Land Cover Classification of Multispectral LiDAR Data With an Efficient Self-Attention Capsule Network

Yongtao Yu, *Senior Member, IEEE*, Chao Liu, Haiyan Guan, *Senior Member, IEEE*, Lanfang Wang, Shangbing Gao, Haiyan Zhang, Yahong Zhang, and Jonathan Li, *Senior Member, IEEE*

***Abstract*— Periodically conducting land cover mapping plays a vital role in monitoring the status and changes of the land use. The up-to-date and accurate land use database serves importantly for a wide range of applications. This letter constructs an efficient self-attention capsule network (ESA-CapsNet) for land cover classification of multispectral light detection and ranging (LiDAR) data. First, formulated with a novel capsule encoder–decoder architecture, the ESA-CapsNet performs promisingly in extracting high-level, informative, and strong feature semantics for pixel-wise land cover classification by using the five types of rasterized feature images. Furthermore, designed with a novel capsule-based attention module, the channel and spatial feature encodings are comprehensively exploited to boost the feature saliency and robustness. The ESA-CapsNet is evaluated on two multispectral LiDAR data sets and achieves an advantageous performance with the overall accuracy, average accuracy, and kappa coefficient of over 98.42%, 95.15%, and 0.9776, respectively. Comparative experiments with the existing methods also demonstrate the effectiveness and applicability of the ESA-CapsNet in land cover classification tasks.**

***Index Terms*— Capsule feature attention, capsule network, land cover classification, land use mapping, multispectral light detection and ranging (LiDAR).**

I. Introduction

With the continuous urban sprawl, the frequent rural land planning, and the massive human activities, the status of land use is always keeping changing. Comprehensively and precisely mastering the current status of land use in a local region or the whole country is greatly important to promote the integrated management of the urban and rural cadastre, conduct the evaluation and analysis of the land use and management, and facilitate the macroeconomic regulation and control of the land. Moreover, the up-to-date and accurate land use database also serves for a variety of environmental, agricultural, and social applications [1]. Thus, periodically carrying out land cover mapping can assist in the rapid updating of the land use database. A traditional way for land cover information collection is manually performed based on field surveys, which, however, cost a great amount of labor and time expenditures. In recent decades, the advances of remote sensing techniques have provided a promising solution to land cover mapping tasks. The collection of varying-grained and different-range remote sensing data can be efficiently accomplished by using imaging sensors or light detection and ranging (LiDAR) sensors. Generally, remote sensing images captured by imaging sensors have rich spectral information, whereas point clouds collected by LiDAR sensors retain actual 3-D properties. Both the spectral and geometrical information behave significantly for enhancing the land cover mapping accuracy. Fortunately, recent development of multispectral LiDAR systems, which can collect multichannel LiDAR data covering different spectra simultaneously, has broken new ground for land cover mapping tasks due to their superior advantages of providing both abundant spectral and geometrical features.

Existing approaches for land cover mapping of multispectral LiDAR data generally adopt two processing strategies: image-based strategy and point cloud-based strategy. The image-based strategy converts the 3-D multispectral LiDAR data into a set of feature images according to the multichannel geometrical and spectral properties. In contrast, the point cloud-based strategy directly operates the multichannel 3-D LiDAR point clouds. Comparatively, the image-based strategy can achieve high efficiency in processing large scenes, while the point cloud-based strategy can well maintain the geometrical properties of land covers. Ekhtari *et al.* [2] designed a two-step method to, respectively, handle the single-return and multireturn LiDAR points. The single-return points were classified based on the multichannel intensity and elevation information, and the multireturn points were categorized using a rule-based (RB) method. Morsy *et al.* [3] tested two techniques to classify multispectral LiDAR data: image-based classification and point-based classification. The former trained a maximum likelihood (ML) classifier with the input of the

Manuscript received February 24, 2021; revised March 25, 2021; accepted April 1, 2021. Date of publication April 16, 2021; date of current version December 30, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 62076107, Grant 51975239, and Grant 41971414; in part by the Six Talent Peaks Project in Jiangsu Province under Grant XYDXX-098; and in part by the National Key Research and Development Program of China under Grant 2018YFB1004904. *(Corresponding author: Yongtao Yu.)*

Yongtao Yu, Chao Liu, Lanfang Wang, Shangbing Gao, Haiyan Zhang, and Yahong Zhang are with the Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian 223003, China (e-mail: allennessy@hyit.edu.cn).

