ECTI Transactions on Computer and Information Technology
Journal homepage: https://ph01.tci-thaijo.org/index.php/ecticit/
Published by the ECTI Association, Thailand, ISSN: 2286-9131

A Neural Architecture Search CNN for Alzheimer's Disease Classification

Nicodemus Songose Awarayi[1], Frimpong Twum[2], James Ben Hayfron-Acquah[3] and Kwabena Owusu-Agyemang[4]

ABSTRACT

The evolution of automated machine learning (AutoML) is gradually reengineering the design of deep learning architectures for various imaging tasks. AutoML effectively develops model architectures and tunes hyperparameters through neural architecture search (NAS). Deep learning model architecture design is generally considered a tedious and time-consuming task that requires mastery skills to develop robust and better-performing models for imaging tasks. Again, the model's hyperparameters must be well-tuned to ensure optimal performances, which can be tedious and time-consuming if the hyperparameters are manually selected; using existing hyperparameter optimization algorithms can be expensive regarding resources. This study addresses these challenges in developing an optimal convolutional neural network (CNN) for classifying Alzheimer's (AD). The study, therefore, adopted a NAS approach to generate a CNN model architecture using a customized search space comprising only CNN patterns implemented with a NAS framework. The search was done for ten (10) trials, yielding a CNN architecture with an accuracy of 95.85% and a loss of 0.22. Training the model with a 10-fold cross-validation approach using a 0.0009 learning rate for 150 epochs improved the model's performance. The model recorded 97.17% accuracy, 97.21% precision, 97.14% recall, and a 0.99 area under the curve (AUC) in classifying AD as one of AD, mild cognitive impairment (MCI), and normal control (NC). The model obtained 98.06%, 98.66%, and 98.62% accuracy on binary classes of AD/NC, AD/MCI, and NC/MCI, respectively. The model generally showed robustness and better performance than existing CNN architectures.

Article information:

Keywords: CNN, Neural Architecture Search, Alzheimer's Disease, Automated Machine Learning, Classification

Article history:
Received: February 15, 2024
Revised: June 28, 2024
Accepted: September 23, 2024
Published: October 12, 2024 (Online)

DOI: 10.37936/ecti-cit.2024184.255728

1. INTRODUCTION

Convolutional neural network (CNN) has been primarily used in analyzing image data in various ways, such as classification [1]–[3], prediction [4]–[6], object detection [7]–[9], image segmentation [10]–[13] and many more. CNNs have contributed to solving several research problems in various domains, including medical imaging for computer-aided diagnosis [14]–[16], plant disease detection [17]–[19], robotic [14], [20], [21] and many more. In these tasks, it is pertinent to have very accurately trained models to achieve optimal benefits from artificial intelligence solutions such as [22], [23]; therefore, improving model accuracy has been one of the main problems in deep learning research.

Researchers have identified several approaches to attaining enhanced model performance in terms of accuracy, with some focusing on building novel model architectures such as ResNet [24], MobileNet [25], VGG16 [26], and many others. Some have employed techniques such as data augmentation [27]–[29], rigorous data preprocessing [30], [31], and data sampling techniques [32] to improve the predictive accuracy of deep learning models.

Aside from considering these techniques, there is again a tussle in selecting optimal hyperparameters

[1,2,3,4]The authors are with the Department of Computer Science, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana. Email: nicodemus.awarayi@uenr.edu.gh, twumf@yahoo.co.uk, jbha@yahoo.com and oljacko@gmail.com
[1]The author is with the Department of Computer Science and Informatics, University of Energy and Natural Resources, Sunyani, Ghana.
[1]Corresponding author: nicodemus.awarayi@uenr.edu.gh

for the model. Tuning hyperparameters of the model impacts the model's performance, hence the need to carefully select these hyperparameters [33], [34]. Hyperparameters are model variables that the model relies on during training to learn data examples, which include the learning rate, batch size, optimizers, number of epochs and others depending on the selected model. For instance, the learning rate helps update the weights and bias of a neural network to reduce the training loss and increase the model accuracy. The batch size indicates the number of examples used in forward and backward propagation during training. The model trains well when these parameters are correctly tuned, producing good results. Hyperparameter tuning can be a problem in achieving accurate models if the optimal hyperparameter values are not selected; as such, most researchers use various ways of dealing with them. Some researchers rely on a try-and-error approach, and others use hyperparameter optimization algorithms, which are expensive and time-consuming.

Therefore, developing good-performing CNN models is an excessively cumbersome activity. A CNN model consists of one to many layers stacked together, each with several parameters. The developers must make several decisions, including the number of layers and blocks, how the layers are stacked, the number of filters in a layer, and all other vital hyperparameters.

There has been an attempt to automate these processes using neural architecture search (NAS), a subsidiary of automated machine learning. NAS aids model development by searching a search space using a search algorithm to automate and eliminate manual model development processes [35]. The search algorithms recursively search the search space comprising various architectures or components of a model to formulate a novel model evaluated using an evaluation criterion. NAS models have been applied to various image classification tasks [36]–[38], image segmentation [39], and many more, which, in most cases, produce competitive results compared to some state-of-the-art manually developed models.

Though NAS models have shown prominence in image segmentation and classification tasks in some medical imaging applications, work still needs to be done on using NAS-generated models to classify AD on image datasets. Our search identified a single study [40] that explored NAS with multimodal fusion to classify AD from spontaneous speech. The study used the DARTS model to search for the NAS CNN model from a speech-recorded AD dataset. Their method recorded an accuracy of 91.66% in detecting AD.

Therefore, the motivation of this work is to automate the development of a CNN model architecture for classifying AD by leveraging neural architecture search, which addresses some of the challenges of developing and tuning model hyperparameters. The approach deployed in this study showed uniqueness as NAS is yet to be used to develop AD classification models. No existing work has employed NAS to generate a CNN model for classifying AD using an image dataset. The main contribution of this study includes:

- We designed an automated CNN model using the NAS framework to classify AD. The search algorithm was configured on a customized CNN patterns search space.
- A k-fold cross-validation technique sampled the data used to train the CNN model, further improving its accuracy and significantly boosting its performance.

The remainder of the study is as follows: Section 2 presents a literature review, delving into current methods of AD classification. In Section 3, we introduce our proposed method, detailing its implementation and the criteria for its evaluation. Section 4 presents the results of our model's evaluation. Section 5 offers a discussion, comparing our results with existing methods. Finally, Section 6 is the conclusion of the study.

2. LITERATURE REVIEW

Xin *et al.* [41] contributed to early AD diagnosis by proposing a classifier based on CNN and a swin-transformer. The model could transform 3D data into 2D features, enhance generalizability, and train and learn effectively. The model was trained on the AD Neuroimaging Initiative (ADNI) dataset and tested on the AIBL datasets, recording an accuracy of 92.8%.

Thangavel *et al.* [42] designed an EAD-DNN, an early AD classifier with the ResNet50 architecture, which acquires vital information from the network level. The study also adopted a modified Adam optimizer to choose the best features from the image dataset. The model recorded 98% accuracy in predicting AD.

