geosci. Remote Sens. Lett. 19, 1–5.

Chen, X., Ye, C., Li, J., Chapman, M.A., 2018. Quantifying the carbon storage in urban trees using multispectral ALS data. IEEE J. Sel. Topic Appl. Earth Observ. Remote Sens. 11 (9), 3358–3365.

Dai, W., Yang, B., Dong, Z., Shaker, A., 2018. A new method for 3D individual tree extraction using multispectral airborne LiDAR point clouds. ISPRS J. Photogramm. Remote Sens. 144, 400–411.

Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L., Chen, D., Guo, B., 2022. CSwin transformer: A general vision transformer backbone with cross-shaped windows. arXiv:2107.00652v3. [Online]. Available: https://arxiv.org/abs/2107.00652v3.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., 2021. An image is worth 16×16 words: Transformers for image recognition at scale. In: Proc. Int. Conf. Learn. Rep., Vienna, Austria, pp. 1-22.

Ekhtari, N., Glennie, C., Fernandez-Diaz, J.C., 2018. Classification of airborne multispectral LiDAR point clouds for land cover mapping. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 11 (6), 2068–2078.

Fang, J., Yang, C., Shi, Y., Wang, N., Zhao, Y., 2022. External attention based TransUNet and label expansion strategy for crack detection. IEEE Trans. Intell. Transp. Syst., early access, 10.1109/TITS.2022.3154407.

Ghaseminik, F., Aghamohammadi, H., Azadbakht, M., 2021. Land cover mapping of urban environments using multispectral LiDAR data under data imbalance. Remote Sens. App. Soci. Environ. 21, 100449.

Guo, J., Han, K., Wu, H., Xu, C., Tang, Y., Xu, C., Wang, Y., 2021. CMT: Convolutional neural networks meet vision transformers. arXiv:2107.06263v2. [Online]. Available: https://arxiv.org/abs/2107.06263v2.

Hänsch, R., Hellwich, O., 2021. Fusion of multispectral LiDAR, hyperspectral, and RGB data for urban land cover classification. IEEE Geosci. Remote Sens. Lett. 18 (2), 366–370.

Hartzell, P., Glennie, C., Biber, K., Khan, S., 2014. Application of multispectral LiDAR to automated virtual outcrop geology. ISPRS J. Photogramm. Remote Sens. 88, 147–155.

Hong, D., Chanussot, J., Yokoya, N., Kang, J., Zhu, X.X., 2020. Learning-shared cross-modality representation using multispectral-LiDAR and hyperspectral data. IEEE Geosci. Remote Sens. Lett. 17 (8), 1470–1474.

Huo, L.Z., Silva, C.A., Klauberg, C., Mohan, M., Zhao, L.J., Tang, P., Hudak, A.T., 2018. Supervised spatial classification of multispectral LiDAR data in urban areas. PLoS One 13(10), e0206185.

Jin, H., Mountrakis, G., 2022. Fusion of optical, radar and waveform LiDAR observations for land cover classification. ISPRS J. Photogramm. Remote Sens. 187, 171–190.

Jing, Z., Guan, H., Zhao, P., Li, D., Yu, Y., Zang, Y., Wang, H., Li, J., 2021. Multispectral LiDAR point cloud classification using SE-PointNet++. Remote Sens. 13 (13), 2516.

Karila, K., Matikainen, L., Puttonen, E., Hyyppä, J., 2017. Feasibility of multispectral airborne laser scanning data for road mapping. IEEE Geosci. Remote Sens. Lett. 14 (3), 294–298.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv: 1412.6980v9. [Online]. Available: https://arxiv.org/abs/1412.6980v9.

Kukkonen, M., Maltamo, M., Korhonen, L., Packalen, P., 2019. Multispectral airborne LiDAR data in the prediction of boreal tree species composition. IEEE Trans. Geosci. Remote Sens. 57 (6), 3462–3471.

Lee, Y., Kim, J., Willette, J., Hwang, S.J., 2021. MPViT: Multi-path vision transformer for dense prediction. arXiv:2112.11010v2. [Online]. Available: https://arxiv.org/abs/2112.11010v2.

Li, D., Shen, X., Guan, H., Yu, Y., Wang, H., Zhang, G., Li, J., Li, D., 2022a. AGFP-Net: Attentive geometric feature pyramid network for land cover classification using airborne multispectral LiDAR data. Int. J. Appl. Earth Observ. Geoinform. 108, 102723.

