SEMARAK ILMU PUBLISHING
202103268166(003316878-P)

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

A Comparative Study on Flood Detection using Deep Learning Techniques

Mohd Shahar Abdullah[1], Ali Selamat[1,2,3,4,*], Nguyet Quang Do[1,*], Mohd Azlan Abu[1]

1 Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, Kampung Datuk Keramat, 54100 Kuala Lumpur, Wilayah Persekutuan Kuala Lumpur, Malaysia
2 School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, Johor Baru, Johor 80000, Malaysia
3 Media and Games Center of Excellence (MagicX), Universiti Teknologi Malaysia, Johor Baru, Johor 80000, Malaysia
4 Center for Basic and Applied Research, Faculty of Informatics and Management, University of Hradec Kralove, Rokitanskeho 62, 500 03 Hradec Kralove, Czech Republic

ARTICLE INFO

Article history:
Received 27 November 2026
Received in revised form 20 February 2026
Accepted 15 April 2026
Available online 27 April 2026

Keywords:
Disaster management; flood detection; image classification; deep learning; pre-trained model

ABSTRACT

Disasters, both natural and man-made, pose significant threats to societies worldwide, causing devastating losses to life, property, and the environment. The frequency and severity of these events have increased in recent years, partly due to climate change and urbanization. Among natural disasters, flood is one of the most common and destructive calamities. Many solutions have been proposed to mitigate the detrimental impacts of flood. However, traditional measures often struggle to keep pace with the rapid evolution of these crises. Meanwhile, artificial intelligence presents a promising avenue for revolutionizing disaster management. Therefore, this paper aims to provide a comparative study on flood detection using deep learning technologies. Specifically, nine pre-trained models based on Convolutional Neural Network (CNN), including DenseNet121, VGG16, ResNet50, MobileNet, InceptionV3, Xception, EfficientNetB2, ConvNeXtTiny, and NASNetMobile, were implemented for flood image classification. Results obtained from the experiments indicated that ConvNeXtTiny produced the best performance compared to other methods, achieving the highest detection accuracy of 96.75%. Findings from this study can be beneficial to stakeholders in the early stages of disaster management. By leveraging deep learning technologies, this paper seeks to automate and optimize disaster preparedness, ultimately saving lives and reducing economic losses.

1. Introduction
1.1 Research Background

Floods are among the most common and detrimental natural disasters worldwide. Financial damage caused by flood between 1998-2017 were more than US$600 billion, with more than 2 billion people affected, and around 142,000 fatalities involved [1]. With the recent shift in urbanization and climate change, these losses tend to rise rapidly. This becomes a challenge faced by many countries which hinder their efforts to overcome poverty and obtain prosperity growth.

* *Corresponding author.*
E-mail address: aselamat@utm.my

https://doi.org/10.37934/araset.58.5.198207

Nevertheless, traditional approaches (e.g., modeling, calculation, dimensionality [2][3]) have limitations in addressing natural disasters, including flood [4]. In addition, little attention has been paid to long-term planning and mitigation prior to disasters. In recent years, a sustainable protection of people and property from negative impacts of natural hazards has captured the attention of the research community, where a more balanced use of structural and non-structural measures is needed. This poses greater uncertainty and complexity than traditional disaster risk reduction methods, which is time-consuming, labor-intensive, and computationally expensive [5]. Yet, artificial intelligence (AI) technologies can address the uncertainty and complexity issues if applied reasonably by reducing computational constraints and labor costs [6].

AI presents a promising avenue for revolutionizing disaster management [7]. By utilizing the power of AI, it is possible to analyze vast amounts of data from diverse sources in real-time, such as satellite imagery, social media, and sensor networks. This capability enables the prediction of disaster events with greater accuracy, the optimization of resource allocation, and the implementation of more effective response strategies [8]. Motivated by the power of AI in disaster management, this study examined the performance of various state-of-the-art models in detecting flood disaster. By using transfer learning approach, several pre-trained models were fine-tuned for flood image classification. These models were trained on a public dataset and evaluated using numerous performance metrics, such as Precision, Recall, F1-Score, AUC, and Accuracy.

1.2 Literature Review

The detection, mitigation, and prevention of natural disasters are vital in guaranteeing the safety and welfare of the affected communities. Align with this goal, the utilization of AI has shown significantly positive impacts as a valuable tool to efficiently handle and manage these catastrophic events [9]. Recently, extensive research has been carried out using AI to assist disaster management in terms of real-time prediction and detection of natural calamities [10]. Among a wide range of AI-based applications, deep learning (DL), a subset of machine learning (ML), has emerged as a promising solution to assist the relevant agencies in handling natural disasters [11]. DL has shown tremendous success in processing image data, which outperforms human-operated systems in numerous scenarios. Once of the most successful DL models is known as Convolutional Neural Network (CNN). In the field of disaster management, CNN can be used to extract patterns from satellite imaging to identify the affected areas and evaluate the magnitude of destruction [12].

