World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:17, No:11, 2023

Continual Learning Using Data Generation for Hyperspectral Remote Sensing Scene Classification

Samiah Alammari, Nassim Ammour

***Abstract*—**When providing a massive number of tasks successively to a deep learning process, a good performance of the model requires preserving the previous tasks data to retrain the model for each upcoming classification. Otherwise, the model performs poorly due to the catastrophic forgetting phenomenon. To overcome this shortcoming, we developed a successful continual learning deep model for remote sensing hyperspectral image regions classification. The proposed neural network architecture encapsulates two trainable subnetworks. The first module adapts its weights by minimizing the discrimination error between the land-cover classes during the new task learning, and the second module tries to learn how to replicate the data of the previous tasks by discovering the latent data structure of the new task dataset. We conduct experiments on hyperspectral image (HSI) dataset on Indian Pines. The results confirm the capability of the proposed method.

***Keywords*—**Continual learning, data reconstruction, remote sensing, hyperspectral image segmentation.

I. Introduction

HYPERSPECTRAL remote sensing image is a cube of data containing hundreds of gray-level images, where each gray-level image is captured in a contiguous channel wavelength with a very high spectral resolution. In order to exploit the rich and useful spectral information, automatic classification using HSI has been developed for many exploration applications [1].

In the last few years, the use of HSI has been increasing due to their rich information capability. Different works, e.g. [2]-[4], have been conducted to have improvement in Convolutional Neural Network (CNN) for HSI to extract the spatial-spectral feature, for instance, 3D CNN model [2].

Although research on artificial intelligence has seen significant progress recently, most of the works mainly focus on stationary environments with fixed datasets, e.g. [2]-[4], where the authors tried to classify scenes upon already defined datasets but not dynamic, which is often not the case of real-world applications. In fact, real-world scenarios are usually dynamic; the data are erratically changing, or sometimes momentarily available which make it impossible to handle using traditional learning processes, since those processes are unable to adapt their behavior with the environment changes, thus resulting in what is called the catastrophic forgetting problem. Indeed, modern practical applications stipulate the use of a more adaptive deep learning model that is able to maintain the learning process over time, hence, the continual learning model is proposed.

Continual learning is a novel area of study that discusses the possibility for artificial systems to learn from a continuous stream of interrelated data sequentially, similar to biological systems such as human cognition [3], which demonstrate the ability of reviewing old learned data. Also known as incremental learning, continual learning requires the presence of a dataset containing all the classes to be used in the training process.

Some previous works have studied continual learning in deep learning. For instance, Kim et al. in [4] solved the problem of catastrophic forgetting by using StackNet. StackNet is a two-module approach that consists of a generative adversarial network (GAN) as an index that generates task-based synthetic data relate. The index module decides which portion of the StackNet should be used for training this new task. Authors in [5] proposed a method to identify the parameters that are sensitive to changes in each task and penalize large change to them when learning a new task. In [6], cumulative learning is achieved by adding new nodes to each layer in the model, and the model is retrained to learn a new task. In [7], the authors proposed a learning without forgetting solution for multi-label and regression problems. Still, this method suffers from the mixing issue between the different tasks encountered in the test phase. In [8], to train several tasks sequentially without forgetting older ones, the author used elastic weight consolidation (EWC) method that remembers previous tasks knowledge during new learning.

Recently, few works were proposed in the remote sensing imagery area. For instance, [9] proposed a continual learning model for sequential scene classification. The problem involved land-cover data. They propose two models, the first model while keeping track of the old tasks, learns the knowledge within the new task using cross-entropy and mean square error (MSE). The second model learns to separate tasks from the stored samples of each class in linear memory.

In this paper, we tackle the catastrophic forgetting problem in deep learning, and propose a continual learning technique that endows the deep-learning model with the ability to perform well without forgetting the previous knowledge. The proposed technique is founded on two main complementary and trainable modules: The first module adjusts its parameters by discriminating between the land-cover classes within the new task using categorical cross-entropy loss, while the second module is trained to extract and save the latent structure of the new task data, and then used as a data generator for old tasks. The proposed strategy is evaluated using the HSI dataset

Samiah Alammari and Nassim Ammour are with Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia (e-mail: 439204074@student.ksu.edu.sa, nammour@ksu.edu.sa)

(Indian Pines) and the experimental results are reported and discussed.