Ambili *et al.* [43] implemented another method based on label propagation involving deep learning to classify AD effectively and recorded 97% accuracy on multiclass. Similarly, the work conducted by Davuluri et al. [44] using the VGG19 model also recorded 87% accuracy in classifying AD.

Agarwal *et al.* [45] implement an effective method using a proposed fusion of end-to-end and transfer learning approaches to automate the diagnosis of AD. The approach obtained a test accuracy of 87.38% on multiclass classification and 93.10% on binary classes of mild cognitive impairment (MCI) and AD classes.

Long *et al.* [46] built a model to predict AD based on quantifying magnetic resonance imaging (MRI) deformation. The method computed and analyzed the regional morphological differences in the brain between the AD and MCI groups. The model recorded 96.5% accuracy in distinguishing AD from normal controls (NC) and 88.99% on MCI versus AD.

Marzban *et al.* [47] used a CNN model to diagnose AD from tensor images. Aiming for a robust classification of AD and MCI against NC, they utilized a cost-effective network with a shallow architecture. A ten-fold cross-validation approach to train and test the model obtained 93.5% and 79.6% accuracy for AD/NC and MCI/NC, respectively.

Punjabi *et al.* [48] employed a modality fusion approach using a CNN model, combining MRI and Positron Emission Tomography (PET) images. This study illuminated the benefits of modality fusion for future deep learning models in AD classification, achieving 92.34% accuracy with a 1.95% standard deviation.

Shanmugam *et al.* [49] used a pre-trained deep-learning model to aid AD detection. The study focused on detecting various stages of MCI and AD by using neuroimages with transfer learning. Different pre-trained models such as ResNet, AlexNet, and GoogleNet were deployed in the experiment, recording accuracy of 97.51%, 94.08%, and 96.39%, respectively.

Shamrat *et al.* [50] also deployed a pre-trained model to predict AD. The approach employed an image enhancement algorithm to preprocess the dataset and data augmentation to manage unbalanced data. The training and evaluation of the chosen pre-trained models, MobileNetV2, AlexNet, ResNet50, and InceptionV3, attained test accuracies of 78.84%, 86.85%, 78.87%, 80.98%, and 96.31%, respectively.

Srinivasan *et al.* [51] introduced a DHO-based pre-trained CNN model, utilizing structural MRI data to combine automatic hippocampus segmentation with AD categorization. They employed a deer hunting optimization algorithm for hyperparameter selection and optimization. Their model achieved 96% accuracy in binary classification and 93% in multiclass categorization.

Hajamohideen *et al.* [52] proposed a four-way AD classification using a deep Siamese CNN with a triple-loss function. They implemented both pre-trained and non-pre-trained CNNs for image transformation. The model achieved 91.83% and 93.85% accuracy on the ADNI and OASIS datasets.

Asgharzadeh-bona *et al.* [53] presented an AD classification technique using brain MRI transforms and deep CNN features. They demonstrated that feature-level fusion using EfficientNet-B7 and the ANN classifier was the most effective, achieving accuracies of 82.7%, 89.7%, and 84.3% for CN/MCI, CN/AD, and MCI/AD, respectively.

Sethi and Ahuja [54] implemented VGG19 and ResNet50 pre-trained models for AD classification. The experimental results proved that VGG19 performed better than ResNet50, with an accuracy of 93.89% and 92.89% for AD/NC and NC/MCI, respectively.

Raghul and Kasim [55] developed a dual-tree complex wavelet transform-based AD classification. To evaluate their performance, the study compared support vector machine (SVM) and artificial neural networks (ANN) models. The SVM performed better, with an accuracy of 96.3%

Samantha *et al.* [56] achieved 91.65% accuracy in detecting early-stage AD using an image enhancement filter combined with CNN. They applied a random up sampler and Gaussian filter to the images and then constructed and trained a simple CNN classifier with the processed images. Joshi *et al.* [57] developed a multilabel classifier for AD using the DenseNet-169 architecture. This model, trained on an MRI image database containing images of patients diagnosed with AD, achieved an accuracy of 91.80%.

Lu *et al.* [58] employed the ConvNeXt network to classify AD by extracting features from AD images. They further parameterized the subject's identity information using Dynamic Multilayer Perceptron (DMLP) and mapped it onto the image features for enhancement. This approach achieved a 78.95% accuracy rate.

Kaur *et al.* [59] introduced an AD detection method using a weighted K-nearest neighbour (KNN) classifier and compared it to a medium KNN classifier. The weighted KNN classifier reported a mean accuracy of 96.59%, while the medium KNN reached 94.68%.

Wen *et al.* [60] presented a fine-grained classification for AD using a wavelet convolution unit network. They introduced a unique wavelet convolution unit to integrate wavelet analysis with standard convolution operations for more efficient deep feature extraction. This novel network achieved an impressive 97.89% accuracy in AD classification.

Archana and Kalirajan [61] also employed a deep-learning approach to classify brain neuroimages into MCI, AD, and CN. The results showed the CNN model classifying images with 95.82% accuracy.

Xiao Liu *et al.* [62] developed a neural architecture search (NAS) model capable of segmenting and classifying brain tumours. This model features a nested transformer U-shape NAS network for segmentation, which predicts tumours from multimodal MRI images. Additionally, it incorporates multiscale features in the encoder of the segmentation model, serving as input features for classification. The model achieves a classification accuracy of 0.941.

Ji and Wang [63] designed a CNN architecture model that outperforms some state-of-the-art manually designed CNN models for classifying functional brain networks. Their CNN model is automated using a particle swarm optimization algorithm developed based on a three-phase procedure. This procedure includes individual expression, which extracts brain topological features with a convolutional layer, an evaluation phase that assesses the previous phase,