In the flood detection domain, CNN can be utilized for flood image classification. The utilization of CNN in image-related tasks has resulted in an increasing number of research works on its application in the context of flood detection. The recent advancement in information technologies has led to a significant number of advanced models based on CNN for image classification. For example, Kapilaratne and Kaneta [13] examined and compared the performance of ResNet102 and MobileNetV2 in flood area extraction from satellite images. Results obtained from their study showed that both models had potential of flood area mapping from optical images, yet the accuracies achieved not more than 91%. Instead of relying only on images, Jony *et al*. [14] fused visual features with textual metadata for flood detection. Three CNN architectures, including VGG16, InceptionV3, and Xception, were used to extract visual features from images on an online social media platform. Although the fusion method improved the classification accuracy compared to the individual techniques, the authors did not benchmark the three CNN architectures with other models to highlight their capabilities.

Besides satellite, drones can also be used for post-flood disaster management. Islam *et al*. [15] proposed an approach to identify flood-affected areas with image classification using DensNet and

InceptionV3. Even though their study utilized merge sorting algorithm to prioritize flood image output, the proposed model was trained on self-made dataset with the obtained accuracies less than 83%. In the study by Patil *et al*. [16], VGG16, DenseNet, and GoogleNet were combined with remote sensing for flood detection. Yet, the accuracies of these models were less than 83%. To further improve the detection accuracy, Huang *et al*. [17] employed an attention mechanism to the existing ResNet model for urban waterlogging recognition. When benchmarked with the classic ResNet model, EfficientNet, and 3D CNN, the proposed solution provided comparable results.

In addition to the aforementioned models, Jackson *et al*. [18] used different CNN architectures, such as ShuffleNet, RegNet, and Vision Transformer, for flood image classification. The authors also examined the impact of different uncertainty offset on the performance of the classification models. Nevertheless, a manual annotation approach was used for image categorization, which is labor-intensive and time-consuming. Similarly, Mittal *et al*. [19] compared the performance of seven different CNN architectures in detecting inundation areas. However, all the obtained accuracies were less than 85%. Yasi *et al*. [20] took an extra mile by combining eight CNN-based models in an ensemble manner to provide a more effective approach for flood classification. Nonetheless, the ensemble model was trained on a relatively small dataset and achieved almost similar detection accuracies as one of the individual architectures. Overall, the strengths and limitations of the aforementioned studies are summarized in Table 1.

Table 1
Strengths and limitations of the related works

Reference	Strengths	Limitations
[13]	- Computed the processing time during testing	- Obtained accuracies not more than 91%
[14]	- Fused visual features and textual metadata	- Did not benchmark with other models, such as DenseNet, ResNet, MobileNet, etc.
[15]	- Utilized merge sorting algorithm to prioritize flood image output	- Trained on self-made flood dataset - Obtained accuracies not more than 83%
[16]	- Used GoogleNet in addition to VGG and DenseNet	- Obtained accuracies less than 83%
[17]	- Employed attention mechanism to improve the recognition accuracy of waterlogging - Applied data augmentation to increase the training data	- Did not use other models, such as DenseNet, VGG, MobileNet, Inception, Xception, ConvNeXt, NASNet - Did not benchmark with other models
[18]	- Used different models, such as ShuffleNet, RegNet, Vision Transformer - Examined the impact of different uncertainty offset	- Did not employ other models, such as DenseNet, Inception, Xception, and NASNet - Used manual annotation approach
[19]	- Compared the performance of various models	- Obtained accuracies less than 85%
[20]	- Combined various models in an ensemble approach	- Trained on a relatively small dataset

Unlike the previous works, this study provides a comparative analysis of nine different CNN architectures, including DenseNet121, VGG16, ResNet50, MobileNet, InceptionV3, Xception, EfficientNetB2, ConvNeXtTiny, and NASNetMobile as shown in Table 2. By extensively examining various state-of-the-art CNN-based models, a comprehensive view of the existing solutions for flood image classification can be obtained. The main objectives of this paper are as follows:

i. To investigate nine state-of-the-art CNN-based models for flood image classification using transfer learning approach;
ii. To examine the performance of these pre-trained models in detecting flood events through a comparative analysis;

iii. To evaluate the performance of numerous deep learning models through several experiments using various performance metrics.

