Pertanika J. Sci. & Technol. 32 (6): 2609 - 2629 (2024)

PERTANIKA JOURNALS

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

Evaluating Convolutional Neural Network Architecture for Historical Topographic Hardcopy Maps Analysis: A Study on Training and Validation Accuracy Variation

Saiful Anuar Jaafar[1], Abdul Rauf Abdul Rasam[1,2] and Norizan Mat Diah[3]*

[1]Centre of Studies Surveying Science and Geomatics, College of Built Environment, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia
[2]Malaysia Institute of Transport (MITRANS), Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia
[3]School of Computing Sciences, College of Computing, Informatics and Mathematics, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia

ABSTRACT

Convolutional Neural Networks (CNN) are widely used for image analysis tasks, including object detection, segmentation, and recognition. Given the advanced capability, this study evaluates the effectiveness and performance of CNN architecture for analysing Historical Topographic Hardcopy Maps (HTHM) by assessing variations in training and validation accuracy. The lack of research specifically dedicated to CNN's application in analysing topographic hardcopy maps presents an opportunity to explore and address the unique challenges associated with this domain. While existing studies have predominantly focused on satellite imagery, this study aims to uncover valuable insights, patterns, and characteristics inherent to HTHM through customised CNN approaches. This study utilises a standard CNN architecture and tests the model's performance with different epoch settings (20, 40, and 60) using varying dataset sizes (288, 636, 1144, and 1716 images). The results indicate that the optimal operation point for training and validation accuracy is achieved at epoch 40. Beyond epoch 40, the widening gap between training and validation accuracy suggests overfitting. Hence, adding more epochs does not significantly improve accuracy beyond the optimum phase. The experiment also shows that the CNN model obtains a training accuracy of 98%, validation accuracy of 67%, and F1-score overall performance of 77%. The analysis demonstrates that the CNN model performs reasonably well in classifying instances from the HTHM dataset. These

ARTICLE INFO

Article history:
Received: 12 December 2023
Accepted: 10 July 2024
Published: 25 October 2024

DOI: https://doi.org/10.47836/pjst.32.6.11

E-mail addresses:
saifulanuar@uitm.edu.my (Saiful Anuar Jaafar)
rauf@uitm.edu.my (Abdul Rauf Abdul Rasam)
norizan289@uitm.edu.my (Norizan Mat Diah)
* Corresponding author

ISSN: 0128-7680
e-ISSN: 2231-8526

findings contribute to a better understanding of the strengths and limitations of the model, providing valuable insights for future research and refinement of classification approaches in the context of topographic hardcopy map analysis.

Keywords: Convolutional Neural Network (CNN), deep learning, feature map recognition, Historic Topographic Hardcopy Map (HTHM)

INTRODUCTION

The digitisation of archive collections by heritage and library institutions has led to the accessibility of vast amounts of historical data, including the topographic hardcopy maps. Existing methodologies in historical map analysis have traditionally relied on manual and semi-automatic procedures of feature extraction techniques for vectorisation. This process is often labour-intensive, time-consuming, and prone to subjectivity (Anuar et al., 2021). In contrast, CNNs offer the potential to automate and streamline the analysis process, especially in object detection. According to research in the field of CNNs, increasing the epoch number and the training dataset size can potentially enhance the accuracy of both training and validation (Althnian et al., 2021; Barry-Straume et al., 2018).

However, the influence of these factors on accuracy may be contingent upon various aspects, such as the specific dataset characteristics and the complexity of the problem at hand (Ali et al., 2021). The number of epochs plays a crucial role in the learning process of a CNN. By increasing the number of epochs, the model can iterate over the training dataset multiple times, enabling it to capture more intricate patterns and improve accuracy (Garbin et al., 2020; Kumar et al., 2024). Nonetheless, it is essential to strike a balance, as excessively high epoch numbers may lead to overfitting (Chauhan et al., 2018; Poojary et al., 2020). Overfitting occurs when the model becomes overly specialised in the training data and fails to generalise well to unseen data. Monitoring the validation accuracy during training is recommended to determine an optimal number of epochs. Once the validation accuracy plateaus or begins to decrease, further training may not yield substantial improvements (Dawson et al., 2023; Johny & Madhusoodanan, 2021).

The training dataset size also influences the CNN performance. Increasing the dataset size gives the model a more diverse set of examples, enhancing its ability to generalise and perform well on unseen data. However, ensuring that the dataset remains representative of the problem domain and encompasses an adequate range of variations and scenarios is essential. Acquiring or generating a more extensive dataset may entail additional costs and efforts, necessitating careful consideration of the available resources. Increasing the number of epochs and training dataset size generally positively impacts CNN accuracy (Kandel & Castelli, 2020). However, finding the optimal values requires empirical investigation and diligent monitoring of validation performance to prevent overfitting. Achieving the best possible accuracy necessitates carefully balancing model complexity, computational resources, and data availability.

Thus, the evaluation results were presented, highlighting the CNN architecture performance on the HTHM dataset. The relationship between epoch variation, dataset size, and training and validation accuracy was analysed and discussed. The findings provide insights into the effectiveness of the CNN architecture for analysing HTHM and offer guidance for determining the optimal epoch value and dataset size for achieving satisfactory performance. This study aims to test the CNN architecture on the Historical Topographic Hardcopy Map (HTHM) dataset; thus, the objectives of this study are:

1. To review the CNN model structure on trained Historical Topographic Hardcopy Map dataset.
2. To evaluate the training and validation accuracy by varying the epoch on different dataset amounts.

By explicitly comparing the CNN approach to existing methodologies, the study contributes to the growing body of literature on computational methods for historical map analysis, offering insights into the strengths and limitations of CNNs and providing guidance for future research in this area.

BACKGROUND STUDY

While numerous studies have focused on using satellite imagery as the dataset for CNN-based research, this study stands out by utilising topographic hardcopy maps as the dataset's domain. This novel approach introduces a unique perspective in applying CNN, exploring the potential of extracting valuable information and insights from traditional cartographic representations. By shifting the focus from satellite imagery to topographic hardcopy maps, this research opens new avenues for leveraging CNN in geospatial analysis. It contributes to a broader understanding of the digital transformation in cartography and spatial data analysis.

Based on the available literature, a significant body of research has utilised CNN for analysing satellite imagery as their primary dataset in various domains, including remote sensing and geospatial analysis (Bhosle & Musande, 2022; Li et al., 2021). These studies have demonstrated the effectiveness of CNN in extracting meaningful information and patterns from satellite images, leading to advancements in fields such as land cover classification, object detection, and change detection. Audebert et al. (2019) introduced a deep-learning approach for hyperspectral data classification using CNNs. The proposed framework surpasses the limitations of traditional methods by leveraging the power of CNNs to capture both spatial and spectral information. The results highlight the effectiveness of CNNs in enhancing hyperspectral data analysis through improved classification accuracy and better utilisation of the rich information in hyperspectral images. Chen et al. (2016) and Sharifi et al. (2022) proposed a CNN-based method for accurate hyperspectral image classification by leveraging its hierarchical representation learning capabilities. The study demonstrated the effectiveness of CNN in extracting discriminative features from complex hyperspectral data. This finding underscores the potential of CNN as a valuable tool for improving the analysis and classification of hyperspectral imagery. Ji et al. (2018) presented a 3D CNN-based method for deep feature extraction and classification of hyperspectral images. By harnessing CNN's hierarchical representation learning capabilities, the aim is to enhance the accuracy of hyperspectral image classification. Hamouda et al. (2020) demonstrated that smart feature extraction and classification of hyperspectral images using CNN improves classification accuracy while reducing computing time. Findings from Liu et al. (2020) demonstrate the effectiveness of CNN in extracting discriminative features from complex and high-dimensional hyperspectral data and focus on multi-label land cover classification using CNN for remote sensing images. The aim is to tackle the challenge of simultaneous classification of multiple land cover types from satellite imagery. The study achieved promising results by employing CNN to identify various land cover categories accurately. These findings highlight the potential of CNN in facilitating comprehensive land cover analysis in remote sensing applications.