II. Proposed Continual Learning Method

Continual learning multi-class classification task consists of a long stream of tasks $T_l = \left\{X_i^{(l)}, y_i^{(l)}\right\}_{i=1}^{n_l}, l = 1, \dots k, .., K$, for each task T_l we have n_l images $X^{(l)}$ and their corresponding labels $y^{(l)}$. The objective of the role is to increase the capability by adding c_l unseen classes. We aim to train a unified classifier on a new task T_{New} and make it able to perform well on both old and new tasks (T_{New} and T_{old}). In the next parts, we explain the main steps of proposed model.

A. Learning First Task T_1

In the experiments, we adopt the Fast 3D CNN model from [2]. A feature extractor sub-module in the model utilizes spatial and spectral feature maps. In order to overcome the overlapping and the high intra-classes variability in the HSI pixels, we apply the incremental principle component analysis (iPCA) which reduces the number of bands. Furthermore, to generate 3D feature maps, the HSI cube is divided into small overlapping 3D patches, the ground label of these patches is based on the central pixel. The small 3D patches are used as input into the 3D CNN model. The architecture of the model is found in Table I.

TABLE I
Layer of 3D CNN Model Architecture with Window Size Set as 11 × 11 of Indian Pines Dataset

Layer	Output Shape	# of Parameters
Input Layer	(11, 11, 20, 1)	0
Conv3D 1 (Conv3D)	(9, 9, 14, 8)	512
Conv3D 2 (Conv3D)	(7, 7, 10, 16)	5776
Conv3D 3 (Conv3D)	(5, 5, 8, 32)	13856
Conv3D 4 (Conv3D)	(3, 3, 6, 64)	55360
Flatten 1 (Flatten)	(3456)	0
Dense 1 (Dense)	(256)	884992
Dropout 1 (Dropout)	(256)	0
Dense 2 (Dense)	(128)	32896
Dropout 2 (Dropout)	(128)	0
Dense 3 (Dense)	(16)	2064
total trainable parameters		995,459

The model function is described as:

$$v_{i,j}^{x,y,z} = \mathcal{F}\left(\sum_{T=1}^{d_l-1}\sum_{\lambda=-v}^{v}\sum_{P=-\gamma}^{\gamma}\sum_{\phi=-\delta}^{\delta} w_{i,j,T}^{v,\rho,\lambda} \times v_{(i-1),T}^{(x+v),(y+p),(z+\lambda)} + b_{i,j}\right) \quad (1)$$

where (x, y, z) are the activation values at spatial position in the i^{th} layer and j^{th} feature map, $\mathcal{F}$ is an activation function, d_{l-1} is the number of 3D feature maps at $(l\text{ -}1)^{th}$ layer and $w_{i,j}$ is the depth of the kernel, $b_{i,j}$ is the bias, $2\delta + 1$, $2\lambda + 1$ and $2v + 1$ is the height, width and depth of the kernel.

We truncate this network by augmenting this truncated network with a new softmax classification with the total number of classes outputs C. Then, we use the c_1 outputs related to the task T_1 number of classes and put zeros at the $C - c_1$ remaining outputs. Afterwards, we train the network to learn the weights $\varphi^{(1)}$ by minimizing the standard categorical cross-entropy loss. For learning the first task $T_1 = \left\{X^{(1)}, y^{(1)}\right\}$, we first:

$$\mathcal{L}_{\varphi} = -\sum_i^{c_1} y_i^{(1)}\, log(\hat{y}_i^{(1)}) \quad (2)$$

where $y_i^{(1)}$ is true categorical class label of the image $X^{(1)}$ and $\hat{y}_i^{(1)}$ is the output probability vector provided by the softmax classification layer.

B. The Generative model

To fight the forgetting phenomenon, we design a deep learning-based generative model which recycles the old learned tasks using an information generator sub-module. Moreover, a Variational Auto-Encoder (VAE) structure is added and trained in an end-to-end way with the classifier. The first block of the VAE is the encoder neural network which encodes the input data into a latent representation in a lower stochastic space. The second block of the VAE is the decoder neural network which takes as input a representation sample from the encoder distribution to reconstruct. The parameters of the VAE are learned by minimizing the loss function which encapsulates a negative likelihood and a regularizer.

$$\mathcal{L}_{\theta,\phi} = \sum_i^N \left[-\mathbb{E}_{z\sim q_\theta(Z|x_i)}\left[log p_\phi(x_i|Z)\right] + \mathbb{KL}\left(q_\theta(Z|x_i) \parallel p(Z)\right)\right] \quad (3)$$

where N is the total data-points. The first term in $\mathcal{L}_{\theta,\phi}$ is the reconstruction loss, or expected negative log-likelihood of the *i-th* data-point. The second term in the loss function is the Kullback-Leiber used to measure the divergence between the encoder's distribution $q_\theta(Z|x)$ and $p(Z)$.