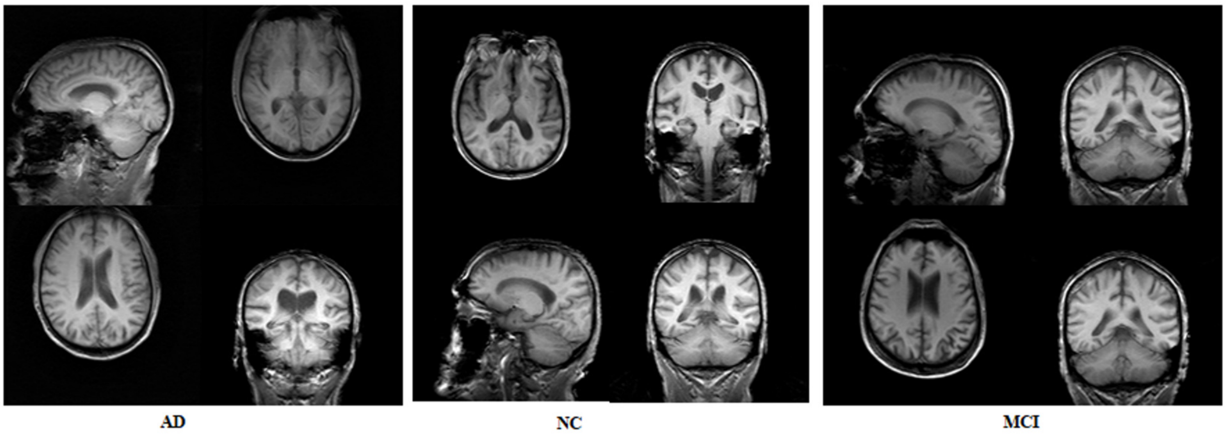


Fig.1: *Sample of ADNI MRI dataset of AD/MCI/NC.*

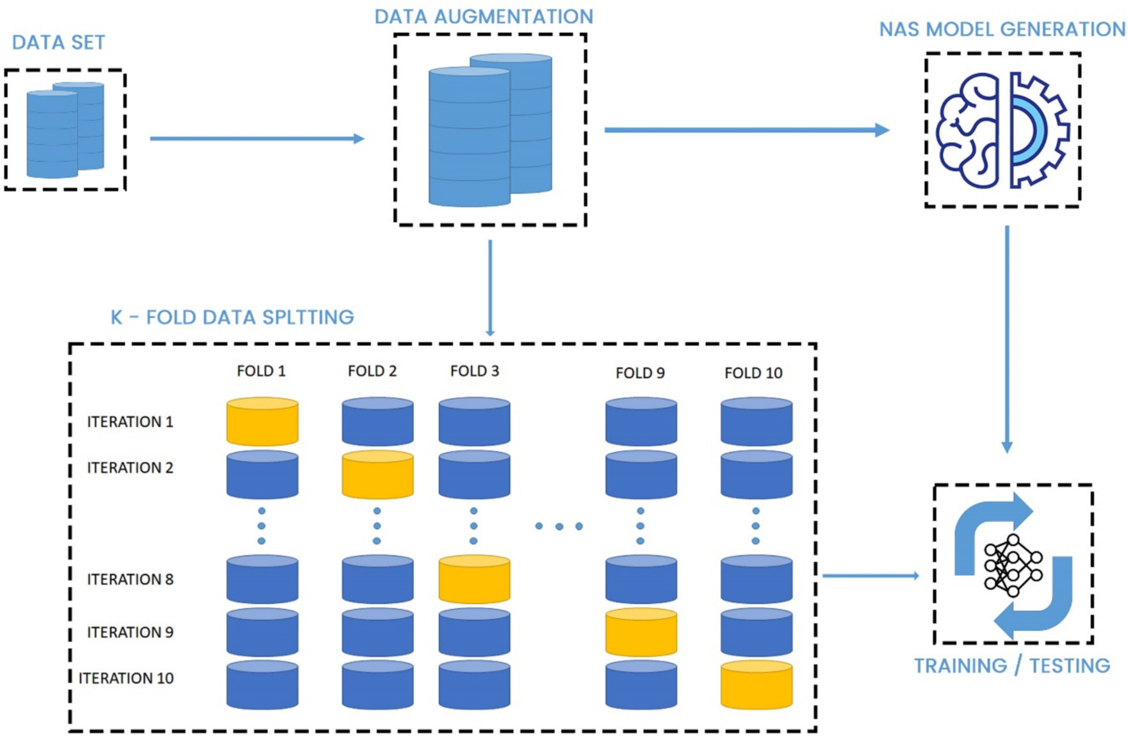


Fig.2: *Proposed approach for training and testing the NAS model.*

and an update phase that optimizes the individuals in the particle swarm.

A cross-task NAS model by [64] automated the network structure design across tasks for detecting electroencephalogram (EEG) signals. This NAS model deployed an efficient search space for cross-tasks and a search method that effectively addresses the challenges of EEG signal processing. Ma et al.[65] also used NAS to design a CNN model to classify malignant and benign lesions using cone-beam CT images. Their proposed model performed significantly better than a tuned ResNet-50 implemented in their study.

3. METHOD AND MATERIALS

3.1 Subjects/Datasets

The dataset was obtained from the ADNI database [66], a public access control data repository. This repository was established through a public-private collaboration to slow the progression from MCI to AD using MRI, PET, and other biomarkers. Initially, the images were in DICOM format and subsequently converted to JPG. Three class labels were collected and used in the experiment, comprising 1581 AD images, 1,310 MCI images, and 1,591 NC images. Each image was initially dimensioned at 256 $\times$ 256 pixels and later resized to 64 $\times$ 64 pixels. Fig. 1 presents sample images of the dataset.

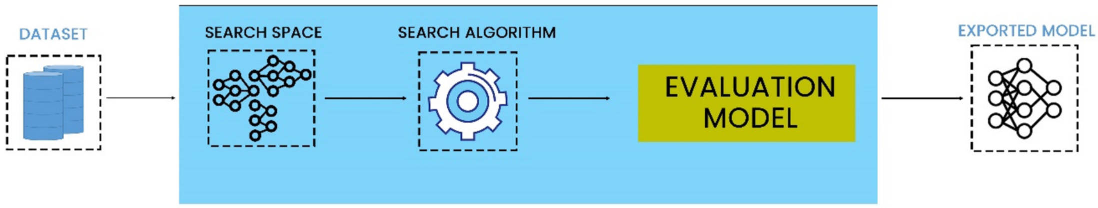


Fig.3: *Neural architecture search procedure for selecting the CNN model.*

Again, a Python script generated more data to augment the existing data and increase the dataset's volume. The script used rotation and flipping techniques to create the augmented dataset, increasing the AD images to 9486, the MCI to 7860, and the NC images to 9546.

3.2 Method

This study explores using NAS to eliminate the difficulties in model architecture design, time required in model development, manual tuning of hyperparameters, and applying the same to AD classification. The study, therefore, leveraged the AutoKeras NAS framework to automate the design of a CNN model for the classification. AutoKeras is an automated machine learning (AutoML) library that streamlines model selection and hyperparameter tuning [67]. The training and assessment of the automated CNN model used the K-fold cross-validation method. The technique partitions the dataset into K-folds; one-fold is used for testing, while the rest are for training. Fig.2 indicates the detailed outline of the method, including techniques such as data augmentation, k-fold data split, model training, and evaluation.

3.3 Model Generation

This study used a NAS approach to build an automated CNN model using the autoKeras NAS framework. Building a NAS model consists of constructing a search space, searching the search space to construct a model, and then evaluating the model using an evaluation criterion to select the best model. The search space includes a custom search space with CNN operators for the search process. When selecting a model, the autoKeras library relies on a series of carefully constructed and in-built search spaces comprising existing state-of-the-art deep learning models. The search algorithm implemented for the NAS task is a novel algorithm inspired by the hill-climbing algorithm.

In designing the automated model, the NAS framework learns from the input dataset and the search space to generate a suitable NAS model with tuned hyperparameters. The hill-climbing-inspired search algorithm carefully examines the search space to construct and evaluate a model over fewer epochs. The experiment runs recursively, corresponding to the number of trials indicated. The best model is selected and exported at the end of the trials. The diagram in Fig. 3 depicts the model's selection process.