Haiyan Guan is with the School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: guanhy.nj@nuist.edu.cn).

Jonathan Li is with the Department of Geography and Environmental Management, University of Waterloo, Waterloo N2L 3G1, Canada (e-mail: junli@uwaterloo.ca).

Digital Object Identifier 10.1109/LGRS.2021.3071252

intensity and elevation images, whereas the latter was based on ground filtering and normalized difference vegetation indices calculation. Matikainen *et al.* [4] proposed an object-based random forest analysis method for land cover classification. First, homogeneous regions were segmented for computing features. Then, the random forest classifier and histogram analysis were applied for land cover type determination. Likewise, Karila *et al.* [5] leveraged multispectral LiDAR data for road mapping by using object-based random forest analysis. Dai *et al.* [6] investigated the application of tree delineation by using multispectral LiDAR data. In their work, mean shift and support vector machine (SVM) were used to, respectively, segment tree crowns and classify undersegmentations. Differently, Naveed *et al.* [7] presented an improved multiscale treetop detection method, cooperated with a region-based segmentation approach, to extract individual tree crowns. Via linear discriminant analysis, Kukkonen *et al.* [8] explored the feasibility of multispectral LiDAR data to predict tree species. Wang and Gu [9] constructed a discriminative tensor representation model to characterize the spatial, spectral, and geometrical features of multispectral LiDAR points. The classification was finalized using an SVM classifier. In addition, multispectral LiDAR data were also considered for virtual outcrop geology [10], land–water classification [11], and surface fuel load estimation [12].

Due to the advanced characteristics of abstracting multilevel and multigrained features in an end-to-end manner without manual interferences, deep learning techniques have boosted great achievements in a wide range of remote sensing applications. Consequently, deep learning techniques have also been investigated for land cover mapping of multispectral LiDAR data. Pan *et al.* [13] proposed an optimized convolutional neural network (CNN) model for land cover classification of multispectral LiDAR data. Specifically, four sets of feature images were rasterized based on the intensity and elevation properties for labeling pixel categories. In addition, a deep Boltzmann machine (DBM) model was also presented by Pan *et al.* [14] to carry out land cover classification by using multispectral LiDAR data. Yu *et al.* [15] designed a hybrid capsule network (HCapsNet) architecture, which consisted of a capsule convolutional branch for mining local feature encodings and a fully connected capsule branch for characterizing global feature representations. The classification of land cover types was conducted with the combination of the local and global features. Li *et al.* [16] trained a graph geometric moments CNN to extract buildings from multispectral LiDAR data. In this model, first, a farthest point sampling-k nearest neighbors sample generation strategy was applied to obtain operable samples. Then, a graph convolutional network was leveraged to obtain the point-wise labeling of the multispectral LiDAR points. Furthermore, multisource data fusion strategies by integrating multispectral LiDAR data and remote sensing images have also been exploited for land cover mapping purposes [17], [18].

In this letter, we design an effective capsule network integrated with capsule attention mechanisms for land cover classification of multispectral LiDAR data. This network takes five types of feature images interpolated from the multispectral LiDAR data as the input and outputs a pixel-wise land cover labeling result. The contributions include the following: 1) an effective capsule encoder–decoder architecture is investigated to extract high-quality features for pixel-wise land cover classification and 2) an efficient capsule-based self-attention module is designed to boost the feature encoding semantics.

II. Multispectral LiDAR Data and Data Preprocessing

A. Multispectral LiDAR Data

In this letter, two study areas located in Ontario, Canada were surveyed to collect the multispectral LiDAR data for the land cover classification task. The multispectral LiDAR data were collected using an airborne Titan multispectral LiDAR system manufactured by the Teledyne Optech. This system was configured with three active spectral channels that worked in intermediate infrared (1550 nm), near infrared (1064 nm), and visual (532 nm) wavelengths, respectively. The three channels emitted laser beams with separate forward angles (3.5°, 0°, and 7°) to produce independent scan lines, thereby resulting in an independent point cloud for each channel. The first data set was collected in Whitchurch-Stouffville (WS) covering an area of about 3.21 km^2. It was composed of 19 flying strips. The second data set was collected in Tobermory (TB) covering an area of about 1.99 km^2. It was composed of ten flying strips.