C. Learning Task T_k

After $k - 1$ tasks, we need to train the model on a new task $T_k = \left\{X^{(k)}, y^{(k)}\right\}$. In order to overcome the old tasks knowledge missing, the probabilistic generative model produces fictional data sampled from old tasks data distributions, then, we classify the generated input data representatives to obtain their labels. After this step, the new task dataset is augmented by the generated data. And finally, the obtained dataset is used to train the classifier and the generative module. we update the weights of the model $(\varphi, \theta, \phi)^{(1)}, \dots, (\varphi, \theta, \phi)^{(k)}$ by minimizing the loss function:

$$\mathcal{L}_{\varphi,\theta,\phi,k} = -\lambda_1 \sum_i^{c_k} y_i^{(k)}\, log\left(\hat{y}_i^{(k)}\right) + \lambda_2 \sum_i^{N_k}\left[-\mathbb{E}_{z\sim q_\theta\left(Z|x_i^{(k)}\right)}\left[log p_\phi\left(x_i^{(k)}\middle|Z\right)\right] + \mathbb{KL}\left(q_\theta\left(Z|x_i^{(k)}\right) \parallel p(Z)\right)\right] \quad (4)$$

where the first term in (4) is the categorical cross-entropy loss and the second term contains the negative likelihood and the regularizer losses for the task T_k. The balancing parameters $\lambda_1$and λ_2 are used for controlling the contribution of each loss term to the total loss $\mathcal{L}_{\varphi,\theta,\phi,k}$.

III. Experimental Results

A. Dataset Description

In the experiments, we use Indian Pines dataset (IP). The dataset was gathered by using an AVRIS sensor in the north-western part of the state of Indiana. The dataset consists of 145 * 145 pixels images and a 224 spectral reflectance band of 0.4-2.5 · 10^{-6} meters wavelengths. The dataset has a spatial resolution of 20 m/px. This dataset for the most part covers the agricultural areas, and the rest is forests and dense vegetation. The dataset is labelled into 16 classes [10]. Fig. 2 shows ground image from IP dataset.

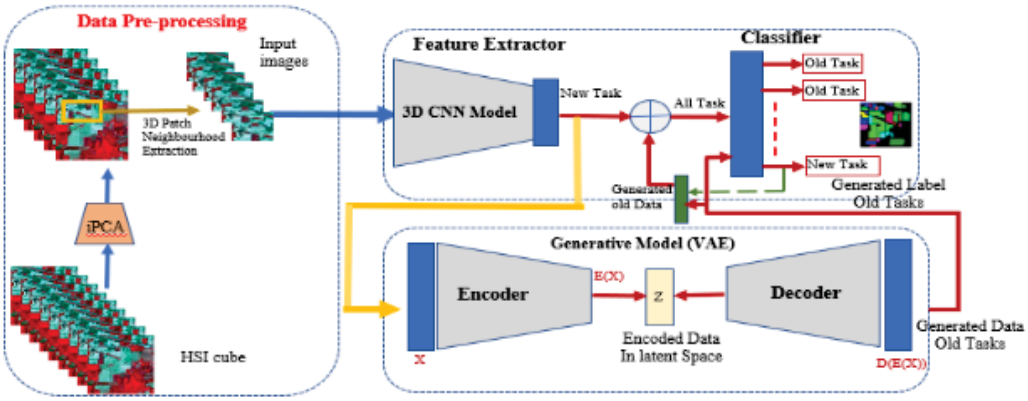

Fig. 1 Flowchart of the proposed continual learning approach

B. Experiment Setup

To explicitly demonstrate the performance of our proposed model, we divide the IP dataset into eight groups of tasks, and each group includes two classes. In the learning phase, we train the model each time on a new task, thereafter, the model auto-generates samples of the previously learned tasks and adds them automatically to the new task's dataset. Regarding the optimization algorithm, we use Adam optimizer with default parameters to update the learning weight. In order to test the accuracy of the proposed model, we split the dataset into 80% for training and 20% for testing.

C. Results

1. Joint Learning

Initially, we conduct a first experiment by training the model using all the classes in the dataset (16 classes). The purpose of this experiment is to used it accuracy as baseline to evaluate the performances of the continual learning strategies. An Overall Accuracy (OA) of 97% is obtained for the baseline experiment.