Table 2
Comparison between the related works and this study

Reference	DenseNet	VGG	ResNet	MobileNet	Inception	Xception	EfficientNet	ConvNeXt	NASNet
[13]			✓	✓					
[14]		✓			✓	✓			
[15]	✓				✓				
[16]	✓	✓							
[17]			✓				✓		
[18]		✓	✓	✓			✓	✓	
[19]	✓	✓	✓	✓	✓	✓			✓
[20]	✓	✓	✓	✓	✓	✓	✓		✓
This study	✓	✓	✓	✓	✓	✓	✓	✓	✓

2. Methodology

2.1 Flood Detection Model

The deep learning model for flood image classification is demonstrated in Figure 1. The model first takes an image as an input and converts it into an array using pre-processing techniques. Next, nine CNN-based models are loaded with pre-trained weights and fine-tuned for binary classification task using a single dense layer. Finally, the output layer applies a Sigmoid function to classify the given image as flood or non-flood. A detailed description of specific processes inside the flood detection model is provided in Algorithm 1.

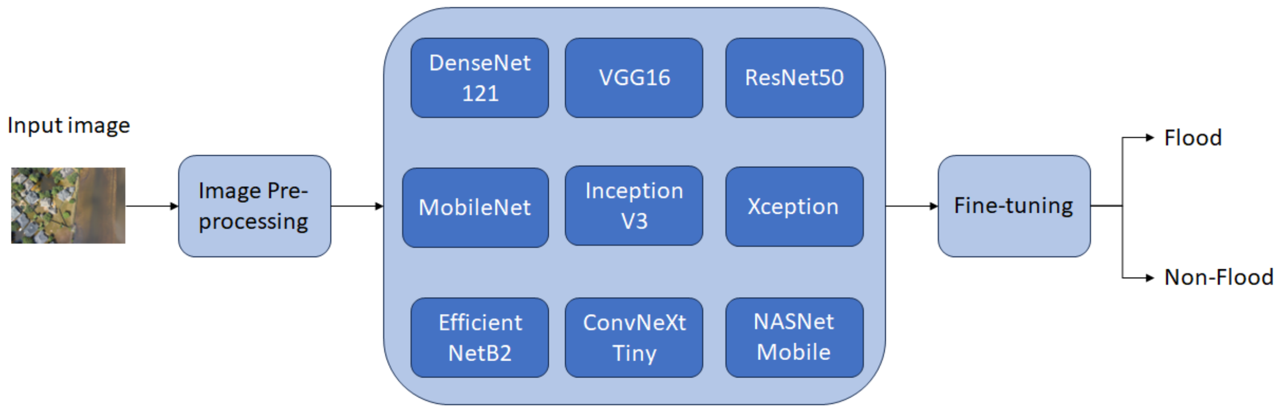

Fig. 1. Flood Detection Model

DenseNet121: DenseNet consists of multiple fully-connected blocks where the preceding block is connected to every other succeeding block. DenseNet121 is one of the DenseNet-based implementations. DenseNet121 contains four dense blocks with transition layers in between these blocks to reduce spatial dimension and the number of channels [20].

VGG16: VGGNet is one of the CNN-based models with deep network architecture and high performance [21]. VGGNet has demonstrated outstanding results in several tasks, including remote sensing, image classification, and object detection [22]. VGG16 is one of the two popular versions of VGGNet, consisting of 13 convolutional layers and 3 fully-connected layers.

Algorithm 1: Flood Detection Model

Input: *I*
Output: *Label*

I ← Load_and_Preprocess(*Input_images*);
I_{train}, I_{test} ← train_test_split (*I*);
base_model ← Load_model (*weights*, *Input_shape*);
Flat ← Flatten(*base_model*);
FC ← Dense (*Flat*);
M ← Sigmoid (*FC*);
M_c ← Model_compile (*Adam_optimizer*, *Learning_rate*);
for *i* in *Epoch* **do**
    M_f ← Model_fit ([I_{train}, I_{test}], *Batch_size*);
end for
Pr, *Rc*, F_1, *Acc* ← Model_evaluate (I_{test});
y_p ← Model_prediction (I_{test});
if $y_p >= 0.5$ **then**
    Label ← Flood;
else
    Label ← Non-Flood;
end if
return *Label*;

ResNet50: ResNet was first proposed to address the vanishing and exploding gradient issue in deep neural network [23]. ResNet consists of multiple convolutional layers, allowing information to go through using skip connection and residual learning [24]. ResNet50, which contains 50 layers, is one of the ResNet-based architectures.