For instance, Liu et al. (2020) and Dwivedi and Patil (2022) employed CNN for land cover classification using satellite imagery, achieving high accuracy in identifying different land cover classes. Guo et al. (2018) utilised CNN for object detection in satellite images, enabling the automated identification of specific objects, such as buildings, roads, and vegetation. Similarly, Li et al. (2020) employed CNN for change detection in satellite imagery, facilitating the identification of temporal changes in land cover over different periods. Based on this evidence, CNNs have gained popularity in Remote Sensing due to their effectiveness in handling various image analysis tasks.

However, concerning Geospatial and Digital Cartography (Geospatial Cartography), CNNs are also utilised for performing vectorisation through hardcopy maps, object classification, and image analysis. It highlights the versatile capabilities of CNNs in both Remote Sensing and Geographic Information Systems (GIS) domains, enabling the extraction of valuable information from various types of data sources and facilitating comprehensive spatial analysis. It is important to note that there appears to be limited research explicitly focusing on applying CNN in analysing topographic hardcopy maps as the dataset domain. While topographic maps are crucial in various fields, such as urban planning, environmental assessment, and infrastructure development, most existing studies have primarily focused on satellite imagery. Therefore, the study on CNN using topographic hardcopy maps as the dataset introduces a novel perspective to the field. By exploring the application of CNN in analysing topographic maps, the study will have the opportunity to address unique challenges and extract valuable insights specific to this domain. It includes identifying features, patterns, and characteristics inherent to topographic maps, which may require customised approaches for effective analysis and interpretation.

This study expands the scope of CNN applications in geospatial analysis and provides a valuable contribution to the field, focusing on topographic hardcopy maps. It fills a gap in the existing literature and opens avenues for further exploration, ultimately advancing the understanding and utilisation of topographic maps in various domains. The study also explores and identifies its limitations and areas for improvement. Analysing misclassifications reveals patterns or challenges the model struggles with, informing refinements in pre-processing, training data augmentation, and model architecture. Addressing these insights can enhance the model's accuracy and reliability for HTHM analysis.

METHODOLOGY

The HTHM dataset was utilised to train the CNN architecture. The selected CNN model underwent rigorous evaluation and analysis on the HTHM dataset to determine its effectiveness in extracting meaningful information from the maps. Various metrics, including training and validation accuracy, were assessed to quantify the CNN architecture performance. The impact of varying the epoch during training and utilising different dataset sizes was investigated. This analysis involved training the CNN model with different epoch values on subsets of the HTHM dataset and observing the corresponding training and validation accuracy changes. Epochs in this study refer to the number of times the entire training dataset is presented to the model during training. Setting epochs at intervals of 20 allowed the study to assess the model's performance early in training (20 epochs), at a mid-point (40 epochs), and after further training (60 epochs). This approach enabled the study to analyse how accuracy varied as the model underwent different stages of learning and whether additional training beyond a certain point yielded significant improvements or led to overfitting. By exploring different epoch durations and dataset sizes, valuable insights regarding optimal training conditions can be obtained.

The study expected that training and validation accuracy would improve by increasing the number of epochs and datasets. Thus, the outcomes of this paper were used to further improve the selection of the best architecture for implementing automatic vectorisation for HTHM. The following are the detailed steps of this research methodology: Data Collection, Data Pre-processing, CNN Training Model, Evaluation of Performance, and lastly, Result and Conclusion.

Data Collection

The scanned HTHM were collected from Perpustakaan Tun Abdul Razak at UiTM Shah Alam, specifically in the Mapping section. The study classified four objects: buildings, water bodies, land use, and roads. All the images were cropped on ten hardcopy map samples, which moderated conditions. The map was scanned using an A0 flatbed scanner with 500 dpi. All the datasets were cropped in dimensions of 224 pixels by 224 pixels (Figure 1) to ensure the dataset is of standard size. Figure 1 shows a sample of each object class in HTHM. The examples of every dataset class are shown in Table 1.

Table 1 and Figure 1 outline the composition of the dataset. The dataset focuses on map features, including contour building, land use, road and water bodies. Each dataset presents unique challenges for model generalisation. Variations in map feature complexity and quality may introduce bias and affect the model's ability to generalise across diverse conditions. The dataset may not fully represent the diversity of historical topographic hardcopy maps (HTHM) in terms of geographical regions, periods, or map styles. This limitation could result in the model being biased towards the characteristics of the included maps.

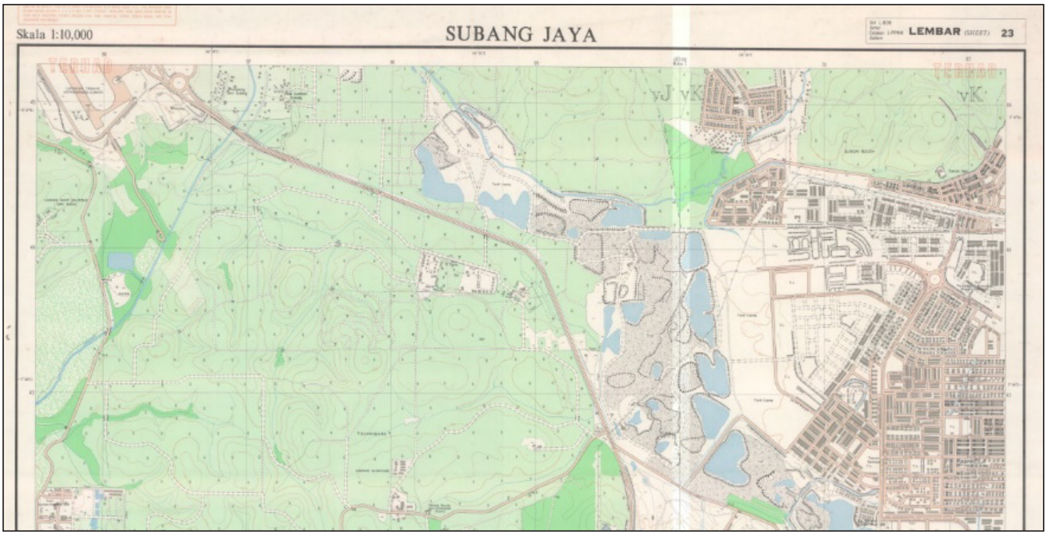


Figure 1. Sample of historical topographic hardcopy map

Table 1
Object in historical topographic hardcopy map

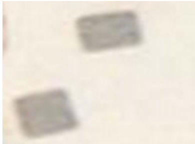
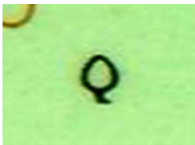
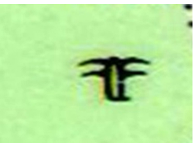
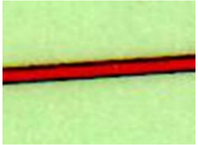
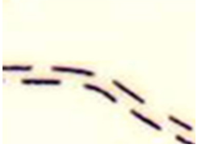
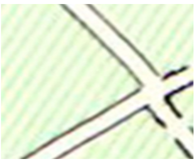
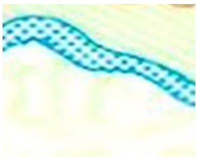
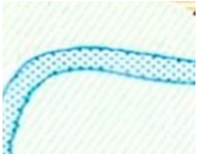
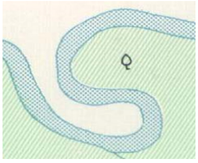
Dataset	Sample 1	Sample 2	Sample 3
Building	 Building 1	 Building 2	 Building 3
Land Use	 Rubber Tree Plantation	 Palm Oil Plantation	 Forest