3.4 Experiment Setup

The experiment was configured and run on a Windows 10 local machine. The machine has an NVIDIA GeForce GTX 1060 Graphic Processing Unit (GPU) with 8 GB of memory allocation. The model was implemented in Python using the TensorFlow frameworks.

3.5 Evaluation Metrics

Evaluating a deep learning model during training is essential to gauge its performance on data it has not seen. Several metrics can quantify a model's performance and help distinguish between models. In this study, we employed evaluation metrics like accuracy, precision, and recall derived from the true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) of the model's confusion matrix. TP refers to the number of positive class labels accurately predicted as positive, TN signifies those correctly identified as negatives, FP indicates negatives mistakenly labelled as positives, and FN denotes positives incorrectly labelled as negatives.

The accuracy metric presented in (1) aggregates the model's correctly predicted labels.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

As shown in (2), precision is the ratio of true positives to the total number of predictions by the model.

$$precision = \frac{TP}{Tp + TN} \tag{2}$$

Recall calculates the true positive rates as presented in (3).

$$recall = \frac{TP}{TP + FN} \tag{3}$$

An additional metric, the area under the curve (AUC), was utilized to evaluate the model. The AUC represents the integral of the receiver-operating characteristic curve (ROC). It corresponds to the probability that a randomly chosen positive sample is ranked higher than a randomly chosen negative sample. Since the ROC curve is formed by successively connecting coordinates, (4) calculates the AUC.

$$AUC = \frac{1}{2}\sum_{i=1}^{m-1}(x_{i-1} - x_i)(y_i + y_{i+1}) \quad (4)$$

During the neural architecture search to design the CNN model, the best model was selected based on the loss function result, and the model that recorded the most minor loss was selected. The search algorithm generates several models based on the number of trials specified and the best model chosen based on the loss.

4. RESULTS

This study sought to explore NAS to generate an automated model for classifying AD into three class labels: AD, MCI and NC. This experiment used ten (10) trials to produce the best version of a CNN architecture. Each trial produces a unique CNN architecture with autotuned hyperparameters. The best architecture presented in Table 1 recorded a loss of

Table 1: *The layout of the best NAS CNN model.*

Layer(type)	Output Shape	Parameters
Input_1(input Layer)	(None, 46,64,3)	0
Cast_to_float32( )	(None, 46,64,3)	0
Conv2d(Conv2D)	(None, 62,62,32)	896
Conv2d_1(Conv2D)	(None, 60,60,32)	9248
Max_pooling2d( )	(None, 30,30,32)	0
Dropout (Dropout)	(None, 30,30,32)	0
Conv2d_2(Conv2D)	(None, 28,28,32)	9248
Conv2d_3(Conv2D)	(None, 26,26,16)	4624
Max_pooling2d_1( )	(None, 13,13,16)	0
Dropout_1(Dropout)	(None, 13,13,16)	0
Flatten (Flatten)	(None, 2704)	0
Dense (Dense)	(None, 64)	173120
Re_lu(ReLu)	(None, 64)	0
Dense_1 (Dense)	(None, 32)	2080
Re_lu_1(ReLu)	(None, 32)	0
Dense_2 (Dense)	(None, 3)	99
Classification_head (Soft max)	(None, 3)	0
Total parameters		199,315
Trainable parameters		199,315
Non-trainable parameters		0

Table 2: *Test results on classifying AD, MCI and NC.*

Fold	Loss	Accuracy (ACC) (%)	Precision (Prec) (%)	Recall (Rec) (%)	AUC
1	0.14	97.51	97.54	97.43	0.99
2	0.17	97.23	96.99	96.93	0.99
3	0.16	97.38	96.86	96.82	0.99
4	0.19	96.82	97.16	97.14	0.99
5	0.15	97.29	97.08	97.01	0.99
6	0.20	97.02	97.32	97.27	0.99
7	0.16	97.14	96.89	96.80	0.99
8	0.18	96.86	97.41	97.34	0.99
9	0.19	96.93	97.25	97.21	0.99
10	0.15	97.49	97.58	97.49	0.99
Average	0.17	97.17	97.21	97.14	0.99
Standard deviation	0.02	0.24	0.24	0.23	0.001

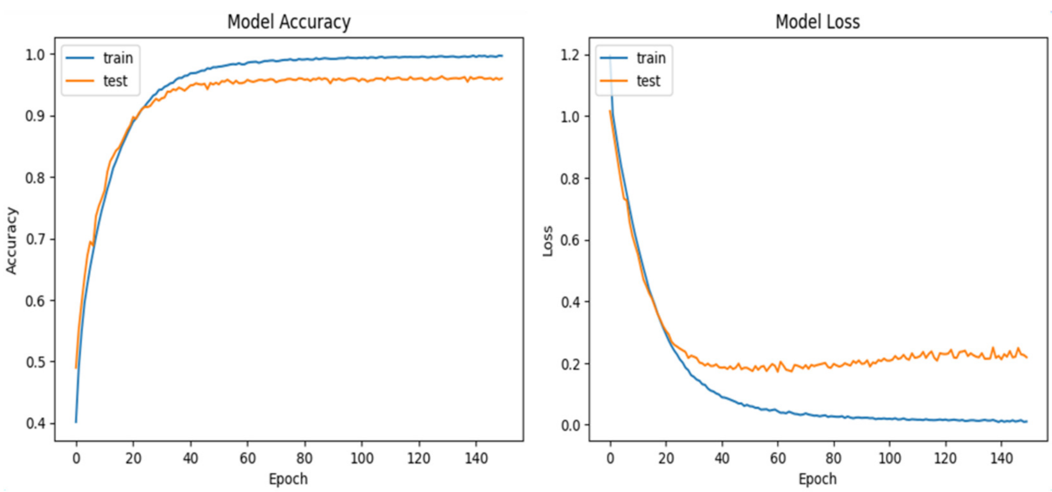
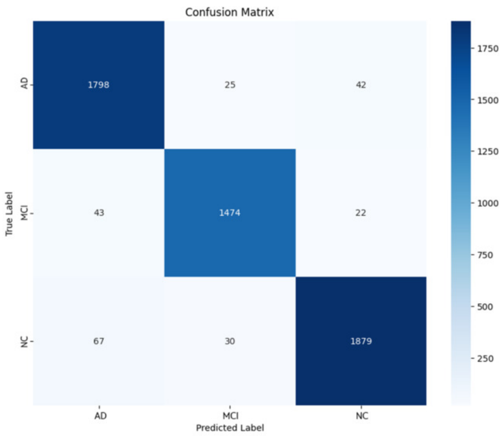


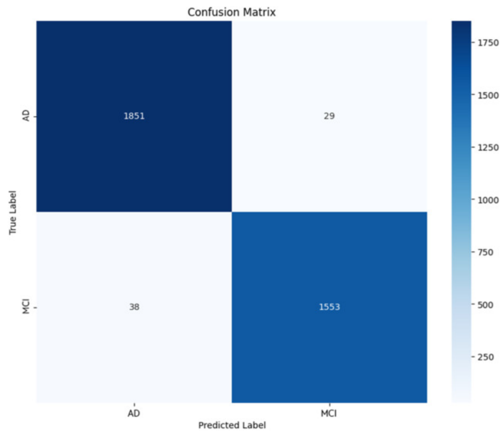
Fig.4: *Training accuracy and loss curve of the NAS model.*