MobileNet: MobileNet was designed for image processing on limited-resource devices. Due to its high computing efficiency, MobileNet is suitable for devices with limited computing power, such as IoT devices, mobile phones, tablets, and so on [25].

InceptionV3: InceptionV3 is an improved version of GoogleNet (or InceptionV1). InceptionV3 was proposed to overcome the drawbacks of its older version, which are accuracy deficiency and high convergence time [15].

Xception: Xception is an enhanced version of InceptionV3, making use of depth-wise separable convolutions. The model comprises 14 modules of 36 convolutional layers with residual connections in 12 modules, excluding the first and the last ones [20].

EfficientNetB2: EfficientNet was developed to overcome the random and manual scaling of the conventional models. It uses the compound coefficient to effectively scale up the depth, width, and image resolution of a baseline model, leading to a balance between model accuracy and computational cost [20].

ConvNeXtTiny: ConvNeXt was originally proposed to reduce the gap between classical CNN and transformer-based architectures. While maintaining the efficiency of the conventional CNN models, ConvNeXt provides better performance than transformer-based architectures across a wide range of computer vision tasks [18]. ConvNeXtTiny is a light version of ConvNeXt with smaller size and fewer parameters.

NASNetMobile: NASNet was developed based on the Neural Architecture Search (NAS) technique. Compared to NASNetLarge, NASNetMobile is a lighter version that is suitable for devices with limited computing power [20].

2.2 Dataset Description

Two public datasets are used in this study to form a balanced dataset for model training. The first dataset is obtained from Kaggle, consisting of 3202 flood images and 717 non-flood images. The second dataset is called FloodNet [26] with 2307 labelled images; out of which 320 and 1987 are flood and normal instances, respectively. Since both datasets are imbalanced, they are combined to form a balanced dataset with a total of 4000 images, including 2013 samples as flood and 1987 samples as non-flood. The balanced dataset is then split into training and testing sets with a ratio of 8:2 using hold-out validation method.

2.3 Parameter Settings

All images in the balanced dataset are downsized to 224x224x3 dimensions to make them compatible with the nine deep learning models, while pre-trained weights from ImageNet are used for these models. A single dense layer with 128 neurons is added to fine-tune them for a binary classification task. Finally, a single neuron is used in the output layer with the Sigmoid activation function to categorize the input image as flood or normal. The parameter settings for the experiments in this study are provided in Table 3.

Table 3
Parameter settings for the experiments

Layer	Parameter	Value
Input	Input shape	224x224x3
Base model	Weights	imagenet
Dense	Number of neurons	128
	Activation function	Rectified Linear Unit (ReLU)
Output	Number of neurons	1
	Activation function	Sigmoid
Other	Optimizer	Adam
	Learning rate	0.0001
	Batch size	32
	Epoch	10

2.4 Evaluation Metrics

Several metrics are used to evaluate the performance of the flood detection models, including Precision, Recall, F1-Score, and Accuracy. The formulas for these metrics are given in Eq. (1)-(4).

$$\text{Precision} = \frac{TP}{TP+FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{2}$$

$$\text{F1} - \text{Score} = 2\,\frac{Precision.Recall}{Precision+Recall} \tag{3}$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

where TP, TN, FP, and FN refer to True Positive, True Negative, False Positive, and False Negative, respectively. For the flood detection problem in this study, a flood image is considered a positive instance, whereas a normal image is regarded as a negative instance.

3. Results

Table 4 summarizes the results obtained from a series of experiments in this study. Notably, ConvNeXtTiny achieved the highest detection accuracy of 96.75%, whereas ResNet50 produced the lowest accuracy of 91.62%. Other models, such as VGG16, InceptionV3, Xception, and EfficientNetB2 showed similar performance, with the detection accuracies around 94%. DenseNet121 and MobileNet produced comparable performance, achieving 95.38% and 95.88% in accuracy, respectively. Meanwhile, NASNetMobile performed slightly better than ResNet50 by 0.88%, but still poorer than other models as shown in Figure 2. Overall, ConvNeXtTiny outperformed other eight models in three out of four evaluation metrics, including Recall, F1-Score, and Accuracy.