Table 1 *(continue)*

Dataset	Sample 1	Sample 2	Sample 3
Road	 Main Road	 Small Road	 Small Road
Water Bodies	 River 1	 River 2	 River 3

Data Pre-processing

Four datasets in this study represent objects on hardcopy maps. Dataset 1 represents buildings, Dataset 2 represents land use, Dataset 3 represents roads, and Dataset 4 represents water bodies. While preparing the datasets, each image was augmented with the following techniques: rotations of 90°, 180°, 270°, flip vertical, and flip horizontal. Data augmentation can assist in lessening overfitting, a significant issue in Deep Learning, and enhancing model performance (Khalifa et al., 2022). All classes of objects were subjected to image augmentation. The specifics of the augmentation are shown in Figure 2.

The dataset was divided into a 70:20:10 ratio, with 70% of samples for training, 20% for validation, and 10% for model testing. The study performed five training sets, each with several datasets and epochs. The distribution of training sets is shown in Table 2.

Based on Table 2, experiments 1 to 4 undergo training using the CNN model at epochs 20, 40, and 60. Each training was tested for its capability of achieving training and validation accuracy. Experiment 1 used 288 images; experiment 2 used 636 images; experiment 3 used 1144 images; and experiment 4 used 1716 images.

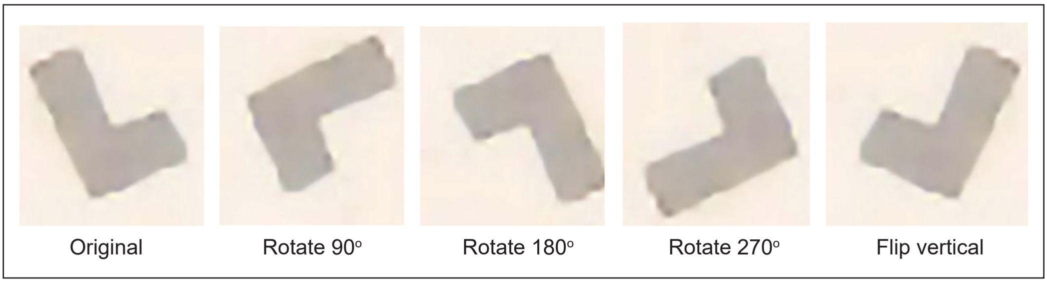


Figure 2. Example of augmentation image on building dataset

Table 2
Training set details

No. experiment	1	2	3	4
Epochs	20/40/60	20/40/60	20/40/60	20/40/60
Training data	200	444	800	1200
Validate data	60	128	228	344
Testing data	28	64	116	172
Total data set	288	636	1144	1716

The Proposed CNN Training Model

In this study, a CNN model was developed to detect the objects on HTTM that had already been scanned. Figure 3 shows the structure of the CNN model, and Table 3 displays the layers and details for each layer.

The model used an adopted technique from Roslan et al. (2023). The first layer in this model is a convolutional layer, calculating 16 features for each 3×3 kernel. The second

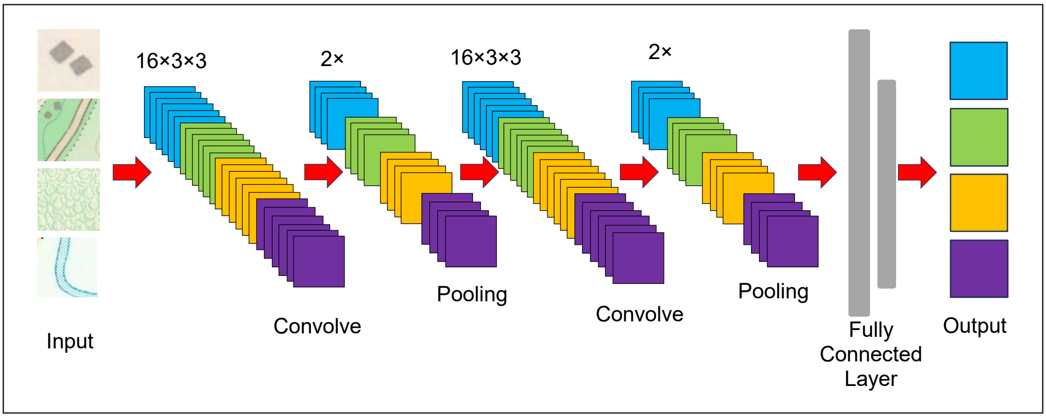


Figure 3. The proposed structure of the CNN model

Table 3
CNN layers in the proposed model

No	Layer (Type)	Layer type and filter shape
1	Conv2D	Convolution-ReLU, Kernel <16×3×3>
2	Pooling2D	Max pooling, 2×2
3	Batch_normalization	-
4	Conv2D_1	Convolution-ReLU, Kernel <32×3×3>
5	Pooling2D_1	Max pooling, 2×2
6	Batch_normalization_1	-
7	Flatten	Flatten
8	Dense	ReLU activation
9	Activation_5 (Softmax)	Classifier

layer is a max-pooling layer with a 2×2 kernel. In the next step, another convolutional layer calculates 32 features for each 3×3 kernel. It is followed by another max pooling with a 2×2 filter. Note that the batch normalisation was applied after each max-pooling layer, and the rectified linear unit (ReLU) served as an activation function. After four layers, a fully connected layer can be found. In the last phase, a SoftMax layer creates a vector with four entries from the proceeding layers' results vector. These four entries indicate the four types of objects in HTHM.

In this study, dropout was applied to hidden layers of the CNN architecture with varying dropout rates. By randomly masking a fraction of neurons during each training iteration, dropout helped prevent overfitting by ensuring that no single neuron or feature became overly reliant on specific input patterns. The impact of dropout regularisation was observed through improvements in validation accuracy and reduced overfitting tendencies, particularly as the model underwent additional training epochs. In the study methodology, optimising hyperparameters such as learning rate and batch size was essential to ensure the CNN model's optimal performance. The learning rate and batch sizes determine the step size taken during the optimisation process by analysing the learning curve on the graph to update the model's weights.

Evaluation of Performance

Each of the training sets underwent an accuracy evaluation. The observation was based on four elements, which are the results of this study.