B. Data Preprocessing

Instead of directly processing the 3-D multispectral LiDAR data, we rasterize them into a set of feature images according to the multichannel geometrical and spectral properties to improve the processing efficiency. Concretely, first, the three sets of multichannel LiDAR points are registered and merged into a single LiDAR point set based on their geographical coordinates. Then, vertical gridding along the Z axis is performed to structurize the merged LiDAR points into a grid representation with a grid size (spatial resolution) of $r_g = 0.5$ m. Finally, a single pixel is interpolated for the LiDAR points within each grid to form a feature image. The gray value of a pixel is interpolated according to the properties of the LiDAR points in the corresponding grid by using the inverse distance weighted interpolation method [19]. In this study, we rasterize five types of feature images by fully considering the elevation, number of returns, and three channels of spectral intensities. Specifically, all the merged LiDAR points in a grid are leveraged to generate the elevation and number of returns images, whereas only the LiDAR points from the related channel are considered for obtaining the spectral intensity images.

III. Method

A. Revisit of Capsule Network

The basic components constituting the capsule networks are vectorial capsules, which can be viewed as a kind of 1-D tensor representation. Such a capsule formulation can simultaneously encode both the existence probability and the instantiation property of a feature by, respectively, using its length and parameters. Furthermore, it can also enable a capsule to self-adaptively identify a feature and its variants. In a capsule

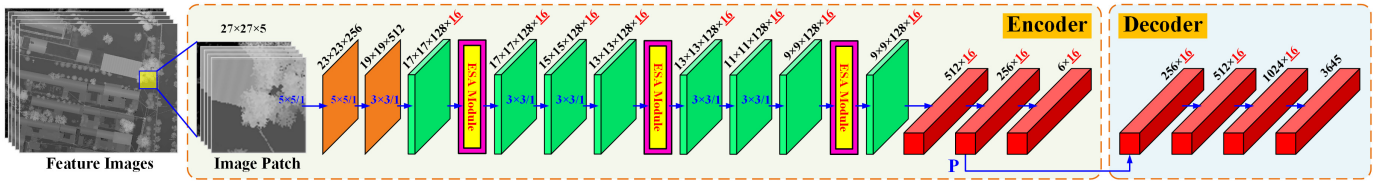

Fig. 1. Architecture of the ESA-CapsNet. The number 16 denotes the dimension of a capsule.

network, the input to a capsule is a weighted aggregation over the predictions from the prepositive capsules as follows:

$$\boldsymbol{C}_j = \sum_i a_{ij} \boldsymbol{U}_{ij} \tag{1}$$

where $\boldsymbol{C}_j$ is the aggregated input to capsule j; a_{ij} is a coupling coefficient reflecting the contribution degree of capsule i to activate capsule j, which is dynamically determined by the improved dynamic routing process [20]; $\boldsymbol{U}_{ij}$ is the prediction cast from capsule i and is computed as follows:

$$\boldsymbol{U}_{ij} = \mathbf{W}_{ij} \boldsymbol{U}_i \tag{2}$$

where $\boldsymbol{U}_i$ is the normalized output of capsule i and $\mathbf{W}_{ij}$ a feature mapping matrix.

As for the capsule length-based feature probability encoding mechanism, the longer a capsule is, the higher the probability prediction should be. To this end, a squashing function [21] is specially designed as the activation function to normalize the aggregated input of a capsule as follows:

$$\boldsymbol{U}_i = \frac{\|\boldsymbol{C}_i\|^2}{1 + \|\boldsymbol{C}_i\|^2} \cdot \frac{\boldsymbol{C}_i}{\|\boldsymbol{C}_i\|}. \tag{3}$$

As a result, a long capsule is restrained to a length close to one to cast a high prediction, whereas a short capsule is weakened to almost a zero length to contribute quite few.

B. Efficient Self-Attention Capsule Network

To make full use of the advanced properties of the capsule representations in high-order feature encoding, we design an efficient self-attention capsule network (ESA-CapsNet) for carrying out land cover classification by using the rasterized feature images of the multispectral LiDAR data. To facilitate processing, we fuse the five types of feature images into a multispectral image structure, each of whose pixels contains five channels of intensities made of the corresponding values from the five feature images. As shown in Fig. 1, the input of the ESA-CapsNet is an image patch of $n \times n$ pixels centered at a position. The output of the ESA-CapsNet is the predicted class label of the central pixel of the image patch.