Table 4
Performance metrics of various deep learning models

Model	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
DenseNet121	92.82	97.91	95.30	95.38
VGG16	93.28	94.26	93.77	94.00
ResNet50	94.38	87.73	90.93	91.62
MobileNet	95.57	95.82	95.70	95.88
InceptionV3	95.23	93.73	94.47	94.75
Xception	93.15	95.82	94.47	94.63
EfficientNetB2	93.16	96.08	94.60	94.75
ConvNeXtTiny	95.19	**98.17**	**96.66**	**96.75**
NASNetMobile	88.00	97.65	92.57	92.50

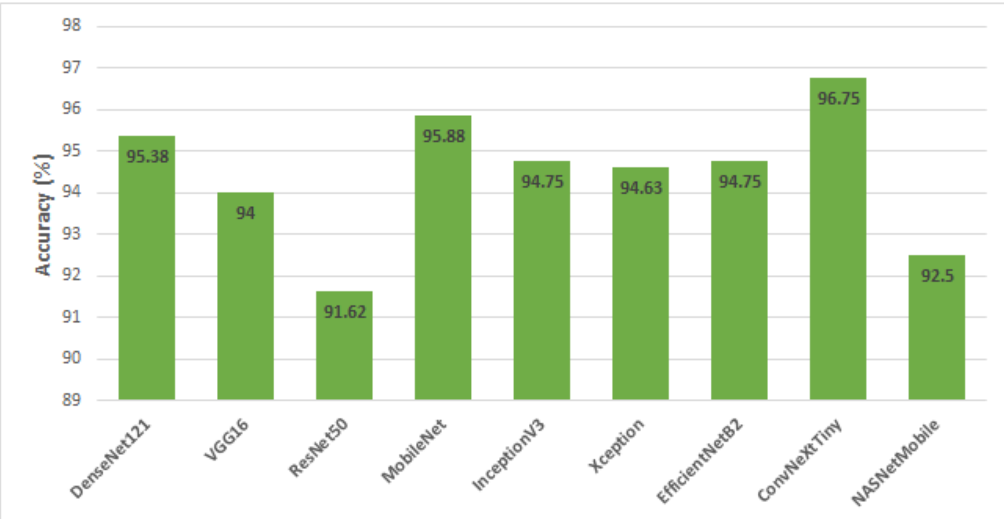

Fig. 2. Accuracies of various CNN-based models

The training and validation accuracies of nine CNN-based models are illustrated in Figure 3. Both training and validation accuracies started to converge after 10 epochs, achieving the maximum accuracies from 10 iterations onwards. Moreover, it is observed that training accuracies were higher than validation accuracies, indicating that overfitting occurred. This is partly because data augmentation was not included in the data pre-processing phase.

