VOL 2, NO.1 (2022) 31-48
https://jagst.utm.my

JAGST

Journal of Advanced Geospatial
Science & Technology

Lightweight Convolutional Neural Network for Land Use Image Classification

Dwijendra Nath Dwivedi[1], Ganesh Patil[2]
[1] Crakow university of Economics, Rakowicka 27, Kraków, 31-510, Poland
[2] Finance and Strategy, Indian Institute of Management, Lucknow, Uttar Pradesh, India
Email: dwivedy@gmail.com, ga3435@gmail.com

Abstract **–** Convolutional Neural Networks (CNN) have proven to be pivotal in advancements in the domain of computer vision. One of the most important and highly researched applications of CNN is in land use classification using aerial imagery. While there is a lot of research conducted using advanced techniques like transfer learning, data augmentation, CNN cascading, and many more to elevate the performance of classification models, the power of a combination of simpler and computationally efficient approaches to CNN configuration like relevant filtering, better normalization, and accurate placement of dropouts is often underestimated. Although highly deep and complex architectures provide better accuracy, they exhibit adverse performance in terms of computational cost, time, and effort required to develop and deploy the models. Thus, there is always a tradeoff between improved accuracy and ease of development and implementation. This paper demonstrates a lightweight CNN configuration that results in a significantly high validation accuracy of 88.29% on the well-known UC Merced land-use image classification dataset without underfitting or overfitting. This accuracy is competitive with that achieved by many advanced states of the art architectures on the same dataset. This shows that putting time and effort into the correct utilisation and configuration of simple features is always worth consideration before pursuing complex and computationally expensive approaches. Developing and implementing such simple architectures would be particularly useful in land use classification in developing countries and/or municipalities with limited budgets and less powerful systems with memory constraints.

Keywords - Convolutional Neural Network, Image Classification, Deep Learning, Land Use Classification, Remote Sensing

Article History: Received 14 January 2022, Accepted 25 March 2022, Published 31 March 2022

How to cite: Dwivedi, D.N, and Patil, G (2022). Lightweight Convolutional Neural Network for Land Use Image Classification. Journal of Advanced Geospatial Science & Technology, 2(1), 31-48.

1. Introduction

With the ever-growing world population, industrialisation, and globalisation, land has become one of the most sought-after natural resources worldwide. The amount of land we have at our disposal is constant. Moreover, not every piece of land can be used for all purposes. For example, a certain land quality can be used only for agriculture. At the same time, there is another area of land where it's not possible to conduct agricultural activities at all. However, human beings have started exploiting the land for their short-term conveniences over the past century. For instance, much agricultural and forest land has been diverted for human habitats and industries to satisfy the needs of exponentially expanding urban areas. Such high scale changes in land use have a significant impact on the climate and are detrimental to society. Therefore, we as human beings must make the most efficient use of the land in the best possible manner.

One of the prerequisites for effective management of any entity is efficient measurement and monitoring. This applies to land management too. If we want to avoid the misuse of the land, the very first requirement is to classify it appropriately based on its current use. In land-use classification, the land imagery is classified into different classes based on the spectral features of the associated pixels. An accurate land use classification is helpful for effective deforestation management, well-planned irrigation, crop monitoring, water management, and many more.

Traditionally, municipalities and governments worldwide keep track of land use manually. Manual classification is cumbersome and time-consuming, with low accuracy in regularly updating the records. Recent advances in deep learning, computer vision, and satellite imagery availability for free to the public provide a promising prospect for land use classification (Alhassan et al. 2019; Baamonde et al. 2019; Liang et al. 2020; Zhang et al. 2021). This paper implements a lightweight CNN architecture to classify the overhead land images into various classes. CNN is one of the dominant network architectures gaining popularity in image processing, video processing, natural language processing, and many other applications. Particularly in image processing CNN has proven to be extremely efficient than regular neural networks and other complex image processing processes (Naranjo-Torres et al. 2020; Mohapatra, Swarnkar, and Das 2021; Wambugu et al. 2021).

The main contribution of this paper is to show how a simple CNN with a combination of properly tuned convolutional and max-pooling layers, improved normalisation, and a dropout concept can be implemented to achieve significantly higher accuracy than expensive state-of-the-art techniques in aerial land image classification using the same data (Hu et al. 2015; Penatti et al. 2015; Castelluccio et al. 2017).

2. Literature Review

CNN is one of the most powerful deep learning architectures used for efficient image processing. It processes the image pixels using different filters and tries to learn the image characteristics. It achieves this using image convolutions, pooling and fully connected neural network layers. Section 3 of this paper discusses the detailed architecture of a typical CNN. CNN and its various forms have become a de facto architecture for various computer vision-related applications (Zeiler and Fergus 2014; Sultana, Sufian, and Dutta 2018; Samudrala et al. 2021). The use of CNN spans across domains and industries. CNN has been proven effective in medical imaging (Yamashita et al. 2018; Lundervold, Alexander, and Lundervold 2019), health informatics (Daniele et al. 2017), change detection of deforestation (De Bem et al. 2020), vegetation remote sensing (Kattenborn et al. 2021), scene change detection (Sakurada and Okatani 2015), waste disposal systems (Haque et al. 2019) and many more. One of the most revolutionary works on image classification using CNN is a research paper on ImageNet data classification (Krizhevsky, Sutskever, and Hinton 2017) by researchers at the University of Toronto. The proposed model that is later widely referred to as “AlexNet” uses a convolutional neural network for effective image classification. After that, researchers have proposed many complex formulations of various deep learning techniques and model configuration ideas to improve the accuracy of image classification models.