Training and Validation Accuracy of the Model's Accuracy Graph Interpretation Analysis

Visualising the model's accuracy over epochs is valuable for understanding its performance and learning progress during training. It provides insights into how well the model fits the training data and its ability to generalise to unseen data. The benefits of visualising accuracy over epochs are as follows: First, it allows monitoring of the training progress by observing the accuracy trends over time. Steady increases or high plateaus indicate effective learning. Second, comparing training and validation accuracy helps detect overfitting or underfitting. A large gap suggests overfitting, while low values for both indicate underfitting. Third, it aids in determining convergence by identifying stable performance, where training and validation accuracy reach a plateau or show diminishing improvements. Finally, it facilitates model comparison by visualising the accuracy trends of multiple models, enabling the selection of the best-performing one based on convergence, generalisation, and overall accuracy (Alzubaidi et al., 2021).

Model Loss Graph Pattern

The loss function quantifies the disparity between the model's predicted and expected output to minimise this difference during training. Plotting the loss over epochs provides valuable information about the model's learning progress and convergence. The significance of the loss graph and its interpretation are as follows: First, it allows monitoring of the model's learning progress, with decreasing or plateauing loss indicating effective learning. Second, overfitting or underfitting can be identified by comparing the training loss with the validation loss. A significant decrease in training loss with high validation loss suggests overfitting, while high values for both indicate underfitting. Third, the loss plot helps determine if the model has converted to stable performance, as evidenced by a plateau or diminishing improvements in training and validation losses. By analysing the loss over epochs, valuable insights can be gained regarding the model's learning behaviour, issues like overfitting or underfitting, and an overall assessment of its convergence and performance.

Confusion Matrix

Confusion Matrix is an important measure to evaluate the accuracy of credit scoring models (Zeng, 2020). The confusion matrix is a comprehensive summary of a model's predictions and the actual labels of the data points, particularly in classification problems (Tharwat, 2020). It consists of a table where rows represent true labels and columns represent predicted labels. The matrix provides counts or proportions of true positives (correctly predicted positives), true negatives (correctly predicted negatives), false positives (incorrectly predicted positives), and false negatives (incorrectly predicted negatives). True positives and true negatives indicate correct predictions, while false positives and false negatives represent prediction errors. Analysing the confusion matrix helps identify the model's accuracy and the specific types of errors it makes, enabling targeted improvements.

Classification Report

Evaluation metrics, such as accuracy, precision, recall, and F1-score, are commonly used to measure a model's performance in predicting correct class labels. Accuracy represents the overall correctness of the predictions, precision measures the ability to identify positive instances correctly, recall measures the ability to identify positive instances out of all actual positives correctly, and the F1-score provides a balanced measure between precision and recall. Comparing predictions against a labelled dataset with known ground truth is crucial to assess model accuracy. These metrics help evaluate performance, identify areas for improvement, and inform decisions on model selection, hyperparameter tuning, and feature engineering. The accuracy, precision, recall, and F1 score will be calculated after implementing the model. It will use the confusion matrix, including true positive (TP), true negative (TN), false positive (FP), and false negative (FN), to measure accuracy, precision, recall, and F1-score. The formulas for accuracy, precision, recall, and F1-score are represented in Equations 1 to 4.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad [1]$$

$$Precision = \frac{TP}{TP+FP} \quad [2]$$

$$Recall = \frac{TP}{TP+FN} \quad [3]$$

$$F1\text{-}score = 2 \times \frac{Precision \ \ x \ Recall}{Precision \ +Recall} \quad [4]$$

EXPERIMENT RESULTS AND ANALYSIS

The experiment was conducted by performing data training based on several train datasets. The datasets were based on four classes: buildings, roads, water bodies, and land use. The images from these four classes totalled 288, 636, 1144, and 1716. These were tested on 20, 40, and 60 epoch variations to find the best training performance for the HTHM dataset. Each dataset was evaluated based on its model accuracy, model loss, confusion matrix, and classification report as the model's performance indicator. The results are shown below:

Experiment 1: 288 Data Set

In Table 4, the CNN model's results show an improvement in accuracy as the number of epochs increases. At epoch 20, the model achieved a training accuracy of 69% and a validation accuracy of 45%. By epoch 40, the training accuracy had significantly improved to 97.5%, with a validation accuracy of 48.33%. Concerning epoch 60, the training accuracy drops to 87.5%, while the validation accuracy drops to 41.67%.

Table 5 shows the validation accuracy of the model's accuracy graph for experiment 1. The confusion matrix and Classification report for HTHM data are tabulated in Table 6. The total image of a confusion matrix for testing data is 28, 10% for data testing from 288

Table 4
Result of performance loss, accuracy, validation loss, and validation accuracy achieved for Experiment 1

288 data set				
Epoch	**Loss**	**Accuracy**	**Val_loss**	**Val_accuracy**
20	0.9605	0.6900	1.2575	0.4500
40	0.1490	0.9750	1.9651	0.4833
60	0.3450	0.8750	2.4341	0.4167

images. Table 7 shows that epoch 40 achieved high accuracy by testing average precision, recall F1 score and accuracy.

Table 5
Validation accuracy of the model's accuracy graph for Experiment 1

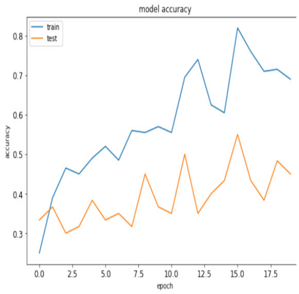
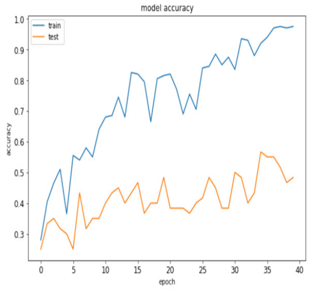
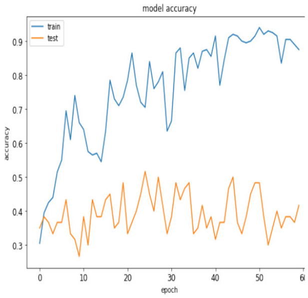
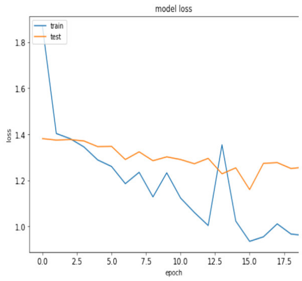
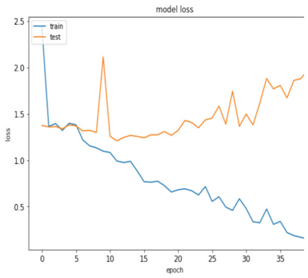
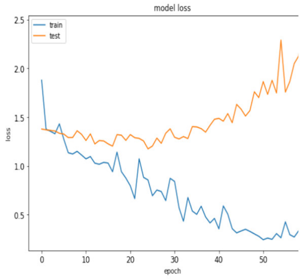
Epoch	20	40	60
Model accuracy			
Model loss			

Table 6
Result of the performance confusion matrix and classification report for the testing model Experiment 1