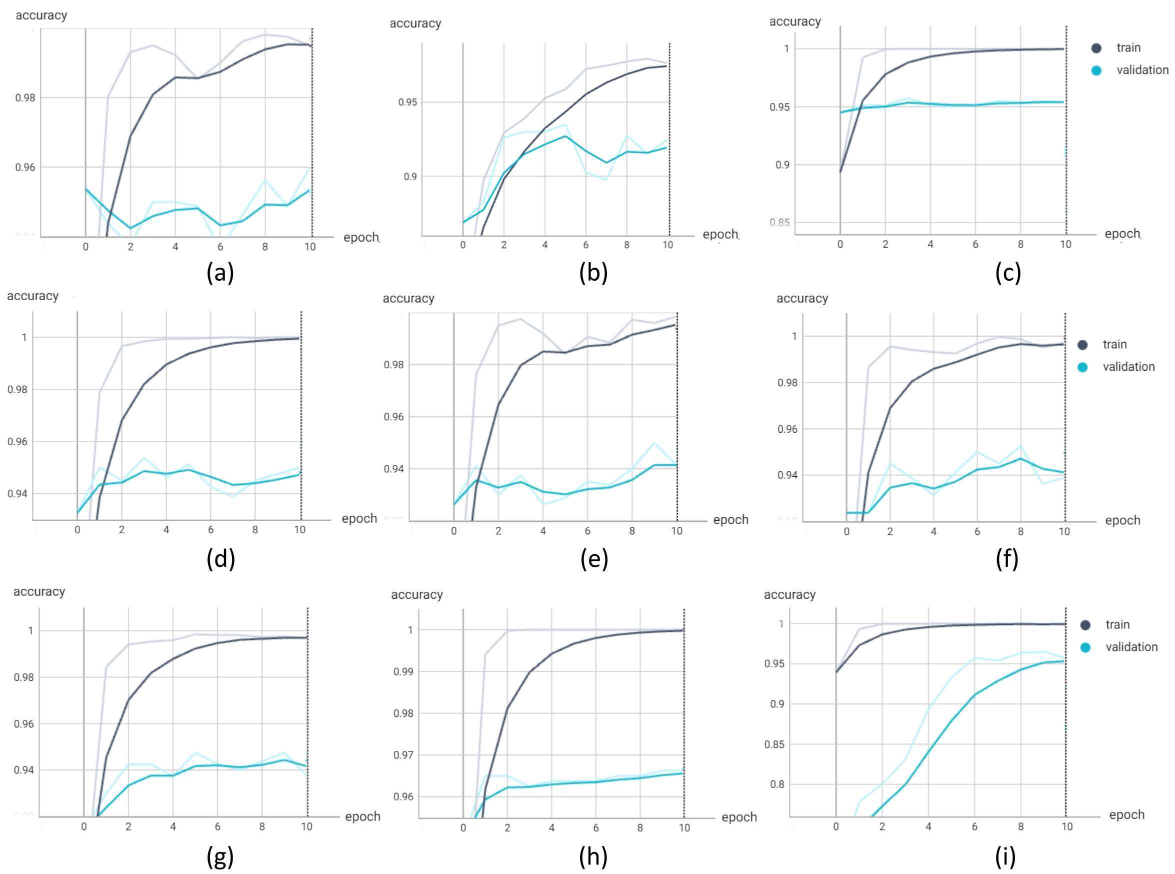

Fig. 3. Train and validation accuracy (a) DenseNet121 (b) VGG16 (c) ResNet50 (d) MobileNet (e) InceptionV3 (f) Xception (g) EfficientNetB2 (h) ConvNeXtTiny (i) NASNetMobile

Figure 4 plots the Receiver Operating Characteristics (ROC) curves of nine deep learning models. Similar to Accuracy, the highest Area Under the Curve (AUC) value was achieved by the ConvNeXtTiny model (99.37%). The highest AUC value suggests that ConvNeXtTiny is better than other eight models in flood image classification.

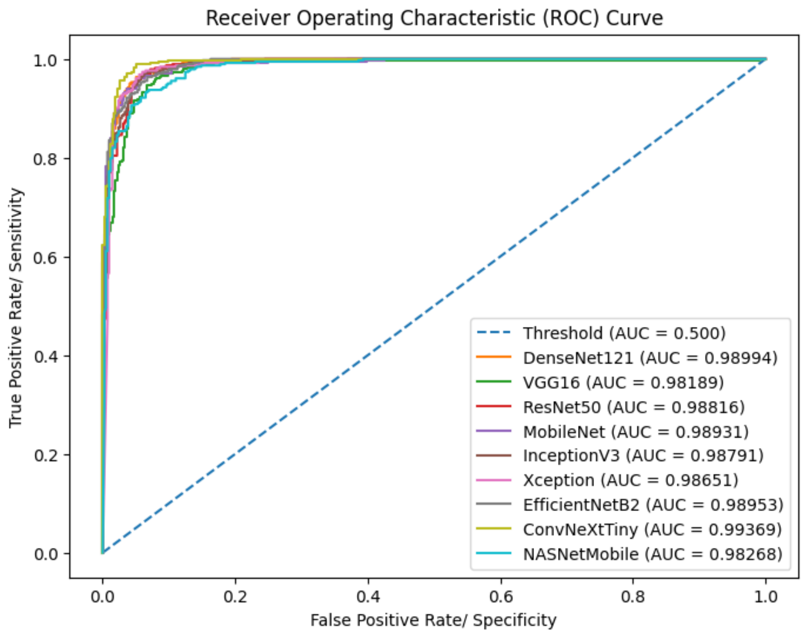

Fig. 4. ROC curves of various deep learning models

4. Conclusions

In conclusion, this paper provides a comparative study of different CNN-based models' performance for flood detection. These models were trained on a public dataset and evaluated using various performance metrics. The experimental results showed that ConvNeXtTiny outperformed all other eight CNN architectures in flood image classification. Findings obtained from this study can lead to more innovations of effective and accurate disaster preparedness, response, and recovery solutions, which could help in mitigating the negative impacts of natural disasters.

However, there are several limitations in this study which need to be further improved. First, data augmentation was not involved in the data pre-processing phase, causing overfitting issue in the deep learning models. Second, the dataset used in the experiments was rather small, which could not explore the full potential of deep learning algorithms. As a result, in the future research, we plan to perform data augmentation for image processing to address the overfitting problem. Additionally, we will also use more significant datasets to examine the generalization ability of the deep learning models for flood detection.