The deep neural networks often tend to overfit during image classification and other computer vision tasks. A lot of research has undergone and is still in progress to develop the methods like dropout layer (Srivastava et al. 2014) and data augmentation (Shorten and Taghi 2019) to avoid overfitting in using CNN. Using transfer learning (Weiss, Taghi, and Wang 2016) that utilises already trained networks on large but nonrelated datasets and fine-tuning them to the target dataset has shown some encouraging results. The pre-trained architectures like Caffenet and GoogLenet (Castelluccio et al. 2017) have significantly improved image classification accuracy. UC Merced data set has been widely used in developing models for land use classification. The dataset was created by Yi Yang and Shawn Newsam for their research (Yang and Newsam 2010) back in 2010. Since then, various modelling techniques have been tried and tested using this dataset. Researchers (Hu et al. 2015; Fan, Chen, and Lu 2017) have used this dataset to train an unsupervised feature learning model. In one of the papers (Liang et al. 2020), researchers have developed a constrained extreme learning classifier using the same dataset. Furthermore, researchers (Marmanis et al. 2016) have demonstrated the use of pre-trained models on large datasets like ImageNet to improve the classification accuracy of the CNN model on UC Merced data. The paper by (Castelluccio et al. 2017) nicely lists various modelling techniques and their accuracy in land use classification using UC Merced data.

Apart from using aerial imagery for land use classification, with the availability of a large number of satellite images publicly, the researchers have applied the machine learning models to remote sensing data obtained from satellites like Sentinel II (Baamonde et al. 2019; Xie et al. 2019) and Landsat (Hu et al. 2015; Alhassan et al. 2019). While extensive remote sensing data has opened many possibilities, one of the significant challenges with training models is the data labelling with the ground truth. Researchers have been exploring the possibilities of creating labelled datasets (Helber et al. 2018) most efficiently so that more complex models can be trained for accurate land use classification.

3. Convolutional Neural Networks (CNN)

Deep CNN has become a dominant and most widely acceptable model architecture for image processing and computer vision in the recent past. The success of CNN can be attributed to its capability to extract highly granular features from the images using convolutions and pooling. The typical architecture of CNN is shown in Figure 1.

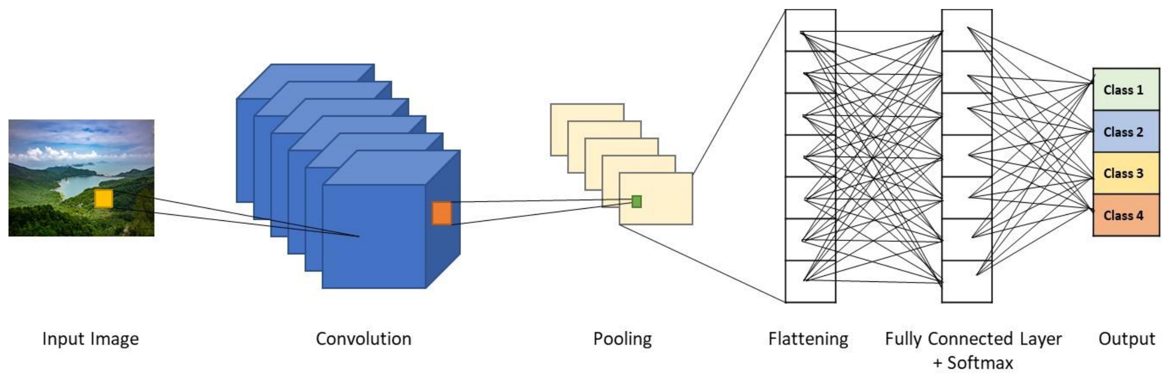

Figure 1. Typical CNN architecture.

In simplistic terms, the CNN contains the following three layers:

I. Convolution layer

In the convolution layer, the model applies a filter (also known as the kernel) of a specific size on the input image (generally a grid of pixels), producing a feature map that indicates the strength and location of the feature in an image. This allows the model to learn the image features like edges, texture, depth, brightness etc., anywhere in the image. A sample example of the first two convolution steps is shown in Figure 2.