Table 3: *Test results on binary classification of Alzheimer's disease.*

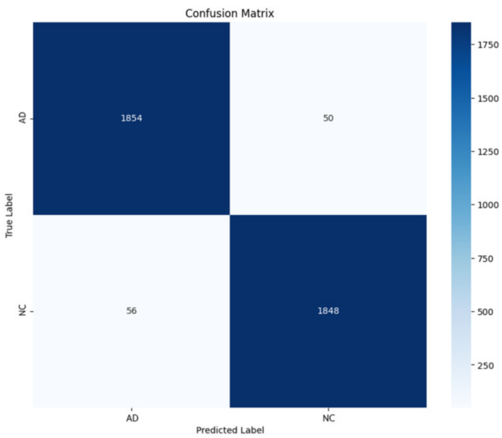
Fold	AD vs NC			AD vs MCI			MCI vs NC		
	Loss	ACC	AUC	Loss	ACC	AUC	Loss	ACC	AUC
1	0.09	98.19	0.99	0.09	98.50	0.99	0.06	98.97	1.00
2	0.08	98.40	0.99	0.10	98.47	0.99	0.10	98.68	0.99
3	0.13	97.82	0.99	0.10	98.67	0.99	0.10	98.68	0.99
4	0.10	98.06	0.99	0.09	98.59	0.99	0.09	98.39	0.99
5	0.14	97.98	0.99	0.12	98.27	0.99	0.10	98.39	0.99
6	0.13	97.90	0.99	0.09	98.82	0.99	0.06	98.62	0.99
7	0.09	98.13	0.99	0.07	98.82	1.00	0.11	98.59	0.99
8	0.14	97.79	0.99	0.07	98.79	1.00	0.06	98.88	1.00
9	0.11	98.27	0.99	0.05	98.82	0.99	0.10	98.53	0.99
10	0.10	98.11	0.99	0.07	98.88	1.00	0.10	98.56	0.99
Average	0.11	98.06	0.99	0.09	98.66	0.99	0.09	98.62	0.99
Standard deviation	0.02	0.19	0.002	0.02	0.19	0.001	0.02	0.18	0.002



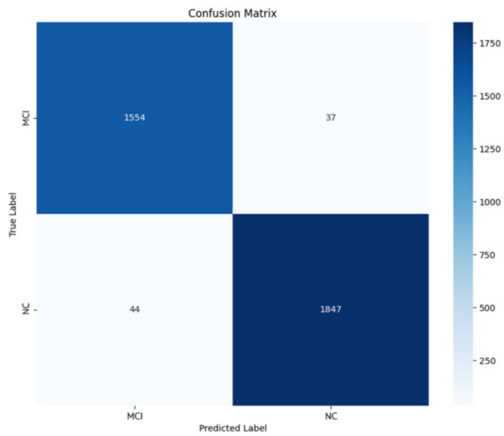
(a) AD/MCI/NC Confusion Matrix



(b) AD/MCI Confusion Matrix



(c) AD/NC Confusion Matrix



(d) MCI/NC Confusion Matrix

Fig.5: *Confusion matrix for the trained models.*

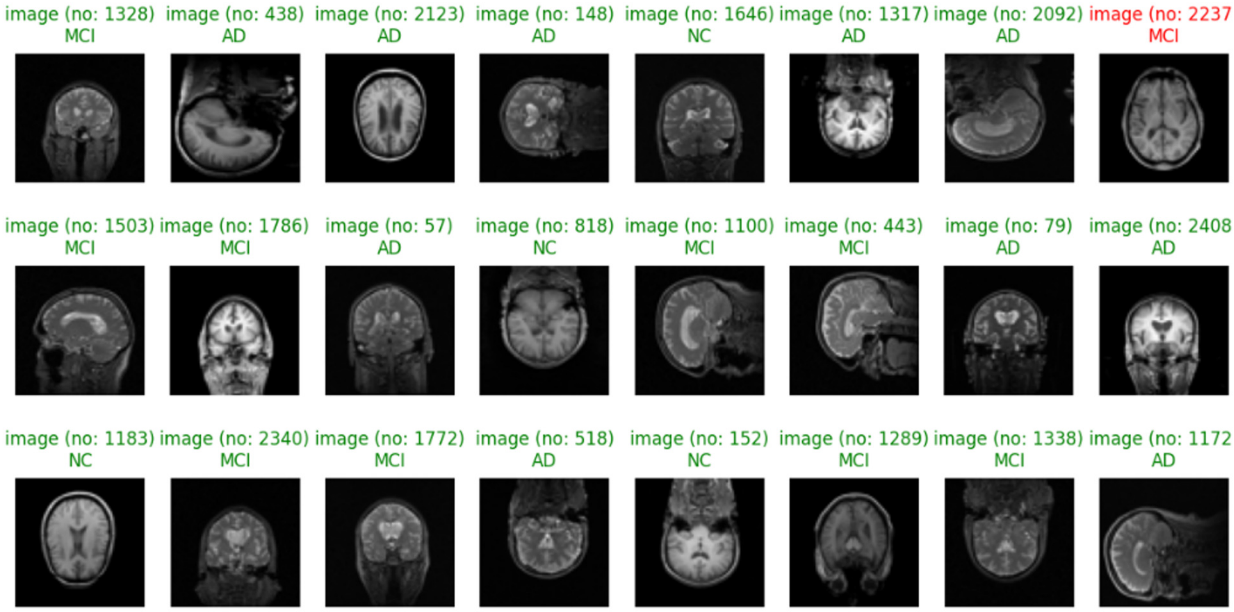


Fig.6: *Predicted Alzheimer's disease classification test samples. The green colour represents the true prediction, and the red represents the false prediction.*

Table 4: *Comparative analysis of proposed model and existing models.*

S/No.	Models	Classification			
		AD/MCI/NC (%)	AD/NC (%)	AD/MCI (%)	MCI/NC (%)
1	[40]	-	91.66	-	-
2	[41]		92.8		
3	[42]	-	98	-	-
4	[43]	97.0	-	-	-
5	[45]	87.38	-	93.10	-
6	[44]	87	-	-	-
7	[50]	96.31	-	-	-
8	[53]	-	89.7	84.3	82.7
9	[54]	-	93.89	-	92.89
10	[59]	-	96.59	-	-
11	Proposed model	97.17	98.06	98.66	98.62

0.22 and an accuracy of 95.85%. Fig. 4 presents the training accuracy and loss curve of the model. The architecture comprises four convolutional layers, two max-pooling layers, dropouts, and two fully connected layers. In the model architecture, every two convolution layers pair with a max-pooling and dropout operation. The convolution operations utilized a three-by-three kernel filter, a valid padding, a rectified linear unit (ReLU) activation and a stride of 1. The fully connected layers comprised two hidden layers, each with a ReLU activation.