Acknowledgement
This work was supported by Universiti Teknologi Malaysia (UTM) under Malaysia Research University Network (MRUN) Vot 4L876. This work was also partially funded by the Ministry of Higher Education (MOHE) under the Fundamental Research Grant Scheme (FRGS/1/2022/ICT08/UTM/01/1). The work and contribution were also supported by the SPEV project "Smart Solutions in Ubiquitous Computing Environments", University of Hradec Kralove, Faculty of Informatics and Management, Czech Republic (under ID: UHK-FIM-SPEV-2022-2102).

References
[1] Tuyen, Do Ngoc, Tran Manh Tuan, Le Hoang Son, Tran Thi Ngan, Nguyen Long Giang, Pham Huy Thong, Vu Van Hieu, Vassilis C. Gerogiannis, Dimitrios Tzimos, and Andreas Kanavos. 2021. "A Novel Approach Combining Particle Swarm Optimization and Deep Learning for Flash Flood Detection from Satellite Images." *Mathematics* 9 (22): 2846. https://doi.org/10.3390/math9222846.

[2] Qi, Wenchao, Chao Ma, Hongshi Xu, Zifan Chen, Kai Zhao, and Hao Han. 2021. "A Review on Applications of Urban Flood Models in Flood Mitigation Strategies." *Natural Hazards* 108: 31–62. https://doi.org/10.1007/s11069-021-04715-8.

[3] Rahiman, Wan. 2021. "Prototype of Water Dam System by Using FPGA Platform." *Journal of Advanced Research in Computing and Applications* 24 (1): 1–13.

[4] Li, Yu Hong, Zou Liang, and Li Li. 2022. "Improving Urban Stormwater Resilience by Promoting Sponge City Construction System." *Journal of Human Settlements in West China* 37 (1): 22–26. https://doi.org/10.13791/j.cnki.hsfwest.20220104.

[5] Anees, Mohd Talha, K. Abdullah, M. N. M. Nawawi, Nik Norulaini Nik Ab Rahman, Abd Rahni Mt Piah, Nor Azazi Zakaria, M. I. Syakir, and A. K. Mohd Omar. 2016. "Numerical Modeling Techniques for Flood Analysis." *Journal of African Earth Sciences* 124: 478–486. https://doi.org/10.1016/j.jafrearsci.2016.10.001.

[6] Dai, Daixin, Mingyang Bo, Xiaosong Ren, and Kaiyu Dai. 2024. "Application and Exploration of Artificial Intelligence Technology in Urban Ecosystem-Based Disaster Risk Reduction: A Scoping Review." *Ecological Indicators* 158: 111565. https://doi.org/10.1016/j.ecolind.2024.111565.

[7] Yigitcanlar, Tan, Luke Butler, Emily Windle, Kevin C. Desouza, Rashid Mehmood, and Juan M. Corchado. 2020. "Can Building 'Artificially Intelligent Cities' Safeguard Humanity from Natural Disasters, Pandemics, and Other Catastrophes? An Urban Scholar's Perspective." *Sensors* 20 (10): 2988. https://doi.org/10.3390/s20102988.

[8] Sun, Wenjuan, Paolo Bocchini, and Brian D. Davison. 2020. "Applications of Artificial Intelligence for Disaster Management." *Natural Hazards* 103 (3): 2631–2689. https://doi.org/10.1007/s11069-020-04124-3.

[9] Chai, Jinchun, and Hao-Ze Wu. 2023. "Prevention/Mitigation of Natural Disasters in Urban Areas." *Smart Construction and Sustainable Cities* 1 (1): 4. https://doi.org/10.1007/s44268-023-00002-6.

[10] Xu, Chong, and Zhiwen Xue. 2023. "Applications and Challenges of Artificial Intelligence in the Field of Disaster Prevention, Reduction, and Relief." *Natural Hazards Research*. https://doi.org/10.1016/j.nhres.2023.11.011.

[11] Rathnasiri, Pavithra, Onaopepo Adeniyi, and Niraj Thurairajah. 2023. "Data-Driven Approaches to Built Environment Flood Resilience: A Scientometric and Critical Review." *Advanced Engineering Informatics* 57: 102085. https://doi.org/10.1016/j.aei.2023.102085.

[12] Akhyar, Akhyar, Mohd Asyraf Zulkifley, Jaesung Lee, Taekyung Song, Jaeho Han, Chanhee Cho, Seunghyun Hyun, Youngdoo Son, and Byung-Woo Hong. 2024. "Deep Artificial Intelligence Applications for Natural Disaster Management Systems: A Methodological Review." *Ecological Indicators* 163: 112067. https://doi.org/10.1016/j.ecolind.2024.112067.