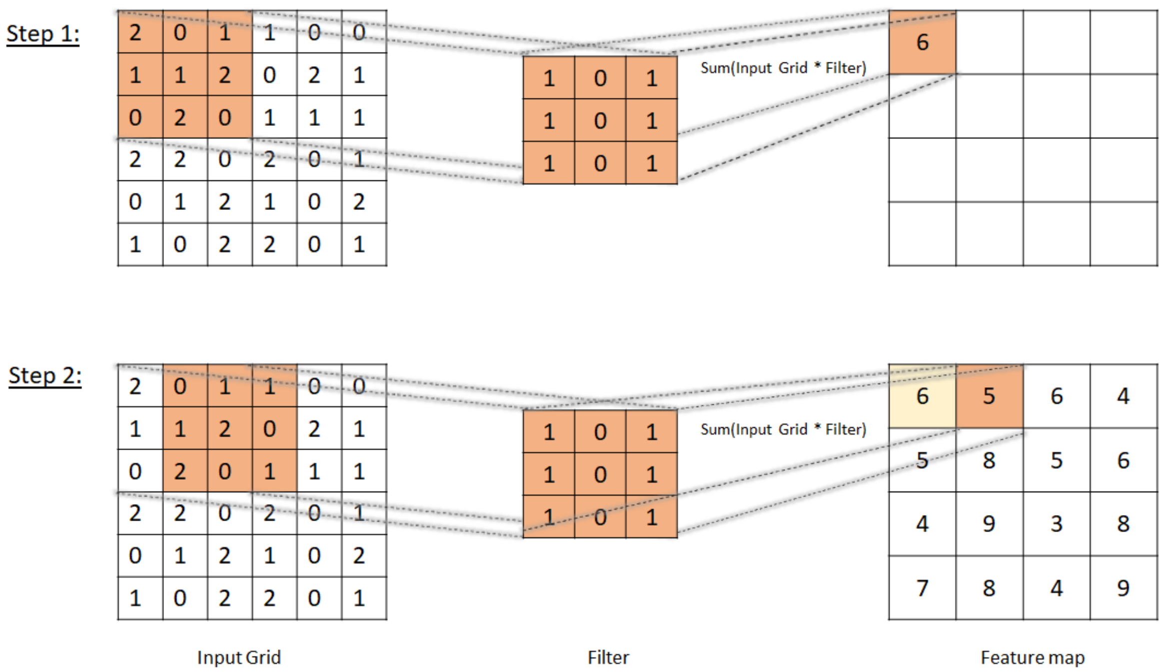

Figure 1. Illustration of convolution.

Odd numbers like 1X1, 3X3, 5X5 and 7X7 are typically used for the filter sizes. The filter size depends on the size of the image being processed. The linear output of each convolutional layer is converted into the non-linear one by a suitable activation function. The choice of activation function is particular to the context of the underlying problem. The most used activation functions are Sigmoid, Rectified Linear Unit (ReLU) and hyperbolic tangent (tanh).

II. Pooling layer

The pooling layer generally follows the convolution layer. The primary purpose of the pooling layer is to downsample the output of the convolution layer, hence avoiding overfitting and reducing the total number of parameters to learn. A sample example of max-pooling is shown in Figure 3.

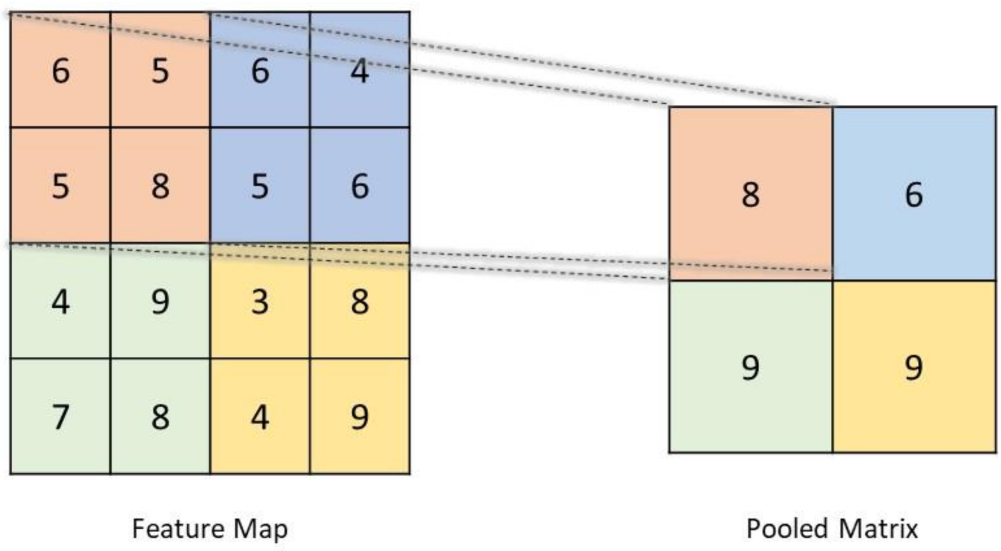

Figure 2. Illustration of pooling process.

III. Fully Connected layer

This layer is similar to the superficial neural network layer. Each input is connected to each output that maps the features extracted by earlier layers to the output categories. The final layer generally has the same number of nodes as the output categories. Moreover, the activation function used in the last layer depends on the type of model being built. The 'SoftMax' function is typically used for a multiclass classification problem, while the 'sigmoid' function is used for binary classification problems.

While training the model, the kernel and the weights of various layers are updated by observing the loss function on the training data to minimise the loss in each epoch. The optimisation algorithm used for each epoch also depends on the trained model. The most used optimisers for image processing using CNN are Adam optimisers or RMSprop.

4. Dataset

The "UC Merced Land Use" dataset (Yang and Newsam 2010) has been used for this study. The dataset consists of 2100 images, each of resolution 256X256, extracted from large images from the USGS National Map Urban Area Imagery collection for various urban areas around the USA. The images are labelled into 21 different land use categories, which include:

Table 1. Land use categories.