The architecture of the ESA-CapsNet involves an encoder for patch feature extraction and classification and a decoder for reconstructing the input patch to enhance the feature encoding capability of the encoder. The encoder consists of a set of conventional convolutional layers, capsule convolutional layers, and capsule fully connected layers. The low-level scalar features extracted by the conventional convolutional layers are further fed into the capsule convolutional layers to abstract high-level capsule features. This is achieved by performing conventional convolutions on the second conventional convolutional layer, followed by feature channel grouping and capsule vectorizing, resulting in a multidimensional capsule

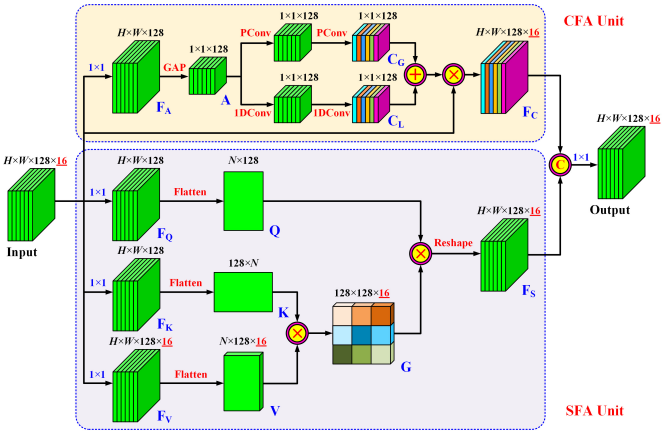

Fig. 2. Architecture of the capsule-based efficient self-attention (ESA) module.

at each position of the feature map. By collecting the local capsule features with a global perspective, the capsule fully connected layers finally predict a class label for the central pixel of the input patch. Specifically, the conventional convolutional layers are activated using the rectified linear unit (ReLU) and the capsule layers are activated using the squashing function. The last layer of the encoder is a softmax layer constituted by a set of class-specific capsules, each of which represents a land cover type to be predicted. Thus, the softmax function is applied to the capsule lengths to produce a one-hot prediction.

To enhance the feature representation quality and the model robustness of the ESA-CapsNet, we design a capsule-based efficient self-attention (ESA) module and integrate it into the encoder (Fig. 1). The architecture of the ESA module is shown in Fig. 2. The ESA module involves two parallel branches named channel feature attention (CFA) unit and spatial feature attention (SFA) unit for, respectively, recalibrating the channel features and spatial features. The output feature map of the ESA module has the identical size to the input feature map. For the CFA unit, first, a 1×1 capsule convolution is performed on the input multidimensional capsule feature map to convert it into a 1-D capsule feature map F_A, which encodes mainly the feature probability properties. Then, global average pooling (GAP) is applied to obtain the channel-wise statistics, generating a channel descriptor A. Next, two sibling branches are mounted on A to exploit channel-wise interdependencies in a global manner and a local manner, respectively. To exploit global channel-wise interdependencies, two convolutional layers with point-wise convolutions (PConv) having a kernel size of 1×1 across the channels are appended. In contrast, to exploit local channel-wise interdependencies, two convolutional layers with 1-D convolutions (1-DConv) having a kernel size of $k = 5$ sliding along the channels are connected. The outputs C_G and C_L of these two branches

are added and normalized by the sigmoid function to form a channel attention descriptor, which encodes the channel-wise feature informativeness. Finally, the input feature map is multiplied by the channel attention descriptor in a channel-wise manner to produce a recalibrated feature map F_C, where the informative features are effectively highlighted.

For the SFA unit, first, three 1×1 capsule convolutions are performed on the input feature map to obtain a query feature map $F_Q \in R^{H\times W\times 128}$, a key feature map $F_K \in R^{H\times W\times 128}$, and a value feature map $F_V \in R^{H\times W\times 128\times 16}$, where H and W are the height and width of the input feature map, respectively. To facilitate computation, these feature maps are channel-wisely flattened to form the query matrix $Q \in R^{N\times 128}$, the key matrix $K \in R^{128\times N}$, and the value matrix $V \in R^{N\times 128\times 16}$, where $N = H \times W$. Then, multiplication is conducted on K and V to generate a global context matrix $G \in R^{128\times 128\times 16}$. Here, each row of K functions as a single-channel spatial attention map, which reflects a semantic property of the entire input and acts as a weight regulator over all the positions to aggregate the value features from V. Thus, each row of G summarizes a global, semantic property of the input feature map. Specifically, a softmax function is applied to K in a row manner before multiplication. Next, regarding each row of Q as the spatial attention coefficients of a position, G is multiplied to Q to produce a recalibrated feature for each position. Finally, through reshaping, the SFA unit outputs a recalibrated feature map F_S, where the class-specific features are effectively emphasized. As shown in Fig. 2, the recalibrated feature maps F_C and F_S from the CFA and SFA units are concatenated and fused through a 1×1 capsule convolution to obtain the output feature map, which is significantly promoted by taking into consideration both the channel and SFAs.

