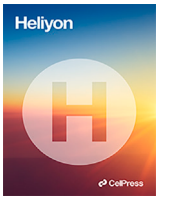
Heliyon 10 (2024) e34420

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



Research article

MedCapsNet: A modified Densenet201 model integrated with capsule network for heel disease detection and classification

Osamah Taher, Kasım Özacar *

Computer Engineering Department, Karabuk University, 78050, Karabuk, Turkey

A R T I C L E I N F O

Keywords:
Capsule network
DenseNet201
Heel spur
Sever
X-ray

A B S T R A C T

Conditions affecting the heel bone, such as heel spurs and sever's disease, pose significant challenges to patients' daily activities. While orthopedic and traumatology doctors rely on foot X-rays for diagnosis, there is a need for more AI-based detection and classification of these conditions. Therefore, this study addresses this need by proposing MedcapsNet, a novel hybrid capsule model combining modified DenseNet201 with a capsule network, designed to accurately detect and classify heel bone diseases utilizing lateral heel x-ray foot images. We conducted a comprehensive series of experiments on the proposed hybrid architecture with several datasets, including the Heel dataset, Breast BreaKHis v1, HAM10000 skin cancer dataset, and Jun Cheng Brain MRI dataset. The first experiment evaluates the proposed model for heel diseases, while the other experiments evaluate the model on a range of medical datasets to demonstrate its performance over existing studies. On the heel dataset, MedCapsNet achieves an accuracy of 96.38%, AUC of 98.35% without data augmentation, cross-validation accuracy of 95.69%, and AUC of 98.87%. The proposed model, despite employing a fixed architecture and hyperparameters, outperformed other models across four distinct datasets, including MRI, X-ray, and microscopic images with various diseases. This is notable because different types of medical image datasets typically require different architectures and hyperparameters to achieve optimal performance.

1. Introduction

Diseases affecting the heel bone can make it challenging for patients to carry out their daily activities. Typically, patients visit orthopedic and traumatology doctors who examine their foot X-ray images and diagnose the disease before recommending suitable treatments. The most common heel bone diseases are heel spur and Sever's disease [1]. While the heel spur, shown in Fig. 1a, is the most frequent bony exposure on the heel bone, Sever's disease, shown in Fig. 1b is an inflammation in young people due to excessive movement, causing heel pain. While traditionally specialists often rely on foot X-rays for diagnosing these diseases there is a growing need for AI-based solutions to enhance the accuracy and efficiency of detecting and classifying these diseases.

Despite extensive research on AI-based disease detection and classification [5–8], there is a notable lack of studies focusing on AI approaches for diagnosing heel bone diseases. Therefore, this paper aims to address this gap by utilizing Heel dataset [30] of 3956 lateral foot X-ray images and presenting a hybrid CapsNet model, MedCapsNet, designed to detect and classify heel bone diseases accurately. The hybrid model integrates a modified DenseNet201 with the original CapsNet architecture for detecting and classifying heel bone diseases without relying on data augmentation. The modifications made to the DenseNet201 architecture reduced

* Corresponding author.
E-mail address: kasimozacar@karabuk.edu.tr (K. Özacar).

https://doi.org/10.1016/j.heliyon.2024.e34420
Received 4 April 2024; Received in revised form 9 July 2024; Accepted 9 July 2024

Available online 14 July 2024

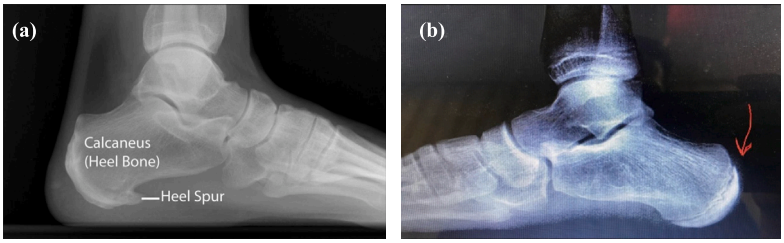


Fig. 1. Heel Spur and Sever disease adapted from [2].

the number of used hyperparameters and simplified the model. As a result, the modified architecture exhibits improved training efficiency, especially when dealing with smaller datasets and images with complex features, thereby eliminating the necessity for data augmentation.