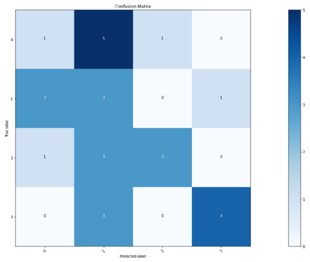
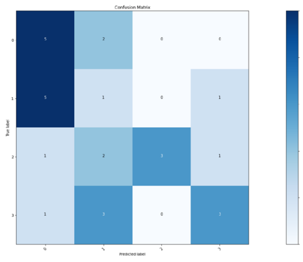
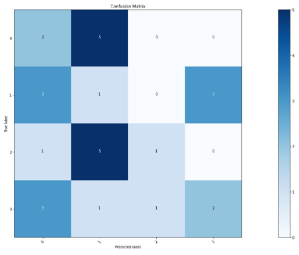
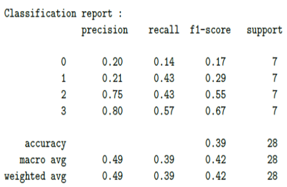
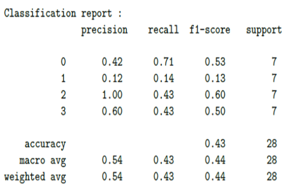
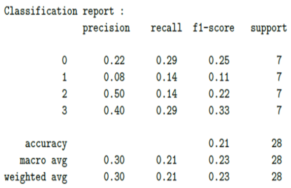
Epoch	20	40	60
Confusion matrix			
Classification report	Classification report : precision recall f1-score support 0 0.20 0.14 0.17 7 1 0.21 0.43 0.29 7 2 0.75 0.43 0.55 7 3 0.80 0.57 0.67 7 accuracy 0.39 28 macro avg 0.49 0.39 0.42 28 weighted avg 0.49 0.39 0.42 28	Classification report : precision recall f1-score support 0 0.42 0.71 0.53 7 1 0.12 0.14 0.13 7 2 1.00 0.43 0.60 7 3 0.60 0.43 0.50 7 accuracy 0.43 28 macro avg 0.54 0.43 0.44 28 weighted avg 0.54 0.43 0.44 28	Classification report : precision recall f1-score support 0 0.22 0.29 0.25 7 1 0.08 0.14 0.11 7 2 0.50 0.14 0.22 7 3 0.40 0.29 0.33 7 accuracy 0.21 28 macro avg 0.30 0.21 0.23 28 weighted avg 0.30 0.21 0.23 28

Table 7
Average precision, average recall, average F1-score, and accuracy for the testing model in Experiment 1

288 data set				
Epoch	**average precision**	**average recall**	**average F1-score**	**Accuracy**
20	0.49	0.39	0.42	0.39
40	0.54	0.43	0.44	0.43
60	0.30	0.21	0.23	0.21

Experiment 2: 636 Data Set

The trained datasets were set to 636 images, and the implemented CNN architecture results are shown in Table 8.

In Table 8, the results of the CNN model show an improvement in accuracy as the number of epochs increases. At epoch 20, the model achieved a training accuracy of 81% and a validation accuracy of 46%. By epoch 40, the training accuracy had significantly improved to 98%, with a validation accuracy of 51.56%. Concerning epoch 60, the training accuracy peaks at 100%, while the validation accuracy drops at 50.78%. The model accuracy graph was shown in Tables 9 and 10 for the confusion matrix with its classification report.

The confusion matrix and classification report for HTHM data are tabulated in Table 11. The total image of a confusion matrix for testing data is 64, 10% for data testing from 636 images.

Table 8
Result of performance loss, accuracy, validation loss, and validation accuracy achieved for Experiment 2

636 data set				
Epoch	**Loss**	**Accuracy**	**Val_loss**	**Val_accuracy**
20	0.4528	0.8176	1.7168	0.4609
40	0.1169	0.9797	3.5790	0.5156
60	0.0054	1.0000	3.4112	0.5078

Table 9
Validation accuracy of the model's accuracy graph for Experiment 2

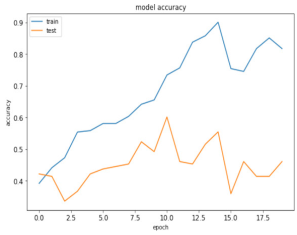
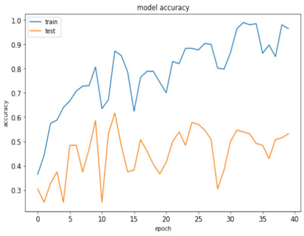
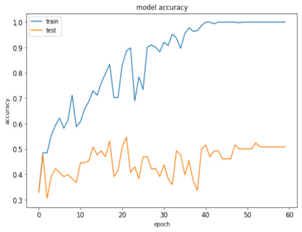
Epoch	20	40	60
Model accuracy			

Table 9 *(continue)*

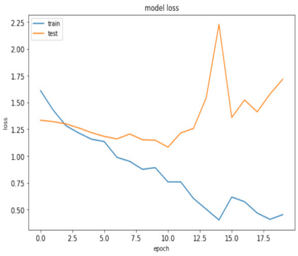
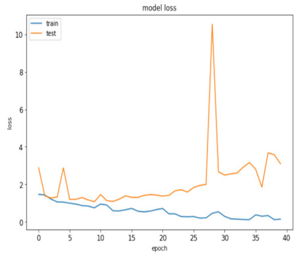
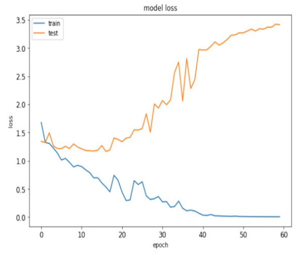
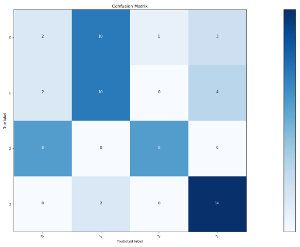
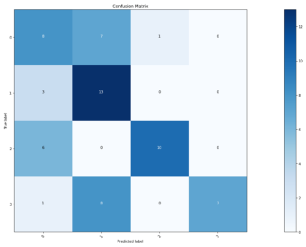
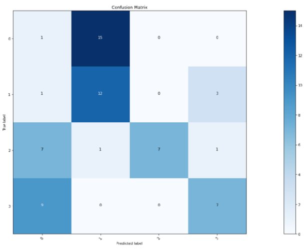
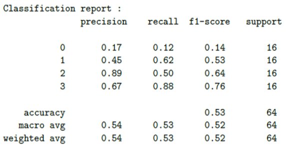
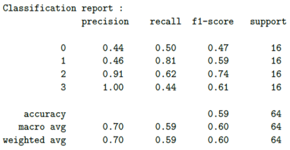
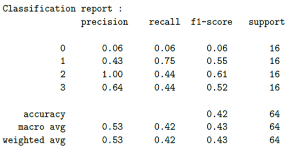
Epoch	20	40	60
Model loss			

Table 10
Result of the performance confusion matrix and classification report for the testing model Experiment 2

Epoch	20	40	60
Confusion matrix			
Classification report			

Confusion Matrix
True label
Predicted label
0
1
2
3
2
10
1
3
2
10
0
4
8
0
8
0
0
2
0
14

Confusion Matrix
True label
Predicted label
0
1
2
3
8
7
1
0
3
13
0
0
6
0
10
0
1
8
0
7
12
10
8
6
4
2
0

Confusion Matrix
True label
Predicted label
0
1
2
3
1
15
0
0
1
12
0
3
7
1
7
1
9
0
0
7
14
12
10
8
6
4
2
0

```
Classification report :
              precision    recall  f1-score   support

           0       0.17      0.12      0.14        16
           1       0.45      0.62      0.53        16
           2       0.89      0.50      0.64        16
           3       0.67      0.88      0.76        16

    accuracy                           0.53        64
   macro avg       0.54      0.53      0.52        64
weighted avg       0.54      0.53      0.52        64
```

```
Classification report :
              precision    recall  f1-score   support

           0       0.44      0.50      0.47        16
           1       0.46      0.81      0.59        16
           2       0.91      0.62      0.74        16
           3       1.00      0.44      0.61        16

    accuracy                           0.59        64
   macro avg       0.70      0.59      0.60        64
weighted avg       0.70      0.59      0.60        64
```

```
Classification report :
              precision    recall  f1-score   support

           0       0.06      0.06      0.06        16
           1       0.43      0.75      0.55        16
           2       1.00      0.44      0.61        16
           3       0.64      0.44      0.52        16

    accuracy                           0.42        64
   macro avg       0.53      0.42      0.43        64
weighted avg       0.53      0.42      0.43        64
```

Table 11
Average precision, average recall, average F1-score, and accuracy for the testing model in Experiment 2

		636 data set		
Epoch	**average precision**	**average recall**	**average F1-score**	**Accuracy**
20	0.54	0.53	0.52	0.53
40	0.70	0.59	0.60	0.59
60	0.53	0.42	0.43	0.42

Experiment 3: 1144 Data Set

The trained datasets were set to 1144 images, and the implemented CNN architecture results are shown in Table 12.