Category #	Category Name
0	Agriculture
1	Airplane
2	Baseball Diamond
3	Beach
4	Buildings
5	Chaparral
6	Dense Residential
7	Forest
8	Freeway
9	Golf Course
10	Harbor
11	Intersection
12	Medium Residential
13	Mobile Home Park
14	Overpass
15	Parking lot
16	River
17	Runway
18	Sparse Residential
19	Storage Tanks
20	Tennis Court

One example of every land use category is shown in Figure 4 below:

(1) Agriculture (2) Airplane (3) Baseball Diamond (4) Beach
(5) Buildings (6) Chaparral (7) Dense Residential (8) Forest
(9) Freeway (10) Golf Course (11) Harbor (12) Intersection
(13) Medium Residential (14) Mobile Home park (15) Overpass (16) Parking lot
(17) River (18) Runway (19) Sparse Residential (20) Storage Tanks
(21) Tennis Court

Figure 4. Land use categories.

5. CNN Implementation

It is well known to the practitioners of deep neural networks that, while it is easy to define the deep neural network, it is hard and time-consuming to configure them with the right set of parameters. This paper configures a simple CNN using a minimum number of layers and training parameters.

CNN works better for Land Use classification because images have high dimensionality, which can be handled exceptionally efficiently using CNN. This is achieved by convolution operations using different filters of smaller sizes which learn about the image in patches. Since Land Use images have different categories and each category has multiple image structures learning through convolutions helps in improving the accuracy of the entire process. All the layers of a CNN have multiple convolutional filters working and scanning the complete feature matrix and carrying out the dimensionality reduction. The modelling process as a whole can be summarised in Figure 5 below:

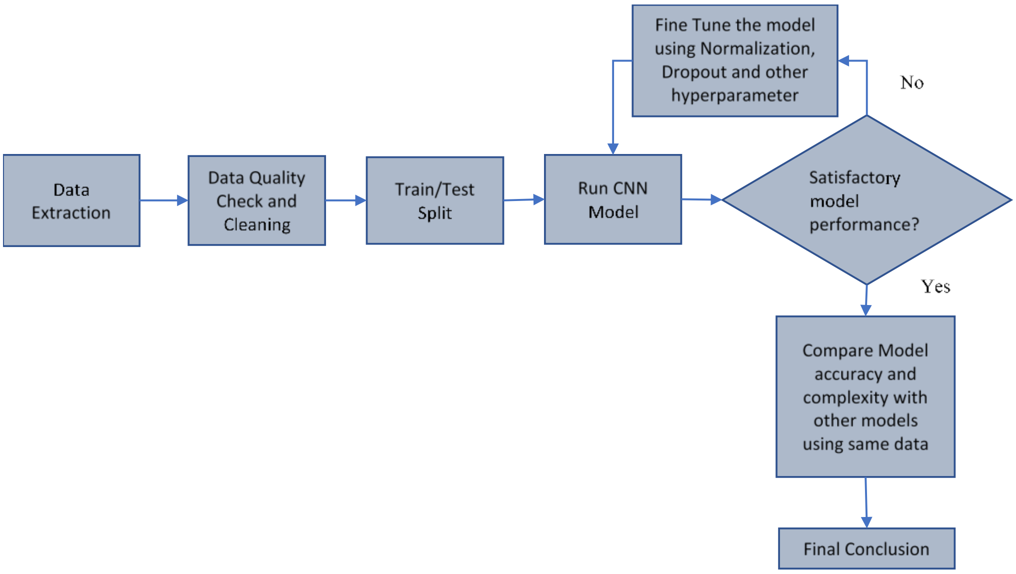

Figure 5. Modelling approach.

First, the 100 images from each class were divided into a training dataset of 90 images and a test dataset of 10 images. Every 10th image for the class was selected for the test dataset. Few images that were not of the desired size (256X256 pixels) were deleted from the training and test datasets. Different combinations of convolutional network layers were tried to reach optimum accuracy. The final CNN had the following layers:

- Five convolution layers, each with 3 X 3 kernel and ReLU non-linear activation function
- Five max-pooling layers of dimension 2X2
- Five dropout layers with a 50% dropout rate
- One fully connected layer
- Final output layer with 21 categorical nodes and Softmax activation function

Changing the filter size after a few layers while moving across the architecture allows the model to capture more diverse image features and keep the trainable parameters under control. To avoid model overfitting, the following two approaches were utilised:

I. Normalisation

Normalisation refers to reducing the actual input values to a smaller scale so that it becomes easy and less time consuming to derive the optimal solution. To normalise the model inputs, the pixel grid of each image was recomputed by subtracting the mean pixel value of that image. Furthermore, each pixel value was divided by 255.

II. Dropout

'Dropout' is a widely used approach to avoid overfitting in a deep neural network with a minimal increase in computational cost. The method was famously introduced by a team of researchers from the University of Toronto in their paper (Srivastava et al. 2014). In dropout, few neural connections were randomly dropped during each training epoch. This prevents the network from learning too much about one image and its features and forces it to learn new features, avoiding model overfitting.