TABLE II.
Overall Accuracy (OA) in [%] Obtained for (IP) Datasets

Task	Accuracy (%)
	IP Dataset
Joint	97.78
1	100 ± 0.00
2	99 ± 0.00
3	99 ± 0.00
4	96 ± 3.46
5	96 ± 0.57
6	93 ± 1.73
7	90 ± 2.51
8	84 ± 2.51

2. Continual Learning Aspect

We implement the proposed continual learning architecture and report the model performance results in Table II. As can be noticed, the proposed architecture performs the first task with an OA of 100% then, and after sequentially adding all the tasks, the performance decays slowly and reaches an accuracy of 84 ± 2.51 compared to the joint training. From Table I, we can notice that the accuracy decays slowly, which indicates a slow forgetting effect. The proposed continual learning strategy fights this challenging forgetting effect and tries to sustain a high performance by building the hidden structure of the already seen data of old tasks.

3. Comparison to Other Methods

In this section, we conduct a comparative experiment of our approach with other generative methods using the IP dataset. The first method surmounts the forgetting by selectively slowing down learning on the weights important for those tasks (EWC). On the other hand, in the second method we use another generator deep model GAN to fight the forgetting phenomenon. We performed the conditioning by feeding the label as an additional input layer into the discriminator and generator. The results of the comparison experiments shown in Table III, confirm, obviously, the efficiency of the proposed method.

TABLE III
Comparison between the Results of the Proposed Network Using VAE, EWC and CGAN on IP Dataset in Terms of Classification Accuracy (acc %)

Task	EWC	CGAN	Proposed
1	64	100	100
2	31	88	99
3	97	90	99
4	98	91	96
5	29	78	96
6	74	70	93
7	76	78	90
8	76	82	84

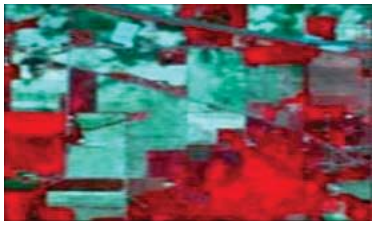

Fig. 2 Ground image of IP dataset

IV. Conclusions

In this paper, we studied the catastrophic forgetting problem in deep learning and proposed a continual learning strategy for scene classification in remote sensing imagery. The proposed model demonstrated the importance of learning and conserving the latent structure of the previous tasks for solving the deep learning forgetting problem. Experiment results on the Indian Pines dataset revealed the efficiency of the proposed continual learning model in comparison to other methods.

References

[1] "Hyperspectral and Multispectral Imaging," Edmund Optics, 07 06 2020. (Online). Available: https://www.edmundoptics.com/knowledge-center/application-notes/imaging/hyperspectraland-multispectral-imaging/. (Accessed 27 03 2022).

[2] M. Ahmad, A. M. Khan, M. Mazzara, S. Distefano, M. Ali and M. S. Sarfraz, "A Fast and Compact 3-D CNN for Hyperspectral Image Classification," *in IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1-5, 2022, Art no. 5502205, doi: 10.1109/LGRS.2020.3043710

[3] R. Hadsell, D. Rao, A.A. Rusu, and R. Pascanu, "Embracing Change: Continual Learning in Deep Neural Networks," Trends in Cognitive Sciences, November 2020

[4] J. Kim, J. Kim and N. Kwak, "StackNet: Stacking feature maps for Continual learning," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 2020, pp.975-982, doi:10.1109/ CVPRW50498 .2020.00129.

[5] L. Butyrev, G. Kontes, C. Löffler, and C. Mutschler, "Overcoming Catastrophic Forgetting via Hessian-free Curvature Estimates," 2019.

[6] J. Xu and Z. Zhu, "Reinforced continual learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 899–908.

[7] Z. Li and D. Hoiem, "Learning without Forgetting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2935–2947, Dec. 2018, doi: 10.1109/TPAMI.2017.2773081.

[8] J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," *Proc. Nat. Acad. Sci. USA*, vol. 114, no. 13, pp. 3521–3526, Mar. 2017, doi:10.1073/pnas. 1611835114.

[9] N. Ammour, Y. Bazi, H. Alhichri, and N. Alajlan, "Continual Learning Approach for Remote Sensing Scene Classification," *IEEE Geosci. Remote Sens. Lett.*, 2020.

[10] Hyperspectral Datasets Description, 2022 (accessed 2022-01-12), http://www.ehu.eus/ccwintco/index.php/HyperspectrlRemoteSensing Scenes.