In Table 12, the results of the CNN model show an improvement in accuracy as the number of epochs increases. At epoch 20, the model achieved a training accuracy of 22.12%

and a validation accuracy of 25%. By epoch 40, the training accuracy had significantly improved to 86.62%, with a validation accuracy of 57%. Concerning epoch 60, the training accuracy dropped to 76.62%, followed by its validation accuracy at 52.19%. The model accuracy graph was shown in Tables 13 and 14 for the confusion matrix with its classification report.

Based on the result in Table 15, epoch 40 achieved high accuracy by testing average precision, recall F1 score and accuracy compared to epochs 20 and 60. It shows that epoch 40 was the optimum epoch for the dataset training.

Table 12
The results of performance loss, accuracy, validation loss, and validation accuracy were achieved for Experiment 3

1144 data set				
Epoch	**Loss**	**Accuracy**	**Val_loss**	**Val_accuracy**
20	1.3872	0.2212	1.3863	0.2500
40	0.3649	0.8662	1.9437	0.5702
60	0.5658	0.7662	1.6103	0.5219

Table 13
Validation accuracy of the model's accuracy graph for Experiment 3

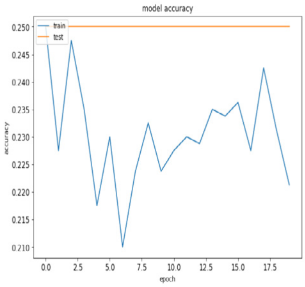
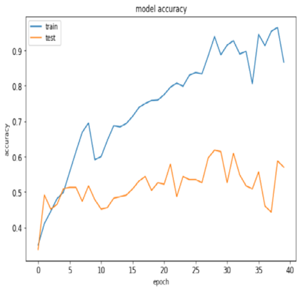
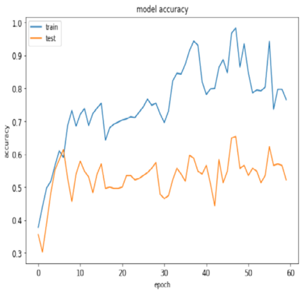
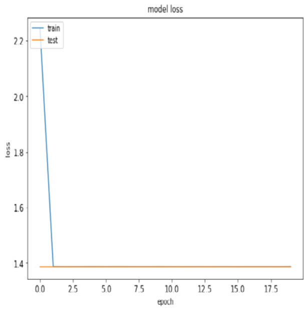
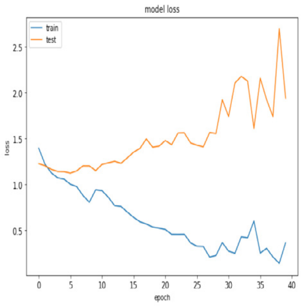
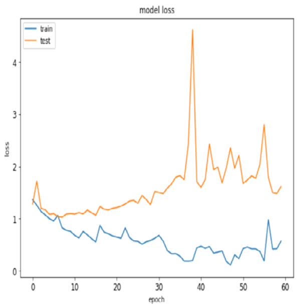
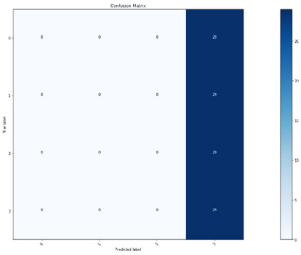
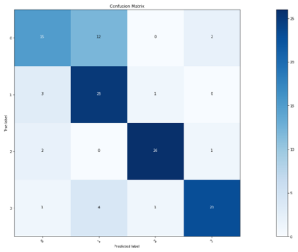
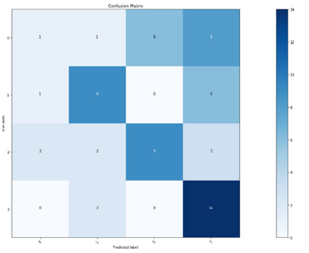
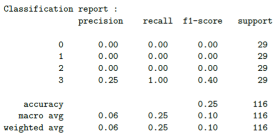
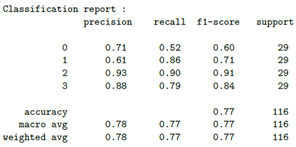
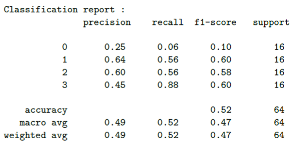
Epoch	20	40	60
Model accuracy			
Model loss			

Table 14
Result of the performance confusion matrix and classification report for the testing model Experiment 3

Epoch	20	40	60
Confusion matrix			
Classification report			

```
Classification report :
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        29
           1       0.00      0.00      0.00        29
           2       0.00      0.00      0.00        29
           3       0.25      1.00      0.40        29

    accuracy                           0.25       116
   macro avg       0.06      0.25      0.10       116
weighted avg       0.06      0.25      0.10       116
```

```
Classification report :
              precision    recall  f1-score   support

           0       0.71      0.52      0.60        29
           1       0.61      0.86      0.71        29
           2       0.93      0.90      0.91        29
           3       0.88      0.79      0.84        29

    accuracy                           0.77       116
   macro avg       0.78      0.77      0.77       116
weighted avg       0.78      0.77      0.77       116
```

```
Classification report :
              precision    recall  f1-score   support

           0       0.25      0.06      0.10        16
           1       0.64      0.56      0.60        16
           2       0.60      0.56      0.58        16
           3       0.45      0.88      0.60        16

    accuracy                           0.52        64
   macro avg       0.49      0.52      0.47        64
weighted avg       0.49      0.52      0.47        64
```

Table 15
Average precision, average recall, average F1-score, and accuracy for the testing model in Experiment 3

636 data set				
Epoch	**average precision**	**average recall**	**average F1-score**	**Accuracy**
20	0.06	0.25	0.10	0.25
40	0.78	0.77	0.77	0.77
60	0.49	0.52	0.47	0.52

Experiment 4: 1716 Data Set

The trained datasets were set to 1716 images, and the implemented CNN architecture results are shown in Table 16.

In Table 16, the results of the CNN model show an improvement in accuracy as the number of epochs increases. At epoch 20, the model achieved a training accuracy of 79.50% and a validation accuracy of 57.27%. By epoch 40, the training accuracy significantly improved to 94.75%, with a validation accuracy of 67.44%. Concerning epoch 60, the training accuracy dropped to 89.17%, followed by its validation accuracy at 61.34%. The model accuracy graph was shown in Tables 17 and 18 for the confusion matrix with its classification report.