[13] Kapilaratne, R. G. C. J., and S. Kaneta. 2020. "Towards an Automated Flood Area Extraction from High Resolution Satellite Images." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 43: 97–104.

[14] Jony, Rabiul Islam, Alan Woodley, and Dimitri Perrin. 2020. "Fusing Visual Features and Metadata to Detect Flooding in Flickr Images." In *2020 Digital Image Computing: Techniques and Applications (DICTA)*, 1–8. IEEE. https://doi.org/10.1109/DICTA51227.2020.9363418.

[15] Islam, Md Azharul, Shawkh Ibne Rashid, Niamat Ullah Ibne Hossain, Robert Fleming, and Alexandr Sokolov. 2023. "An Integrated Convolutional Neural Network and Sorting Algorithm for Image Classification for Efficient Flood Disaster Management." *Decision Analytics Journal* 7: 100225. https://doi.org/10.1016/j.dajour.2023.100225.

[16] Patil, Shivam, Suraj Sawant, and Amit Joshi. 2023. "Flood Detection Using Remote Sensing and Deep Learning Approaches." In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 1–6. IEEE. https://doi.org/10.1109/ICCCNT56998.2023.10306978.

[17] Huang, Haocheng, Xiaohui Lei, Weihong Liao, Haichen Li, Chao Wang, and Hao Wang. 2023. "A Real-Time Detecting Method for Continuous Urban Flood Scenarios Based on Computer Vision on Block Scale." *Remote Sensing* 15 (6): 1696. https://doi.org/10.3390/rs15061696.

[18] Jackson, Jehoiada, Sophyani Banaamwini Yussif, Rutherford Agbeshi Patamia, Kwabena Sarpong, and Zhiguang Qin. 2023. "Flood or Non-Flooded: A Comparative Study of State-of-the-Art Models for Flood Image Classification Using the FloodNet Dataset with Uncertainty Offset Analysis." *Water* 15 (5): 875. https://doi.org/10.3390/w15050875.

[19] Mittal, Bhavuk, Jui Vanzara, and S. A. Sajidha. 2024. "Navigating Spatial Insights: Comparative Exploration of Deep Learning Algorithms for Flood Detection Using Remote Sensing Data." In *2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, vol. 2, 1–6. IEEE. https://doi.org/10.1109/IATMSI60426.2024.10502981.

[20] Yasi, Ellora, Tasnim Ullah Shakib, Nusrat Sharmin, and Tariq Hasan Rizu. 2024. "Flood and Non-Flood Image Classification Using Deep Ensemble Learning." *Water Resources Management*: 1–18. https://doi.org/10.1007/s11269-024-03906-9.

[21] Simonyan, Karen, and Andrew Zisserman. 2014. "Very Deep Convolutional Networks for Large-Scale Image Recognition." arXiv preprint arXiv:1409.1556. https://doi.org/10.48550/arXiv.1409.1556.

[22] Jain, Pallavi, Bianca Schoen-Phelan, and Robert Ross. 2020. "Automatic Flood Detection in Sentinel-2 Images Using Deep Convolutional Neural Networks." In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, 617–623. https://doi.org/10.1145/3341105.3374023.

[23] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Identity Mappings in Deep Residual Networks." In *Computer Vision – ECCV 2016*, edited by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, 630–645. Lecture Notes in Computer Science. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-46493-0_38.

[24] Wu, Luyuan, Yunxiu Liu, Jianwei Zhang, Boyang Zhang, Zifa Wang, Jingbo Tong, Meng Li, and Anqi Zhang. 2024. "Identification of Flood Depth Levels in Urban Waterlogging Disaster Caused by Rainstorm Using a CBAM-Improved ResNet50." *Expert Systems with Applications*: 124382. https://doi.org/10.1016/j.eswa.2024.124382.

[25] Agung, Andi Sadri, Satria Gunawan Zain, Fhatiah Adiba, Dyah Darma Andayani, and Andi Baso Kaswar. 2023. "Urban Flood Disaster Mitigation through Image Classification Using Transfer Learning Method with MobileNet Fine-tuning." In *2023 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, 364–369. IEEE. https://doi.org/10.1109/ICAMIMIA60881.2023.10427560.

[26] Rahnemoonfar, Maryam, Tashnim Chowdhury, Argho Sarkar, Debvrat Varshney, Masoud Yari, and Robin Roberson Murphy. 2021. "Floodnet: A High Resolution Aerial Imagery Dataset for Post Flood Scene Understanding." *IEEE Access* 9: 89644–89654. https://doi.org/10.1109/ACCESS.2021.3090981.