Finally, the 'categorical cross entropy' loss function was used with the 'adam' optimiser for 200 epochs in 50 each to train the above-defined CNN architecture.

The summary of the final implemented CNN architecture is shown in Figure 6 below:

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_64 (Conv2D)           (None, 254, 254, 32)      896
_________________________________________________________________
max_pooling2d_59 (MaxPooling (None, 127, 127, 32)      0
_________________________________________________________________
dropout_51 (Dropout)         (None, 127, 127, 32)      0
_________________________________________________________________
conv2d_65 (Conv2D)           (None, 125, 125, 32)      9248
_________________________________________________________________
max_pooling2d_60 (MaxPooling (None, 62, 62, 32)        0
_________________________________________________________________
conv2d_66 (Conv2D)           (None, 60, 60, 64)        18496
_________________________________________________________________
max_pooling2d_61 (MaxPooling (None, 30, 30, 64)        0
_________________________________________________________________
dropout_52 (Dropout)         (None, 30, 30, 64)        0
_________________________________________________________________
conv2d_67 (Conv2D)           (None, 28, 28, 64)        36928
_________________________________________________________________
max_pooling2d_62 (MaxPooling (None, 14, 14, 64)        0
_________________________________________________________________
dropout_53 (Dropout)         (None, 14, 14, 64)        0
_________________________________________________________________
conv2d_68 (Conv2D)           (None, 12, 12, 64)        36928
_________________________________________________________________
max_pooling2d_63 (MaxPooling (None, 6, 6, 64)          0
_________________________________________________________________
dropout_54 (Dropout)         (None, 6, 6, 64)          0
_________________________________________________________________
flatten_5 (Flatten)          (None, 2304)              0
_________________________________________________________________
dense_10 (Dense)             (None, 128)               295040
_________________________________________________________________
dropout_55 (Dropout)         (None, 128)               0
_________________________________________________________________
dense_11 (Dense)             (None, 21)                2709
=================================================================
Total params: 400,245
Trainable params: 400,245
Non-trainable params: 0
_________________________________________________________________
```

Figure 6. Summary of final CNN architecture.

6. Results

The results of training the CNN architecture described above were measured using the model's classification accuracy on training and test datasets. The model was run using different parameters and in different batch sizes and several epochs. It was observed that the model accuracy does not improve significantly for any increase in the number of epochs post 200 epochs. The final model accuracy results are as shown in Table 2 below:

Table 2. Model results on training and testing datasets.

Dataset	Accuracy
Train	94.87%
Test	88.29%

The confusion matrix is shown in Figure 7 below:

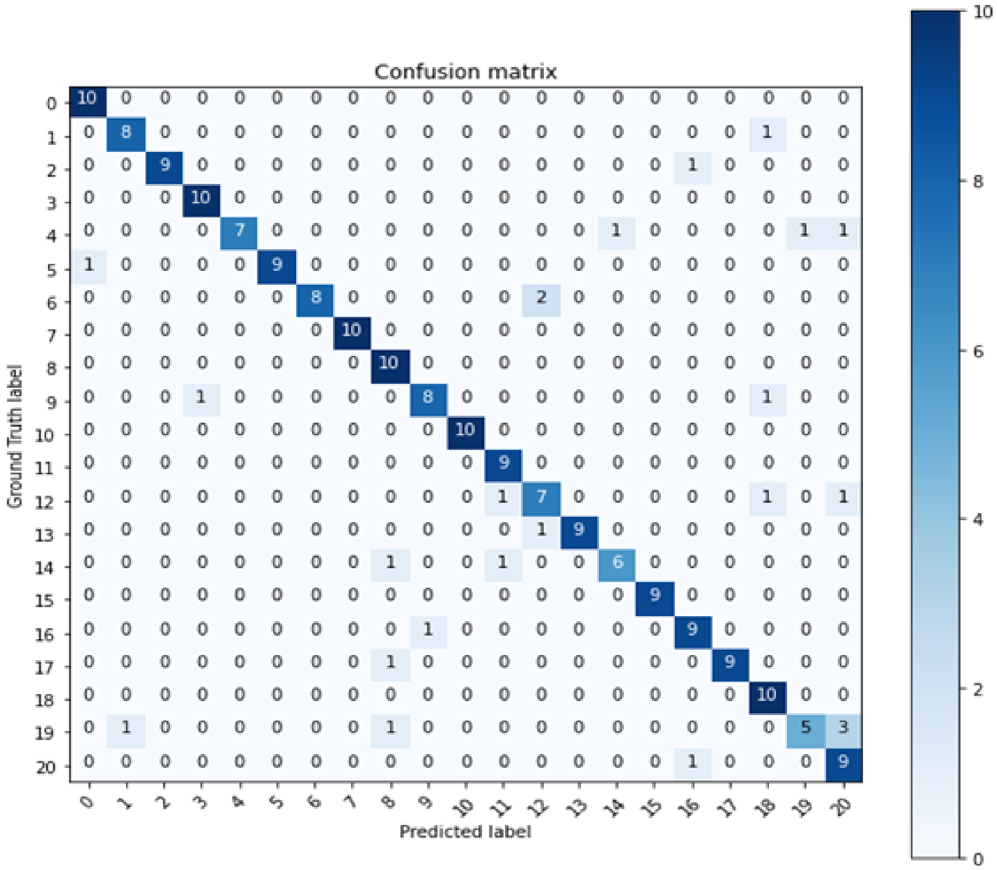

Figure 7. Confusion matrix.

The class numbers on the X and Y axes represent the corresponding categories as specified in Table 1 under the Dataset section above. It is worth noting that most of the classes have very high classification accuracy except for class ‘overpass’ and ‘storage tanks’, where the accuracy is around 50-60%.

The learning rate curves using loss rate and accuracy (on both training and validation sample) are shown below in Figure 8 and Figure 9, respectively:

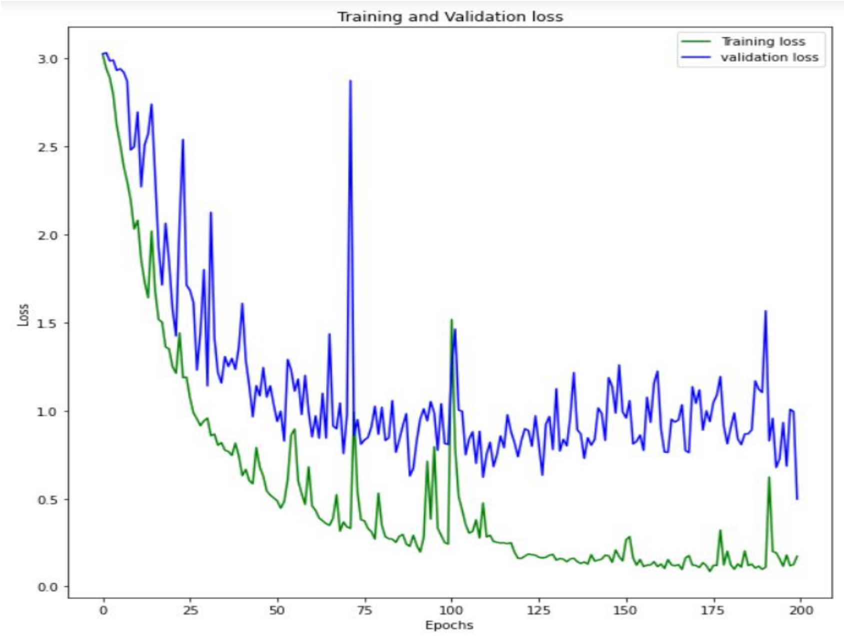

Figure 8. Training and validation loss rates.

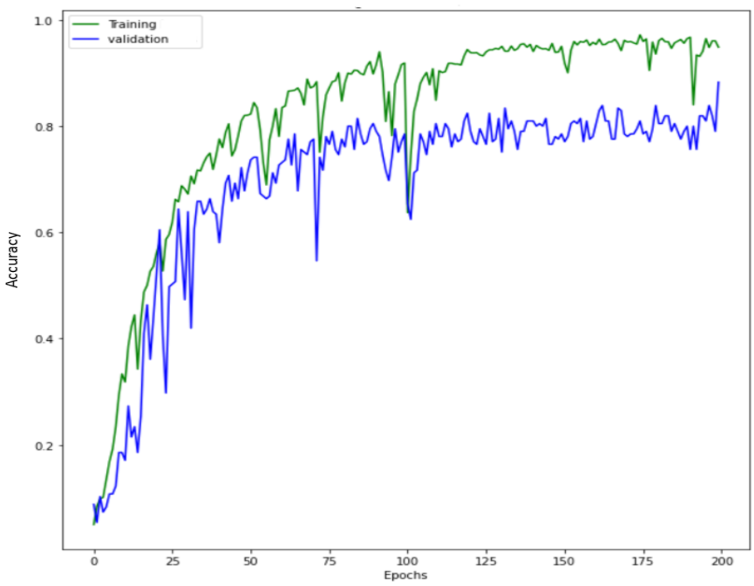

Figure 9. Training and validation accuracy.

The learning curves above clearly indicate that the loss rate decreases with more epochs, and the model performs equally well on training and testing datasets without underfitting or overfitting.

Considering the context of Land Use classification, the limited data size used, and the acceptable classification rate in most classes, the overall accuracy of 88.29% on this image classification can be deemed satisfactory.

Based on the findings of other research papers using the same data but different modelling techniques, the accuracy achieved by various key frontrunning architectures on UC Merced data is shown in the table below:

Table 2: Accuracy benchmarking.

Model	Accuracy
Pretrained GoogLeNet with fine tuning (Castelluccio et al. 2017)	97.10%
Pretrained CaffeNet followed by SVM classifier (Penatti et al. 2015)	93.42%
Unsupervised learning model (Hu et al. 2015)	90.26%
Proposed lightweight model	88.29%

While the accuracy of the proposed lightweight model is a few percentage points lower than some complex architectures, it is still in an acceptable range. It provides a computationally efficient alternative in scenarios where the development and deployment of such heavy models are not possible. The main gap with the other complex studies is that they utilise complex architectures that require a significant amount of time for development and large resources for implementation, limiting their widespread adoption. It is worth noting in the model summary above (Figure 5) that the proposed CNN architecture has only ~400K trainable parameters compared to millions of trainable parameters required in complex architectures like GoogLeNet, CaffeNet, ResNet etc. Moreover, from the available comparison data, the model using pre-trained GoogLeNet with fine-tuning (Castelluccio et al. 2017) was developed after 20K iterations which is 100 times the number of iterations used in the proposed simple CNN architecture.

Development and implementation of such models are instrumental in land use classification in developing countries and/or smaller municipalities with limited budgets and less powerful systems with memory constraints.

7. Conclusion

This paper highlights the possibility and importance of simple and lighter CNN vis-a-vis complex architectures. It demonstrates that using simple techniques like relevant filtering, better normalisation, and accurate placement of dropouts can achieve significantly high accuracy in less time and with lower computational cost. On Land Use data from UC Merced, the overall competitive accuracy of 88.29% (without over or underfitting) was achieved using a much simpler architecture with significantly less trainable parameters than complex architectures solving a significant issue of complexity and resource requirement by other complex networks.

Simple architectures provide an acceptable tradeoff between the model accuracy and ease of development and implementation, highlighting that there is a simpler, lightweight alternative for every complex architecture that is worth considering. Such a well-designed yet simple CNN provide excellent prospects for scenarios where the time and computational resources are limited. Especially in the applications where accuracy can be traded off by a few percentage points with the faster model development and implementation on less powerful computing engines.

Acknowledgement

The data used for this research is publicly available and free to use at the University of California, Merced website http://weegee.vision.ucmerced.edu/datasets/landuse.html.It was generated as part of the research paper by Yi Yang and Shawn Newsam (Yang and Newsam 2010). Special thank the authors for making this data publicly available.