Based on the results in Table 19, epochs 40 and 60 achieve high accuracy by testing average precision, recall F1 score and accuracy compared to epoch 20. It shows that the optimum epoch for the dataset training was at epochs 40 and 60. Through experiments 1, 2, 3 and 4, the variation in quality and diversity of topographic map datasets also pose significant challenges to CNN-based analysis. Variations in image quality and features also impact

Table 16
Result of performance loss, accuracy, validation loss, and validation accuracy achieved for Experiment 4

1716 data set				
Epoch	**Loss**	**Accuracy**	**Val_loss**	**Val_accuracy**
20	0.4311	0.7950	1.2615	0.5727
40	0.1296	0.9475	5.3041	0.6744
60	0.2567	0.8917	2.1453	0.6134

Table 17
Validation accuracy of the model's accuracy graph for Experiment 4

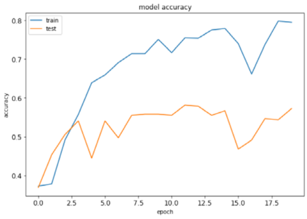
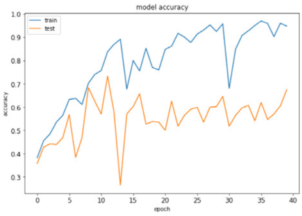
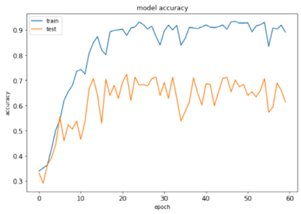
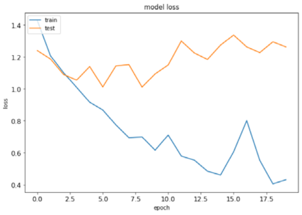
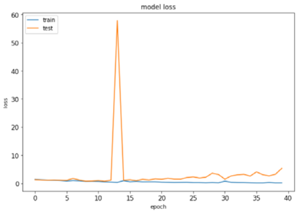
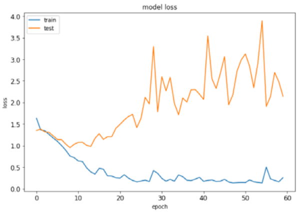
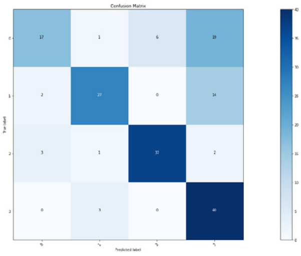
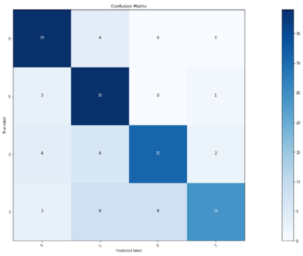
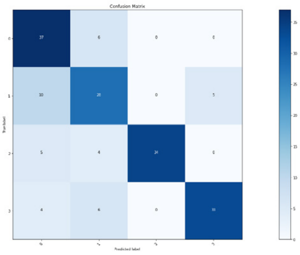
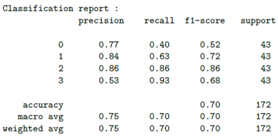
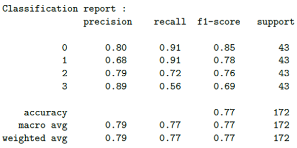
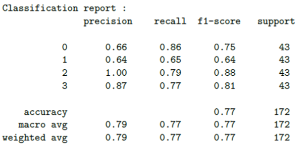
Epoch	20	40	60
Model accuracy			
Model loss			

Table 18
Result of the performance confusion matrix and classification report for the testing model Experiment 4

Epoch	20	40	60
Confusion Matrix			
Classification report			

Confusion Matrix

	0	1	2	3
0	17	1	6	19
1	2	27	0	14
2	3	1	37	2
3	0	3	0	40

```
Classification report :
              precision    recall  f1-score   support

           0       0.77      0.40      0.52        43
           1       0.84      0.63      0.72        43
           2       0.86      0.86      0.86        43
           3       0.53      0.93      0.68        43

    accuracy                           0.70       172
   macro avg       0.75      0.70      0.70       172
weighted avg       0.75      0.70      0.70       172
```

Confusion Matrix

	0	1	2	3
0	39	4	0	0
1	3	39	0	1
2	4	6	31	2
3	3	8	8	24

```
Classification report :
              precision    recall  f1-score   support

           0       0.80      0.91      0.85        43
           1       0.68      0.91      0.78        43
           2       0.79      0.72      0.76        43
           3       0.89      0.56      0.69        43

    accuracy                           0.77       172
   macro avg       0.79      0.77      0.77       172
weighted avg       0.79      0.77      0.77       172
```

Confusion Matrix

	0	1	2	3
0	37	6	0	0
1	10	28	0	5
2	5	4	34	0
3	4	6	0	33

```
Classification report :
              precision    recall  f1-score   support

           0       0.66      0.86      0.75        43
           1       0.64      0.65      0.64        43
           2       1.00      0.79      0.88        43
           3       0.87      0.77      0.81        43

    accuracy                           0.77       172
   macro avg       0.79      0.77      0.77       172
weighted avg       0.79      0.77      0.77       172
```

Table 19
Average precision, average recall, average F1-score, and accuracy for the testing model in Experiment 4

1716 data set				
Epoch	**average precision**	**average recall**	**average F1-score**	**Accuracy**
20	0.75	0.70	0.70	0.70
40	0.79	0.77	0.77	0.77
60	0.79	0.77	0.77	0.77

the model's accuracy and generalizability. Future research should address these challenges by evaluating computational efficiency and resource requirements and exploring methods to enhance the model's adaptability to diverse map characteristics. Integrating CNN models with GIS enhances the usability of topographic map data, supporting automated feature extraction and spatial analysis for informed decision-making in various domains. Additionally, visualisation tools are crucial for ensuring user understanding, with detection boxes overlaid on maps facilitating the interpretation of model outputs. Developing system tools for object detection can further streamline the application of CNN models to topographic map analysis. Throughout the experiment, findings also reveal that all testing achieved their optimum accuracy at epoch 40. It indicates that the greater number of epochs does not affect the higher accuracy achieved for this dataset. Thus, the epoch number needs to be compatible with the dataset types to achieve an optimal detection model.

Examples of practical challenges encountered during map analysis include complex cartographic details and variability in map quality. The complex detail varies in size, shape, and clarity, posing challenges for accurate feature extraction and classification. The challenges introduce the capability of the CNN model, which is able to handle diverse datasets of HTHMs. The model is designed to learn hierarchical representations of these complex cartographic features. The model can capture local spatial patterns and semantic information by leveraging convolutional layers, accurately classifying and interpreting various map elements and quality.