The study uses a 10-fold cross-validation data sampling technique to train and improve the model's classification accuracy. This sampling technique effectively uses all of the datasets in training and testing. The 10-fold implies the model trains ten (10) times, where, at each instance, 1-fold of the data split is for testing. The model trains for 150 epochs with a batch size of 112 and learning at a rate of 0.0009. The test results for each iteration were recorded and averaged to determine the model's performance. Table 2 presents the test results of the model. The results indicate that the model classifies the AD with an accuracy of 97.17% and a standard deviation of 0.24, a precision of 97.21%, a recall of 97.14%, and an AUC of 0.99. When the classes were binarized, the model recorded an accuracy of 98.06%, 98.66%, and 98.62% for AD/NC, AD/MCI, and NC/MCI, respectively. The detailed results of the binary classification are in Table 3. Fig. 5 also presents the confusion matrix of the various trained models. Fig.6 presents the model's prediction on a randomized test sample.

5. DISCUSSION

In this study, we demonstrated using NAS in developing a CNN model for AD classification. NAS presents a new dimension in model development, offering the opportunity to automate the process of new model development for various tasks such as image classification, image segmentation, objection detection, and many more. Using NAS in model architecture designs has significantly reduced the time and expertise required to develop competitive deep-learning models. It has also proven effective in selecting hyperparameters for optimal model performance. We implemented a NAS approach to generate a model for classifying AD to address the difficulty in model development, eliminating manual hyperparameter tuning and reducing the time required to develop new architectures. The results recorded in this experiment from Table 2 and Table 3 demonstrate the ability of the NAS to effectively perform an architecture search for generating deep learning models with outstanding performance.

Though NAS applications exist for various medical imaging tasks, exploring a NAS approach to classifying AD using an image dataset has yet to be done. Chatzianastasis et al. [40] attempted to use the NAS with multimodal fusion to predict AD from a speech/audio dataset and made good headway. Their method achieved an accuracy of 91.66% in identifying two class labels, including AD and non-AD. Though their method was used on a speech/audio dataset as opposed to an image dataset, comparatively, our proposed approach showed superior performances in terms of the accuracy metric.

Comparing the proposed automated model with existing manually designed CNN model architectures showed that the NAS procedure can generate competitive architectures for classifying AD. We compared the works of [41]–[45], [50], [53], [54], and [59], who manually created various CNN models to classify AD. The comparative analysis presented in Table 4 indicates the performance of the proposed NAS CNN model against the existing manually designed models. The results suggest that NAS effectively develops and tunes new deep-learning models for AD and other image classification tasks.

6. CONCLUSION

This study explored NAS for generating a convolutional neural network for classifying AD. A custom CNN search space was designed and searched using a NAS framework based on the hill-climbing algorithm to generate the NAS CNN model. This approach addresses the time-consuming challenges of manually designing CNN models and optimizing hyperparameters. Using a 10-fold cross-validation to train and evaluate the NAS-generated CNN model, it achieved an accuracy of 97.17% in classifying AD into three categories: AD, MCI, and NC. The model also achieved accuracies of 98.06%, 98.66%, and 98.62% for AD vs NC, AD vs MCI, and MCI vs NC, respectively. These results surpass some existing manually configured models for classifying AD, demonstrating the effectiveness of NAS in automating model design. Future studies will examine other NAS frameworks to create a broad search space and explore new search algorithms.

ACKNOWLEDGEMENT

We acknowledge the Alzheimer's Disease Neuroimaging Initiative (ADNI) (adni.loni.usc.edu) for granting us access to the ADNI dataset for this study. Although ADNI researchers played a role in data provision, they were not involved in the analysis or composition of this study. A comprehensive list of ADNI contributors is accessible here: `http://adni.loni.usc.edu/wpcontent/uploads/how\_to\_apply/ADNI\_Acknowledgement\_List.pdf`.

AUTHOR CONTRIBUTIONS

Conceptualization, N.S. and G.H.M.; methodology, N.S.; software, N.S.; validation, N.S. and G.H.M.; formal analysis, N.S.; investigation, N.S.; data curation, N.S.; writing—original draft preparation, N.S.; writing—review and editing, N.S. and G.H.M.; visualization, N.S. and G.H.M.; supervision, G.H.M.; funding acquisition, G.H.M. All authors have read and agreed to the published version of the manuscript.

References

[1] Y. Li *et al.*, "Hierarchical Reinforcement Learning for Sequencing Behaviors," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 2733, no. 6, pp. 2709–2733, 2018.

[2] Y. Li, H. Zhang, X. Xue, Y. Jiang, and Q. Shen, "Deep learning for remote sensing image classification: A survey," *WIREs Data Mining and Knowledge Discovery*, vol. 8, no. 6, pp. 1–17, 2018.

[3] J. Song, S. Gao, Y. Zhu, and C. Ma, "A survey of remote sensing image classification based on CNNs," *Big Earth Data*, vol. 3, no. 3, pp. 232–254, 2019.

[4] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, no. 1995, pp. 60–88, 2017.

[5] H. Aerts, "Artificial Intelligence in Lung Cancer: Ready to Implement?," pp. 1–16.

[6] A. Srinivasulu *et al.*, "RETRACTED ARTICLE: Advanced lung cancer prediction based on blockchain material using extended CNN," *Applied Nanoscience*, vol. 13, no. 2, p. 985, 2023.

[7] F. Sultana, A. Sufian, and P. Dutta, "A review of object detection models based on convolu-

tional neural network," *Intelligent Computing: Image Processing Based Applications*, vol. 1157, pp. 1–16, 2020.

[8] C. B. Murthy, M. F. Hashmi, N. D. Bokde, and Z. W. Geem, "Investigations of object detection in images/videos using various deep learning techniques and embedded platforms comprehensive review,"*Applied Sciences*, vol. 10, no. 9, 2020.

[9] H. Zhang and X. Hong, Recent progress on object detection: a brief review, vol. 78, no. 19. Multimedia Tools and Applications, 2019.

[10] A. Işin, C. Direkoğlu, and M. Şah, "Review of MRI-based Brain Tumor Image Segmentation Using Deep Learning Methods," *Procedia Computer Science*, vol. 102, pp. 317–324, 2016.

[11] S. Rehman, H. Ajmal, U. Farooq, Q. U. Ain, F. Riaz, and A. Hassan, "Convolutional neural network based image segmentation: a review," no. April 2018, p. 26, 2018.

[12] G. Du, X. Cao, J. Liang, X. Chen, and Y. Zhan, "Medical image segmentation based on U-Net: A review," *Journal of Imaging Science and Technology*, vol. 64, no. 2, pp. 1–12, 2020.

[13] T. Hoeser and C. Kuenzer, "Object detection and image segmentation with deep learning on Earth observation data: A review-part I: Evolution and recent trends," *Remote Sensing*, vol. 12, no. 10, 2020.

[14] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis,"*Journal of Big Data* , vol. 6, no. 113, 2019.