References

Alhassan, V., Henry, C., Ramanna, S. and Storie, C., 2020. *“A deep learning framework for land-use/land-cover mapping and analysis using multispectral satellite imagery.”* Neural Computing and Applications, 32(12): 8529-8544. https://doi.org/10.1007/s00521-019-04349-9

Baamonde, S., Cabana, M., Sillero, N., Penedo, M.G., Naveira, H. and Novo, J., 2019. *“Fully automatic multi-temporal land cover classification using Sentinel-2 image data”.* Procedia Computer Science, 159: 650-657. https://doi.org/10.1016/j.procs.2019.09.220

Castelluccio, M., Poggi, G., Sansone, C. and Verdoliva, L., 2017. *“Training convolutional neural networks for semantic classification of remote sensing imagery.”* In 2017 Joint

Urban Remote Sensing Event (JURSE) (pp. 1-4). IEEE. https://doi.org/10.1109/jurse.2017.7924535

De Bem, P.P., de Carvalho Junior, O.A., Fontes Guimarães, R. and Trancoso Gomes, R.A., 2020. *"Change detection of deforestation in the Brazilian Amazon using landsat data and convolutional neural networks."* Remote Sensing, 12(6): 901. https://doi.org/10.3390/rs12060901

Fan, J., Chen, T. and Lu, S., 2017. *"Unsupervised feature learning for land-use scene recognition."* IEEE Transactions on Geoscience and Remote Sensing, 55(4): 2250-2261. https://doi.org/10.1109/tgrs.2016.2640186

Helber, P., Bischke, B., Dengel, A. and Borth, D., 2018, July. *"Introducing eurosat: A novel dataset and deep learning benchmark for land use and land cover classification."* In IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium (pp. 204-207). IEEE. https://doi.org/10.1109/igarss.2018.8519248

Hu, F., Xia, G.S., Wang, Z., Huang, X., Zhang, L. and Sun, H., 2015. *"Unsupervised feature learning via spectral clustering of multidimensional patches for remotely sensed scene classification"*. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 8(5). https://doi.org/10.1109/jstars.2015.2444405

Hu, Y., Zhang, Q., Zhang, Y. and Yan, H., 2018. *"A deep convolution neural network method for land cover mapping: A case study of Qinhuangdao, China."* Remote Sensing, 10(12): 2053. https://doi.org/10.3390/rs10122053

Kattenborn, T., Leitloff, J., Schiefer, F. and Hinz, S., 2021. *"Review on Convolutional Neural Networks (CNN) in vegetation remote sensing."* ISPRS Journal of Photogrammetry and Remote Sensing, 173: 24-49. https://doi.org/10.1016/j.isprsjprs.2020.12.010

Krizhevsky, A., Sutskever, I. and. Hinton, G.E. 2017. *"ImageNet Classification with Deep Convolutional Neural Networks."* Communications of the ACM, 60(6): 84–90. https://doi.org/10.1145/3065386

Kroupi, E., Kesa, M., Navarro-Sánchez, V.D., Saeed, S., Pelloquin, C., Alhaddad, B., Moreno, L., Soria-Frisch, A. and Ruffini, G., 2019. *"Deep convolutional neural networks for land-cover classification with Sentinel-2 images."* Journal of Applied Remote Sensing, 13(2): 024525. https://doi.org/10.1117/1.jrs.13.024525

Liang, J., Xu, J., Shen, H. and Fang, L., 2020. *"Land-use classification via constrained extreme learning classifier based on cascaded deep convolutional neural networks."* European Journal of Remote Sensing, 53(1): 219-232. https://doi.org/10.1080/22797254.2020.1809528

Lundervold, A.S. and Lundervold, A., 2019. *"An overview of deep learning in medical imaging focusing on MRI."* Zeitschrift für Medizinische Physik, 29(2): 102-127. https://doi.org/10.1016/j.zemedi.2018.11.002

Marmanis, D., Datcu, M., Esch, T. and Stilla, U., 2015. *"Deep learning earth observation classification using ImageNet pretrained networks."* IEEE Geoscience and Remote Sensing Letters, 13(1): 105-109. https://doi.org/10.1109/lgrs.2015.2499239

Mohapatra, S., Swarnkar, T. and Das, J., 2021. *"Deep convolutional neural network in medical image processing."* In Handbook of Deep Learning in Biomedical Engineering (pp. 25-60). Academic Press. https://doi.org/10.1016/b978-0-12-823014-5.00006-5