As shown in Fig. 1, the decoder takes the output of the second capsule fully connected layer P of the encoder as the input and reconstructs the input image patch through a series of capsule fully connected layers. The decoder, appearing only at the training stage, functions to force the encoder to extract strong and representative feature semantics toward high-quality classification.

C. Loss Function

The loss function is designed as the following multitask loss function to direct the training of the encoder and decoder:

$$L = \sum_{i=1}^{M} L_{\text{cls}} + \lambda \sum_{i=1}^{M} L_{\text{rec}} \tag{4}$$

where L_{cls} and L_{rec} are the classification and reconstruction loss terms, respectively; M is the number of training image patches; λ is a regularization factor to balance the two loss terms. L_{cls} is formulated as the focal loss of the target output of the encoder. L_{rec} is computed as the mean-squared error loss between the reconstruction of the decoder and the corresponding input.

IV. Results and Discussion

A. Land Cover Classification

At the training stage, 60% of the labeled data were randomly selected from each of the two data sets for constructing the ESA-CapsNet. At the test stage, the remaining 40% of the labeled data were applied to assess the land cover classification performance. The surveyed areas of the two data sets were labeled into six types of land covers: 1) water; 2) vegetation; 3) road; 4) soil; 5) building; and 6) other impervious surface. To provide quantitative evaluations on the classification accuracy, the following metrics were computed: overall accuracy (OA), average accuracy (AA), and kappa coefficient (κ). The land cover classification results obtained on the two data sets are quantitatively reported in Tables I and II, respectively. The results were obtained by conducting ten Monte Carlo runs and calculating the mean and standard deviation of these metrics. For visual inspection purpose, Fig. 3 also presents the land cover classification results on the two data sets. The six types of land covers are rendered with different colors.

As detailed in Table I, the OA, AA, and κ values obtained on the WS data set are 98.42% $\pm$ 0.14%, 95.15% $\pm$ 0.15%, and 0.9776 $\pm$ 0.0013, respectively. Specifically, the ESA-CapsNet achieved a superior accuracy in identifying the land cover type of water. In contrast, a relatively lower classification accuracy was achieved on the land cover type of building. Furthermore, similar classification accuracies were achieved on the land cover types of vegetation and other impervious surface. As reflected in Table II, an overall classification performance with the OA, AA, and κ values of 98.91% $\pm$ 0.09%, 95.73% $\pm$ 0.12%, and 0.9837 $\pm$ 0.0009, respectively,

TABLE I
LAND COVER CLASSIFICATION RESULTS ON THE WS DATA SET

Type	Proposed	RB	ML	CNN	DBM	HCapsNet
1	99.35±0.11	94.91±0.89	94.76±0.82	98.44±0.12	98.10±0.13	99.11±0.12
2	95.27±0.16	88.13±0.98	87.99±1.07	93.17±0.20	92.42±0.24	94.53±0.18
3	94.56±1.08	82.80±2.41	82.92±2.62	91.86±1.23	83.63±1.31	93.72±1.17
4	93.31±0.35	86.27±1.42	85.84±1.35	91.43±0.49	88.71±0.57	92.91±0.41
5	93.17±1.78	87.71±3.44	88.63±3.37	91.15±2.05	88.95±2.43	92.23±1.95
6	95.24±1.57	91.32±3.12	90.25±3.19	93.91±1.98	93.54±2.32	94.76±1.78
OA(%)	**98.42±0.14**	91.57±1.13	91.23±1.12	95.91±0.21	94.36±0.27	97.89±0.15
AA(%)	**95.15±0.15**	88.52±1.32	88.40±1.29	93.33±0.24	90.89±0.32	94.54±0.17
κ×100	**97.76±0.13**	90.22±1.25	89.67±1.23	95.34±0.18	93.78±0.29	97.13±0.14