[15] D. R. Sarvamangala and R. V. Kulkarni, "Convolutional neural networks in medical image understanding: a survey," *Evolutionary Intelligence* , vol. 15, no. 1, pp. 1–22, 2022.

[16] M. Bharath Simha Reddy and P. Rana, "Biomedical image classification using deep convolutional neural networks - Overview," *IOP Conference Series: Materials Science and Engineering*, vol. 1022, no. 1, 2021.

[17] J. Lu, L. Tan, and H. Jiang, "Review on convolutional neural network (CNN) applied to plant leaf disease classification," *Agriculture*, vol. 11, no. 8, pp. 1–18, 2021.

[18] K. L. Narayanan *et al.*, "Banana Plant Disease Classification Using Hybrid Convolutional Neural Network," *Computational Intelligence and Neuroscience*, vol. 2022, p. 9153699, 2022.

[19] L. Falaschetti, L. Manoni, D. Di Leo, D. Pau, V. Tomaselli, and C. Turchetti, "A CNN-based image detector for plant leaf diseases classification," *HardwareX*, vol. 12, p. e00363, 2022.

[20] S. Cebollada, L. Payá, X. Jiang, and O. Reinoso, "Development and use of a convolutional neural network for hierarchical appearance-based localization," *Artificial Intelligence Review*, vol. 55, no. 4, pp. 2847–2874, 2022.

[21] Q. Zhang, J. Zhu, X. Sun, and M. Liu, "HTC-Grasp: A Hybrid Transformer-CNN Architecture for Robotic Grasp Detection," *Electron*, vol. 12, no. 6, 2023.

[22] E. Watanyulertsakul, P. Khanthong, and B. Deeka, "Dementia U-Care: Comprehensive Cognitive Screening Application for Seniors," *ECTI-CIT Transactions*, vol. 18, no. 2, pp. 207–221, Apr. 2024.

[23] F. Angelucci, A. R. Ai, L. Piendel, J. Cerman, and J. Hort, "Integrating AI in fighting advancing Alzheimer: diagnosis, prevention, treatment, monitoring, mechanisms, and clinical trials," *Current Opinion in Structural Biology*, vol. 87, p. 102857, Aug. 2024.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016, pp. 770–778, 2016.

[25] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," 2017, [Online]. Available: `http://arxiv.org/abs/1704.04861.`

[26] K. Simonyan and A. Zisserman, "Intense convolutional networks for large-scale image recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[27] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data* , vol. 6, no. 1, 2019.

[28] R. Poojary, R. Raina, and A. K. Mondal, "Effect of data-augmentation on fine-tuned cnn model performance," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, pp. 84–92, 2021.

[29] M. Das, M. Panda, and S. Dash, "Enhancing the Power of CNN Using Data Augmentation Techniques for Odia Handwritten Character Recognition," *Advances in Multimedia*, vol. 2022, 2022.

[30] M. Zhao, P. Shi, X. Xu, X. Xu, W. Liu, and H. Yang, "Identification System Using Different," 2022.

[31] D. A. Pitaloka, A. Wulandari, T. Basaruddin, and D. Y. Liliana, "Enhancing CNN with Preprocessing Stage in Automatic Emotion Recognition," *Procedia Computer Science*, vol. 116, pp. 523–529, 2017.

[32] S. H. S. Basha, V. Pulabaigari, and S. Mukherjee, "An information-rich sampling technique over spatiotemporal CNN for classification of human actions in videos," *Multimedia Tools and Applications*, vol. 81, no. 28, pp. 40431–40449, 2022.

[33] M. Gupta, K. Rajnish, and V. Bhattacharjee, "Impact of Parameter Tuning for Optimizing Deep Neural Network Models for Predicting

Software Faults," *Scientific Programming*, vol. 2021, p. 6662932, 2021.

[34] L. Liao, H. Li, W. Shang, and L. Ma, "An Empirical Study of the Impact of Hyperparameter Tuning and Model Optimization on the Performance Properties of Deep Neural Networks," *ACM Transactions on Software Engineering and Methodology*, vol. 31, no. 3, pp. 1-40, Apr. 2022.

[35] T. Elsken, J. H. Metzen, and F. Hutter, "Survey on neural architecture search," *Journal of Machine Learning Research*, vol. 26, no. 2, pp. 1–21, 2019.

[36] A. Kwasigroch, M. Grochowski, and A. Mikolajczyk, "Neural architecture search for skin lesion classification," *IEEE Access*, vol. 8, pp. 9061–9071, 2020.

[37] M. Ahmad, M. Abdullah, H. Moon, S. J. Yoo, and D. Han, "Image classification based on automatic neural architecture search using binary crow search algorithm," *IEEE Access*, vol. 8, pp. 189891–189912, 2020.

[38] Y. Weng, T. Zhou, L. Liu, and C. Xia, "Automatic Convolutional Neural Architecture Search for Image Classification under Different Scenes," *IEEE Access*, vol. 7, pp. 38495–38506, 2019.

[39] Y. Weng, T. Zhou, Y. Li, and X. Qiu, "NAS-Unet: Neural architecture search for medical image segmentation," *IEEE Access*, vol. 7, pp. 44247–44257, 2019.

[40] M. Chatzianastasis, L. Ilias, D. Askounis, and M. Vazirgiannis, "Neural Architecture Search with Multimodal Fusion Methods for Diagnosing Dementia," pp. 1–5, 2023.

[41] J. Xin, A. Wang, R. Guo, W. Liu, and X. Tang, "CNN and swing-transformer based efficient model for Alzheimer's disease diagnosis with sMRI," *Biomedical Signal Processing and Control*, vol. 86, p. 105189, 2023.

[42] P. Thangavel, Y. Natarajan, and K. R. Sri Preethaa, "EAD-DNN: Early Alzheimer's disease prediction using deep neural networks," *Biomedical Signal Processing and Control*, vol. 86, p. 105215, 2023.

[43] A. V. Ambili, A. V. Senthil Kumar, and R. Latip, "CNN-Mobilenetv2- Deep Learning-Based Alzheimer's Disease Prediction and Classification," *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 9, pp. 3590–3600, 2023.

[44] R. Davuluri, V. Mallapragada, U. M. R. Mamillapalli, M. M, and S. Peeka, "Alzheimer's Disease Diagnosis Using CNN Based Pre-trained Models," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 4, pp. 315–323, 2023.

[45] D. Agarwal, M. Á. Berbís, A. Luna, V. Lipari, J. B. Ballester, and I. de la Torre-Díez, "Automated Medical Diagnosis of Alzheimer's Disease Using an Efficient Net Convolutional Neural Network," *Journal of Medical Systems*, vol. 47, no. 1, 2023.

[46] X. Long, L. Chen, C. Jiang, and L. Zhang, "Prediction and classification of Alzheimer disease based on quantification of MRI deformation," *PLoS One*, vol. 12, no. 3, pp. 1–19, 2017.

[47] E. N. Marzban, A. M. Eldeib, I. A. Yassine, and Y. M. Kadah, "Alzheimer's disease diagnosis from diffusion tensor images using convolutional neural networks," *PLoS One*, vol. 15, no. 3, pp. 1–16, 2020.