CONCLUSION

In summary, the classification report analysis demonstrates that the model performs reasonably well classifying instances from the dataset. However, further improvements can be made to enhance the model to achieve a more balanced performance for each class. Findings from this study also contribute to understanding the model's strengths and limitations, providing valuable insights for future research and refinement of the classification approach. Overall, evaluating CNN architecture for analysing HTHM has provided valuable insights into its effectiveness and potential applications. The study demonstrated that CNNs can accurately classify instances from the HTHM dataset, showcasing their suitability for analysing complex cartographic details, such as contour lines, symbols, and textual annotations. The model exhibited satisfactory performance in most classes, with room for improvement in specific categories. This study also provides a foundation for advancing map analysis and interpretation within GIS. It underscores the potential of CNNs in automating the vectorisation process and facilitating the broader access and preservation of valuable historical records embedded in topographic maps. For potential future research directions, a hybrid CNN model with other machine learning algorithms for its higher accuracy in classification tasks could enhance the study's accuracy, and additional datasets covering maps from various years would provide a broader range of topographical map data for analysis.

ACKNOWLEDGMENTS

The authors sincerely thank Universiti Teknologi MARA Selangor, Malaysia, for providing Geran Intensif Penyelidikan (GIP) Research Fund 600-RMC/GIP 5/3 (074/2022) and to all those directly or indirectly involved in this research.

REFERENCES

Ali, L., Alnajjar, F., Jassmi, H. A., Gocho, M., Khan, W., & Serhani, M. A. (2021). Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures. *Sensors*, *21*(5), Article 1688. https://doi.org/10.3390/s21051688

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaria, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, *8*, Article 53. https://doi.org/10.1186/s40537-021-00444-8

Althnian, A., AlSaeed, D., Al-Baity, H., Samha, A., Dris, A., Alzakari, N., Elwafa, A. A., & Kurdi, H. (2021). Impact of dataset size on classification performance: An empirical evaluation in the medical domain. *Applied Sciences, 11*(2), Article 796. https://doi.org/10.3390/app11020796

Anuar, S., Ibrahim, J., Rauf, A., & Rasam, A. (2021). Towards automated digitization of cartographic hardcopy maps: Reviews of issues, challenges and potentials in Malaysia Library archives. *Malaysian Journal of Remote Sensing & GIS, 10*(1), 43-51.

Audebert, N., Le Saux, B., & Lefevre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE Geoscience and Remote Sensing Magazine, 7*(2), 159–173. https://doi.org/10.1109/MGRS.2019.2912563

Bhosle, K., & Musande, V. (2022). Evaluation of CNN model by comparing with convolutional autoencoder and deep neural network for crop classification on hyperspectral imagery. *Geocarto International, 37*(3), 813–827. https://doi.org/10.1080/10106049.2020.1740950

Barry-Straume, J., Tschannen, A., Engels, D. W., & Fine, E. (2018). An evaluation of training size impact on validation accuracy for optimized convolutional neural networks. *SMU Data Science Review, 1*(4), Article 12.

Chen, Y., Jiang, H., Li, C., Jia, X., & Ghamisi, P. (2016). Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE transactions on geoscience and remote sensing, 54*(10), 6232-6251. https://doi.org/10.1109/TGRS.2016.2584107

Chauhan, R., Ghanshala, K. K., & Joshi, R. C. (2018, December 15-17). *Convolutional Neural Network (CNN) for image detection and recognition*. [Paper presentation]. First International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India. https://doi.org/10.1109/ICSCCC.2018.8703316.

Dawson, H. L., Dubrule, O., & John, C. M. (2023). Impact of dataset size and convolutional neural network architecture on transfer learning for carbonate rock classification. *Computers and Geosciences*, *171*, Article 105284. https://doi.org/10.1016/j.cageo.2022.105284

Dwivedi, D., & Patil, G. (2022). Lightweight convolutional neural network for land use image classification. *Journal of Advanced Geospatial Science & Technology, 2*(1), 31-48. https://doi.org/10.11113/jagst.v2n1.31

Garbin, C., Zhu, X., & Marques, O. (2020). Dropout vs. batch normalization: An empirical study of their impact to deep learning. *Multimedia Tools and Applications 79*(19), 12777–12815. https://doi.org/10.1007/s11042-019-08453-9

Guo, W., Yang, W., Zhang, H., & Hua, G. (2018). Geospatial object detection in high resolution satellite images based on multi-scale convolutional neural network. *Remote Sensing, 10*(1), Article 131. https://doi.org/10.3390/rs10010131

Hamouda, M., Ettabaa, K. S., & Bouhlel, M. S. (2020). Smart feature extraction and classification of hyperspectral images based on convolutional neural networks. *IET Image Processing, 14*(10), 1999-2005. https://doi.org/10.1049/iet-ipr.2019.1282

Ji, S., Zhang, C., Xu, A., Shi, Y., & Duan, Y. (2018). 3D convolutional neural networks for crop classification with multi-temporal remote sensing images. *Remote Sensing, 10*(1), Article 75. https://doi.org/10.3390/rs10010075

Johny, A., & Madhusoodanan, K. N. (2021). Dynamic learning rate in deep CNN model for metastasis detection and classification of histopathology images. *Computational and Mathematical Methods in Medicine, 2021*(1), Article 5557168. https://doi.org/10.1155/2021/5557168

Kandel, I., & Castelli, M. (2020). The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT Express, 6*(4), 312-315. https://doi.org/10.1016/j.icte.2020.04.010

Khalifa, N. E., Loey, M., & Mirjalili, S. (2022). A comprehensive survey of recent trends in deep learning for digital images augmentation. *Artificial Intelligence Review, 55*(3), 2351-2377. https://doi.org/10.1007/s10462-021-10066-4

Kumar, A., Gaur, N., Chakravarty, S., Alsharif, M. H., Uthansakul, P., & Uthansakul, M. (2024). Analysis of spectrum sensing using deep learning algorithms: CNNs and RNNs. *Ain Shams Engineering Journal, 15*(3), Article 102505. https://doi.org/10.1016/j.asej.2023.102505

Li, Z., Xin, Q., Sun, Y., & Cao, M. (2021). A deep learning-based framework for automated extraction of building footprint polygons from very high-resolution aerial imagery. *Remote Sensing, 13*(18), Article 3630 https://doi.org/10.3390/rs13183630

Liu, B., Du, S., & Zhang, X. (2020). Land cover classification using convolutional neural network with remote sensing data and digital surface model. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 5*(3), 39–43. https://doi.org/10.5194/isprs-annals-V-3-2020-39-2020

Poojary, R., Raina, R., & Mondal, A. K. (2021). Effect of data-augmentation on fine-tuned CNN model performance. *IAES International Journal of Artificial Intelligence, 10*(1), 84-92, https://doi.org/10.11591/ijai.v10.i1.pp84-92

Roslan, N. A. M., Diah, N. M., Ibrahim, Z., Munarko, Y., & Minarno, A. E. (2023). Automatic plant recognition using convolutional neural network on Malaysian medicinal herbs: The value of data augmentation. *International Journal of Advances in Intelligent Informatics, 9*(1), 136-147. https://doi.org/10.26555/ijain.v9i1.1076

Sharifi, O., Mokhtarzadeh, M., & Beirami, B. A. (2022). A new deep learning approach for classification of hyperspectral images: Feature and decision level fusion of spectral and spatial features in multiscale CNN. *Geocarto International, 37*(14), 4208-4233. https://doi.org/10.1080/10106049.2021.1882006

Tharwat, A. (2020). Classification assessment methods. *Applied Computing and Informatics*, *17*(1), 168-192. https://doi.org/10.1016/j.aci.2018.08.003

Zeng, P. (2020) On the confusion matrix in credit scoring and its analytical properties. *Communications in Statistics-Theory and Methods, 49*(9), 2080-2093. https://doi.org/ 10.1080/03610926.2019.1568485