TABLE II
LAND COVER CLASSIFICATION RESULTS ON THE TB DATA SET

Type	Proposed	RB	ML	CNN	DBM	HCapsNet
1	99.52±0.09	95.13±0.87	95.34±0.76	98.76±0.11	98.35±0.12	99.34±0.11
2	96.13±0.14	89.20±0.95	88.92±1.02	94.45±0.18	93.51±0.22	95.25±0.16
3	94.82±0.89	83.69±2.31	83.17±2.52	92.56±1.20	88.26±1.28	94.17±1.16
4	94.47±0.33	87.71±1.33	88.05±1.28	91.77±0.43	90.13±0.55	93.52±0.39
5	93.55±1.54	87.92±3.36	88.21±3.29	91.51±1.96	89.92±2.39	92.41±1.87
6	95.89±1.32	91.55±3.08	91.74±3.10	94.22±1.84	93.81±2.24	95.02±1.66
OA(%)	**98.91±0.09**	91.89±1.12	92.15±1.08	96.68±0.15	95.13±0.24	98.34±0.09
AA(%)	**95.73±0.12**	89.20±1.29	89.24±1.22	93.88±0.19	92.33±0.27	94.95±0.14
κ×100	**98.37±0.09**	91.03±1.21	91.27±1.06	95.85±0.16	94.24±0.25	97.76±0.11

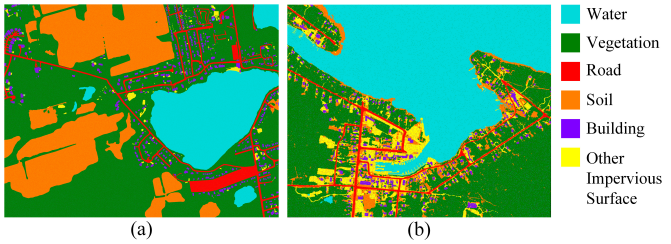

Fig. 3. Illustration of the land cover classification results on (a) WS data set and (b) TB data set.

was obtained on the TB data set. Likewise, the best and worst classification accuracies appeared on the land cover types of water and building, respectively. The reason causing the classification accuracy difference is that, compared with the homogeneous and unique features of the water bodies, the building regions exhibited great diversities in geometrical structures and spectral properties. The classification errors mainly appeared at the border areas of two land cover types. For example, some pixels of the roads were falsely recognized as the soil or other impervious surface. Moreover, structure incompleteness was generated for some building and road regions occluded by high-rise trees. In the whole, benefitting from the design of the capsule network architecture integrated with the ESA modules for capsule feature promotion, the ESA-CapsNet behaved promisingly on land cover classification of multispectral LiDAR data.

B. Comparative Study

To further prove the effectiveness of the ESA-CapsNet in land cover classification of multispectral LiDAR data, a group of comparative tests were also conducted with the following five methods: RB method [2], ML classifier [3], CNN [13], DBM [14], and HCapsNet [15]. For fair comparisons, the same training and test data were used to construct these models and evaluate their performances. The quantitative evaluation results obtained on the two data sets are detailed in Tables I and II, respectively. Apparently, the CNN and HCapsNet performed superiorly over the RB and ML, and obtained a slightly better performance than the DBM. Specifically, the HCapsNet achieved the highest accuracy among the five methods, whereas a relatively lower performance was obtained by the RB. Compared with the low-level features or rules adopted in the RB and ML, the advanced performance of the CNN, HCapsNet, and DBM was due to the exploration of high-level, deep, and semantically strong feature representations by using deep learning models. Note that capsule features were intensively exploited in the HCapsNet, thereby effectively enhancing the classification accuracy. Comparatively, designed with the effective encoder–decoder capsule network architecture boosted by the ESA modules to recalibrate the channel and spatial features to upgrade the feature encoding quality and robustness, the proposed ESA-CapsNet outperformed the compared methods with respect to the overall classification accuracies. In conclusion, the proposed ESA-CapsNet provided a promising and feasible solution to land cover classification of multispectral LiDAR data.

V. Conclusion

This letter has presented a novel capsule network, named ESA-CapsNet, for land cover classification of multispectral LiDAR data. Input with the rasterized feature images of the multispectral LiDAR data, the encoder–decoder architecture of the ESA-CapsNet can generate high-level, informative, and strong feature representations to provide pixel-wise land cover predictions. Integrated with the ESA modules for channel and spatial feature recalibrations, the feature quality and semantics were further upgraded to promote the classification capability of the network. Quantitative evaluations on two data sets showed that an overall classification performance with the OA, AA, and κ values of over 98.42%, 95.15%, and 0.9776, respectively, has been achieved. Comparative studies with five existing methods also proved the effectiveness and feasibility of the ESA-CapsNet in the land cover classification tasks.