Li, D., Shen, X., Yu, Y., Guan, H., Li, J., Zhang, G., Li, D., 2020. Building extraction from airborne multi-spectral LiDAR point clouds based on graph geometric moments convolutional neural networks. Remote Sens. 12(19), 3186.

Li, B., Yao, Y., Tan, J., Zhang, G., Yu, F., Lu, J., Luo, Y., 2022b. Equalized focal loss for dense long-tailed object detection. arXiv:2201.02593. [Online]. Available: https://arxiv.org/abs/2201.02593.

Lindberg, E., Holmgren, J., Olsson, H., 2021. Classification of tree species classes in a hemi-boreal forest from multispectral airborne laser scanning data using a mini raster cell method. Int. J. Appl. Earth Observ. Geoinform. 100, 102334.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin transformer: Hierarchical vision transformer using shifted windows. arXiv: 2103.14030v2. [Online]. Available: https://arxiv.org/abs/2103.14030v2.

Liu, Y., Zhang, Y., Wang, Y., Hou, F., Yuan, J., Tian, J., Zhang, Y., Shi, Z., Fan, J., He, Z., 2022. A survey of visual transformers. arXiv:2111.06091v3. [Online]. Available: https://arxiv.org/abs/2111.06091v3.

Luo, B., Yang, J., Song, S., Shi, S., Gong, W., Wang, A., Du, L., 2022. Target classification of similar spatial characteristics in complex urban areas by using multispectral LiDAR. Remote Sens. 14 (1), 238.

Lv, P., Wu, W., Zhong, Y., Du, F., Zhang, L., 2022. SCViT: A spatial-channel feature preserving vision transformer for remote sensing image scene classification. IEEE Trans. Geosci. Remote Sensing 60, 1–12.

Ma, L., Li, Y., Li, J., Wang, C., Wang, R., Chapman, M.A., 2018. Mobile laser scanned point-clouds for road object detection and extraction: A review. Remote Sens. 10 (10), 1531.

Matikainen, L., Karila, K., Hyyppä, J., Litkey, P., Puttonen, E., Ahokas, E., 2017. Object-based analysis of multispectral airborne laser scanner data for land cover classification and map updating. ISPRS J. Photogramm. Remote Sens. 128, 298–313.

Matikainen, L., Karila, K., Litkey, P., Ahokas, E., Hyyppä, J., 2020. Combining single photon and multispectral airborne laser scanning for land cover classification. ISPRS J. Photogramm. Remote Sens. 164, 200–216.

Mirzaei, K., Arashpour, M., Asadi, E., Masoumi, H., Bai, Y., Behnood, A., 2022. 3D point cloud data processing with machine learning for construction and infrastructure applications: A comprehensive review. Adv. Eng. Inform. 51, 101501.

Morsy, S., Shaker, A., El-Rabbany, A., 2017. Multispectral LiDAR data for land cover classification of urban areas. Sens. 17 (5), 958.

Pan, S., Guan, H., Chen, Y., Yu, Y., Gonçalves, W.N., Junior, J.M., Li, J., 2020. Land-cover classification of multispectral LiDAR data using CNN with optimized hyper-parameters. ISPRS J. Photogramm. Remote Sens. 166, 241–254.

Pan, S., Guan, H., Yu, Y., Li, J., Peng, D., 2019. A comparative land-cover classification feature study of learning algorithms: DBM, PCA, and RF using multispectral LiDAR data. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 12 (4), 1314–1326.

Peng, Z., Huang, W., Gu, S., Xie, L., Wang, Y., Jiao, J., Ye, Q., 2021. Conformer: Local features coupling global representations for visual recognition. arXiv:2105.03889. [Online]. Available: https://arxiv.org/abs/2105.03889.

Rajasegaran, J., Jayasundara, V., Jayasekara, S., Jayasekara, H., Seneviratne, S., Rodrigo, R., 2019. DeepCaps: Going deeper with capsule networks. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Long Beach, USA, pp. 10725-10733.

Sabour, S., Frosst, N., Hinton, G.E., 2017. Dynamic routing between capsules. In: Proc. Conf. Neural Inform. Process. Syst., Long Beach, USA, pp. 1-11.