[48] A. Punjabi, A. Martersteck, Y. Wang, T. B. Parrish, and A. K. Katsaggelos, "Neuroimaging modality fusion in Alzheimer's classification using convolutional neural networks," *PLoS One*, vol. 14, no. 12, pp. 1–14, 2019.

[49] J. V. Shanmugam, B. Duraisamy, B. C. Simon, and P. Bhaskaran,"Alzheimer's disease classification using pre-trained deep networks," *Biomedical Signal Processing and Control*, vol. 71, p. 103217, 2022.

[50] F. M. J. M. Shamrat, S. Akter, S. Azam, A. Karim, P. Ghosh, and Z. Tasnim, "AlzheimerNet: An Effective Deep Learning Based Proposition for Alzheimer' s Disease Stages Classification From Functional Brain Changes in Magnetic Resonance Images," *IEEE Access*, vol. 11, no. February, pp. 16376–16395, 2023.

[51] V. Srinivasan, J. N. Dwivedi, S. Raja, and N. Rajeswari, "Prediction of Alzheimer' s Disease Using DHO-Based Pretrained CNN Model Prediction of Alzheimer' s Disease Using DHO-Based Pretrained," *Mathematical Problems in Engineering*, Jun. 2023.

[52] F. Hajamohideen, N. Shaffi, M. Mahmud, K. Subramanian, and A. Al Sariri, "Four-way classification of Alzheimer's disease using a deep Siamese convolutional neural network with triplet-loss function," *Brain Informatics*, vol. 10, no. 5, 2023.

[53] A. Asgharzadeh-bona, H. Kalbkhani, and S. Azarfardian, "Healthcare Analytics An Alzheimer's disease classification method using a fusion of features from brain Magnetic Resonance Image transforms and deep convolutional networks," *Healthcare Analytics*, vol. 4, no. 100223, 2023.

[54] M. Sethi and S. Ahuja, "Alzheimer disease classification using MRI images based on transfer learning," vol. 040004, p. 040004, 2022.

[55] S. V. Raghul and M. B. M. Kasim, "Dual tree complex wavelet transform based detection and classification of Alzheimer's disease of MRI images using Artificial Neural Network (ANN) in comparison with Support Vector Machine (SVM)," *International Conference on Environmental Engineering and Material Sciences:*

Iceems-2021, vol. 2655, no.1, p. 020098, 2023.
[56] S. Samanta, I. Mazumder and C. Roy, "Deep Learning based Early Detection of Alzheimer's Disease using Image Enhancement Filters," *2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, Bhilai, India, pp. 1-5, 2023.
[57] R. Joshi, P. Negi and T. Poongodi, "Multilabel Classifier Using DenseNet-169 for Alzheimer's disease," *2023 4th International Conference on Intelligent Engineering and Management (ICIEM)*, London, United Kingdom, pp. 1-7, 2023.
[58] P. Lu, Y. Tan, Y. Xing, Q. Liang, X. Yan, and G. Zhang, "An Alzheimer's disease classification method based on ConvNeXt," *2023 3rd International Symposium on Computer Technology and Information Science (ISCTIS)*, Chengdu, China, pp. 884–888, 2023.
[59] M. Kaur, C. Thacker, L. Goswami, T. TR, I. S. Abdulrahman, and A. S. Raj, "Alzheimer's Disease Detection using Weighted KNN Classifier in Comparison with Medium KNN Classifier with Improved Accuracy," *2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Greater Noida, India, pp. 715–718, 2023.
[60] J. Wen, Y. Li, M. Fang, L. Zhu, D. D. Feng, and P. Li, "Fine-Grained and Multiple Classification for Alzheimer's Disease with Wavelet Convolution Unit Network," *IEEE Transactions on Biomedical Engineering*, vol. 70, no. 9, pp. 2592–2603, 2023.
[61] B. Archana and K. Kalirajan, "Alzheimer's Disease Classification using Convolutional Neural Networks," *Int. Conf. Innov. Data Commun. Technol. Appl. ICIDCA 2023 - Proc.*, pp. 1044–1048, 2023.
[62] X. Liu *et al.*, "BTSC-TNAS: A neural architecture search-based transformer for brain tumour segmentation and classification," *Computerized Medical Imaging and Graphics*, vol. 110, no. 102307, Jun. 2023.
[63] J. Ji and X. Wang, "Convolutional architecture search based on particle swarm algorithm for functional brain network classification[Formula presented]," *Applied Soft Computing*, vol. 149, no. 111049, Nov. 2023.
[64] Y. Duan, Z. Wang, Y. Li, J. Tang, Y. K. Wang, and C. T. Lin, "Cross task neural architecture search for EEG signal recognition," *Neurocomputing*, vol. 545, 2023.
[65] J. Ma *et al.*, "Distinguishing benign and malignant lesions on contrast-enhanced breast cone-beam CT with deep learning neural architecture search," *European Journal of Radiology*, vol. 142, no. 109878, Feb. 2021.
[66] ADNI, "ADNI — Alzheimer's Disease Neuroimaging Initiative," `https://adni.loni.usc.edu/` (accessed Jun. 21, 2023).
[67] H. Jin, F. Chollet, Q. Song, and X. Hu, "AutoKeras: An AutoML Library for Deep Learning," *Journal of Machine Learning Research*, vol. 24, pp. 1–6, 2023. [Online]. Available: `http://jmlr.org/papers/v24/20-1355.html`.

Nicodemus Songose Awarayi is a lecturer in the Department of Computer Science and Informatics at the University of Energy and Natural Resources, Ghana. His research interests include medical imaging, computer vision, and image processing. He earned his Bachelor of Science and Master of Philosophy degrees from Kwame Nkrumah University of Science and Technology (KNUST), Ghana. He is currently pursuing a PhD in Computer Science at KNUST.

Frimpong Twum earned his M.Sc. degree in Information Systems from Roehampton University, U.K., his B.Sc. degree in Electrical and Electronic Engineering, and another M.Sc. degree in Internet Engineering from London South Bank University, U.K. He obtained his Ph.D. in Computer Science from the Kwame Nkrumah University of Science and Technology (KNUST). Currently, he is an Associate Professor of Computer Science in the Department of Computer Science at KNUST. His research interests encompass computer network security, machine learning, and computer vision.

James B. Hayfron-Acquah is an Associate Professor of Computer Science and currently serves as the Head of the Department of Computer Science at KNUST. His research interests include Image Processing, Gait Recognition (Biometrics), Computer Programming, Cloud Computing, and Computer Vision. He earned his Master's degree in Computer Science and Application from Shanghai University of Science and Technology (SUST) in Shanghai, China, and his Ph.D. from Southampton University in the UK.

Kwabena Owusu-Agyemang is a lecturer in the Department of Computer Science and has extensive research and professional experience in machine learning and blockchain technology. He earned his doctoral degree in Software Engineering, specializing in Machine Learning, and his master's degree in Information Technology Management from Coventry University.