References

[1] W. Y. Yan, A. Shaker, and N. El-Ashmawy, "Urban land cover classification using airborne LiDAR data: A review," *Remote Sens. Environ.*, vol. 158, pp. 295–310, Mar. 2015.

[2] N. Ekhtari, C. Glennie, and J. C. Fernandez-Diaz, "Classification of airborne multispectral LiDAR point clouds for land cover mapping," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 6, pp. 2068–2078, Jun. 2018.

[3] S. Morsy, A. Shaker, and A. El-Rabbany, "Multispectral LiDAR data for land cover classification of urban areas," *Sensors*, vol. 17, no. 5, p. 958, Apr. 2017.

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

[5] K. Karila, L. Matikainen, E. Puttonen, and J. Hyyppa, "Feasibility of multispectral airborne laser scanning data for road mapping," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 3, pp. 294–298, Mar. 2017.

[6] W. Dai, B. Yang, Z. Dong, and A. Shaker, "A new method for 3D individual tree extraction using multispectral airborne LiDAR point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 144, pp. 400–411, Oct. 2018.

[7] F. Naveed, B. Hu, J. Wang, and G. B. Hall, "Individual tree crown delineation using multispectral LiDAR data," *Sensors*, vol. 19, no. 24, p. 5421, Dec. 2019.

[8] M. Kukkonen, M. Maltamo, L. Korhonen, and P. Packalen, "Multispectral airborne LiDAR data in the prediction of boreal tree species composition," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 3462–3471, Jun. 2019.

[9] Q. Wang and Y. Gu, "A discriminative tensor representation model for feature extraction and classification of multispectral LiDAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 1568–1586, Mar. 2020.

[10] P. Hartzell, C. Glennie, K. Biber, and S. Khan, "Application of multispectral LiDAR to automated virtual outcrop geology," *ISPRS J. Photogramm. Remote Sens.*, vol. 88, pp. 147–155, Feb. 2014.

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

[12] A. Stefanidou, I. Z. Gitas, L. Korhonen, N. Georgopoulos, and D. Stavrakoudis, "Multispectral LiDAR-based estimation of surface fuel load in a dense coniferous forest," *Remote Sens.*, vol. 12, no. 20, p. 3333, Oct. 2020.

[13] S. Pan *et al.*, "Land-cover classification of multispectral LiDAR data using CNN with optimized hyper-parameters," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 241–254, Aug. 2020.

[14] S. Pan, H. Guan, Y. Yu, J. Li, and D. Peng, "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.*, vol. 12, no. 4, pp. 1314–1326, Apr. 2019.

[15] Y. Yu *et al.*, "A hybrid capsule network for land cover classification using multispectral LiDAR data," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 7, pp. 1263–1267, Jul. 2020.

[16] D. Li *et al.*, "Building extraction from airborne multi-spectral LiDAR point clouds based on graph geometric moments convolutional neural networks," *Remote Sens.*, vol. 12, no. 19, p. 3186, Sep. 2020.

[17] R. Hansch and O. Hellwich, "Fusion of multispectral LiDAR, hyperspectral, and RGB data for urban land cover classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 2, pp. 366–370, Feb. 2021.

[18] D. Hong, J. Chanussot, N. Yokoya, J. Kang, and X. X. Zhu, "Learning-shared cross-modality representation using multispectral-LiDAR and hyperspectral data," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 8, pp. 1470–1474, Aug. 2020.

[19] Y. Yu, J. Li, H. Guan, C. Wang, and J. Yu, "Automated detection of road manhole and sewer well covers from mobile LiDAR point clouds," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 9, pp. 1549–1553, Sep. 2014.

[20] J. Rajasegaran, V. Jayasundara, S. Jayasekara, H. Jayasekara, S. Seneviratne, and R. Rodrigo, "DeepCaps: Going deeper with capsule networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, Jun. 2019, pp. 10725–10733.

[21] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Proc. Conf. Neural Inform. Process. Syst.*, Long Beach, CA, USA, 2017, pp. 1–11.