Naranjo-Torres, J., Mora, M., Hernández-García, R., Barrientos, R.J., Fredes, C. and Valenzuela, A., 2020. *"A review of convolutional neural network applied to fruit image processing."* Applied Sciences, 10(10): 3443. https://doi.org/10.3390/app10103443

Penatti, O.A., Nogueira, K. and Dos Santos, J.A., 2015. *"Do deep features generalise from everyday objects to remote sensing and aerial scenes domains?"* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 44-51). https://doi.org/10.1109/cvprw.2015.7301382

Ravì, D., Wong, C., Deligianni, F., Berthelot, M., Andreu-Perez, J., Lo, B. and Yang, G.Z., 2016. *"Deep learning for health informatics."* IEEE Journal of Biomedical and Health Informatics, 21(1): 4-21. https://doi.org/10.1109/jbhi.2016.2636665

Sakurada, K. and Okatani, T., 2015. *"Change Detection from a Street Image Pair using CNN Features and Superpixel Segmentation."* In BMVC (Vol. 61, pp. 1-12). https://doi.org/10.5244/c.29.61

Samudrala, S., Mohammad, S. and Vaddi, R., 2021. *"Change Detection in Land Use-Land Cover Using Convolutional Neural Network."* In International Conference on Communication, Computing and Electronics Systems (pp. 745-752). Springer, Singapore. https://doi.org/10.1007/978-981-33-4909-4_59

Shorten, C. and Khoshgoftaar, T.M., 2019. *"A survey on image data augmentation for deep learning."* Journal of Big Data, 6(1): 1-48. https://doi.org/10.1186/s40537-019-0197-0

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R., 2014. *"Dropout: a simple way to prevent neural networks from overfitting."* The Journal of Machine Learning Research, 15(1): 1929-1958.

Sultana, F., Sufian, A. and Dutta, P., 2018. *"Advancements in image classification using convolutional neural network."* In 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN) (pp. 122-129). IEEE. https://doi.org/10.1109/icrcicn.2018.8718718

Sunny, M.S.H., Dipta, D.R., Hossain, S., Faruque, H.M.R. and Hossain, E., 2019. *“Design of a convolutional neural network based smart waste disposal system.”* In 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT) (pp. 1-5). IEEE. https://doi.org/10.1109/icasert.2019.8934633

Wambugu, N., Chen, Y., Xiao, Z., Wei, M., Bello, S.A., Junior, J.M. and Li, J., 2021. *“A hybrid deep convolutional neural network for accurate land cover classification.”* International Journal of Applied Earth Observation and Geoinformation, 103: 102515. https://doi.org/10.1016/j.jag.2021.102515

Weiss, K., Khoshgoftaar, T.M. and Wang, D., 2016. *“A survey of transfer learning.”* Journal of Big data, 3(1): 1-40. https://doi.org/10.1186/s40537-016-0043-6

Xie, Q., Dash, J., Huete, A., Jiang, A., Yin, G., Ding, Y., Peng, D., Hall, C.C., Brown, L., Shi, Y. and Ye, H., 2019. *“Retrieval of crop biophysical parameters from Sentinel-2 remote sensing imagery.”* International Journal of Applied Earth Observation and Geoinformation, 80: 187-195. https://doi.org/10.1016/j.jag.2019.04.019

Yamashita, R., Nishio, M., Do, R.K.G. and Togashi, K., 2018. *“Convolutional neural networks: an overview and application in radiology.”* Insights Into Imaging, 9(4): 611-629. https://doi.org/10.1007/s13244-018-0639-9

Yang, Y. and Newsam, S., 2010. *“Bag-of-visual-words and spatial extensions for land-use classification.”* In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (pp. 270-279). https://doi.org/10.1145/1869790.1869829

Zeiler, M.D. and Fergus, R., 2014. *“Visualising and understanding convolutional networks.”* In European Conference on Computer Vision (pp. 818-833). Springer, Cham. https://doi.org/10.1007/978-3-319-10590-1_53

Zhang, Z., Cui, X., Zheng, Q. and Cao, J., 2021. *“Land use classification of remote sensing images based on convolution neural network.”* Arabian Journal of Geosciences, 14(4): 1-6. https://doi.org/10.1007/s12517-021-06587-5