Sankey, J.B., Sankey, T.T., Li, J., Ravi, S., Wang, G., Caster, J., Kasprak, A., 2021. Quantifying plant-soil-nutrient dynamics in rangelands: Fusion of UAV hyperspectral-LiDAR, UAV multispectral-photogrammetry, and ground-based LiDAR-digital photography in a shrub-encroached desert grassland. Remote Sens. Environ. 253, 112223.

Shaker, A., Yan, W.Y., LaRocque, P.E., 2019. Automatic land-water classification using multispectral airborne LiDAR data for near-shore and river environments. ISPRS J. Photogramm. Remote Sens. 152, 94–108.

Shao, H., Chen, Y., Yang, Z., Jiang, C., Li, W., Wu, H., Wang, S., Yang, F., Chen, J., Puttonen, E., Hyyppä, J., 2020. Feasibility study on hyperspectral LiDAR for ancient Huizhou-style architecture preservation. Remote Sens. 12 (1), 88.

Shi, S., Bi, S., Gong, W., Chen, B., Chen, B., Tang, X., Qu, F., Song, S., 2021. Land cover classification with multispectral LiDAR based on multi-scale spatial and spectral feature selection. Remote Sens. 13 (20), 4118.

Sun, J., Shi, S., Yang, J., Gong, W., Qiu, F., Wang, L., Du, L., Chen, B., 2019. Wavelength selection of the multispectral LiDAR system for estimating leaf chlorophyll and water contents through the PROSPECT model. Agric. Forest Meteorol. 266–267, 43–52.

Sun, L.e., Zhao, G., Zheng, Y., Wu, Z., 2022. Spectral-spatial feature tokenization transformer for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 60, 1–14.

Wang, Q., Gu, Y., 2020. A discriminative tensor representation model for feature extraction and classification of multispectral LiDAR data. IEEE Trans. Geosci. Remote Sens. 58 (3), 1568–1586.

Wang, L., Li, R., Duan, C., Zhang, C.e., Meng, X., Fang, S., 2022. A novel transformer based semantic segmentation scheme for fine-resolution remote sensing images. IEEE Geosci. Remote Sens. Lett. 19, 1–5.

Wang, W., Xie, E., Li, X., Fan, D., Song, K., Liang, D., Lu, T., Luo, P., Shao, L., 2021a. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. arXiv:2102.12122v2. [Online]. Available: https://arxiv.org/abs/2102.12122v2.

Wang, W., Yao, L., Chen, L., Lin, B., Cai, D., He, X., Liu, W., 2021b. CrossFormer: A versatile vision transformer hinging on cross-scale attention. arXiv:2108.00154v2. [Online]. Available: https://arxiv.org/abs/2108.00154v2.

Yan, W.Y., Shaker, A., El-Ashmawy, N., 2015. Urban land cover classification using airborne LiDAR data: A review. Remote Sens. Environ. 158, 295–310.

Yang, J., Li, C., Zhang, P., Dai, X., Xiao, B., Yuan, L., Gao, J., 2021. Focal self-attention for local-global interactions in vision transformers. arXiv:2107.00641. [Online]. Available: https://arxiv.org/abs/2107.00641.

Yu, Y., Guan, H., Li, D., Gu, T., Wang, L., Ma, L., Li, J., 2020. A hybrid capsule network for land cover classification using multispectral LiDAR data. IEEE Geosci. Remote Sens. Lett. 17 (7), 1263–1267.

Yu, Y., Li, J., Guan, H., Wang, C., Yu, J., 2014. Automated detection of road manhole and sewer well covers from mobile LiDAR point clouds. IEEE Geosci. Remote Sens. Lett. 11 (9), 1549–1553.

Yu, Y., Liu, C., Guan, H., Wang, L., Gao, S., Zhang, H., Zhang, Y., Li, J., 2022. Land cover classification of multispectral LiDAR data with an efficient self-attention capsule network. IEEE Geosci. Remote Sens. Lett. 19, 1–5.

Zhao, P., Guan, H., Li, D., Yu, Y., Wang, H., Gao, K., Junior, J.M., Li, J., 2021. Airborne multispectral LiDAR point cloud classification with a feature reasoning-based graph convolution network. Int. J. Appl. Earth Observ. Geoinform. 105, 102634.