5-fold cross-validation was used to train the model and the results show high accuracy of 95.69% and ROC AUC 98.87% for the heel dataset. Additionally, the proposed model for diagnosing heel diseases was evaluated by comparing its performance with various capsule models. Then, the model was tested on diverse medical datasets to demonstrate its superiority over existing methods. The results of the initial experiment indicated that current models, with their existing architectures and parameters, perform poorly on Heel dataset. In the second experiment, the proposed model consistently outperformed the other models across different types of images and diseases, without necessitating any changes to its architecture or hyperparameters.

The contributions of this study can be summarized as follows:

1. Proposing an improved capsule network that integrates a modified DenseNet201 with the original architecture, creating a model for detecting and classifying heel bone diseases without relying on data augmentation.
2. Using YOLO v8 on a Heel dataset to automatically detect regions of interest (ROIs).
3. Comparing the performance of the proposed hybrid model with other similar models and conducting an evaluation on heel dataset.
4. Providing detailed results of the model's performance on four medical datasets with measurements of 5-fold cross-validation accuracy.

2. Related work

Several research studies have shown that the original capsule network model lacks consistent performance across different datasets. It was originally designed and executed on the MNIST dataset, where pixels are represented by 0 and 1. However, this dataset has characteristics that are significantly different from other datasets [3,4]. Therefore, researchers have presented studies and proposals to enhance the performance of the original capsule network model by incorporating convolutional network models to increase the level of accuracy by strengthening the pre-capsule layers responsible for extracting features from the input.

DenseCapsNet [5], a hybrid CapsNet model, detects and classifies COVID-19 in 750 chest radiographs by integrating standard DenseNet121 and CapsNet. Since CapsNet, like a deep neural network, is weak in extracting deep features from images with complex features. The results indicate that our model achieved an accuracy of 90.7% and an F1 score of 90.9%. Additionally, the model demonstrated a sensitivity of up to 96% for detecting COVID-19. While the model has achieved a high accuracy, it is essential to note that this achievement was facilitated using data augmentation techniques in the dataset. Here, the integration of standard pre-trained models is insufficient to improve CapsNet to work without data augmentation.

DenseCaps [6], a capsule model network architecture based on the dense connections method, improves the processing of complex datasets at the capsule level by using cross-capsule feature concatenation, inspired by DenseNet's cross-layer feature concatenation. The accuracy of the model has been tested on well-known datasets, achieving 99.70% accuracy for MNIST, 94.93% for Fashion-MNIST, 89.41% for CIFAR-10, and 95.99% for Street View House Numbers (SVHN). These datasets contained many images; therefore, no data augmentation was required. Consequently, the model may not provide high accuracy for smaller datasets.

ResCaps [7] applies to the classification of papillary thyroid cancer and uses a residual module instead of a CNN layer in CapsNet to make the capsule more efficient with complex medical images, including 1956 training and 424 test images. The ResCaps network model improved classification accuracy to 81.06%. When they applied CNNCaps to the same dataset, they achieved an accuracy rate of 79.17%.

VGG-CapsNet [8], combines VGG-16 with CapsNet to enhance the performance and increase the accuracy and efficiency of lung cancer classification using CT images by leveraging the strengths of both CNNs and Capsule Networks. The study compared the performance of VGG-CapsNet with other models utilizing LIDC-IDRI and Kaggle datasets. The experimental results for the LIDC-IDRI datasets, VGG-CapsNet, achieves a high Area Under the Curve (AUC) of 0.980 and an F1-Score of 98.61%. The precision, recall, specificity, and accuracy are reported as 99.07%, 98.16%, 99.07%, and 98.61%, respectively. For the Kaggle datasets, the precision, specificity, accuracy, AUC, and F1-Score are reported as 98.14%, 99.16%, 98.07%, 0.98, and 98.14%, respectively.

The IRCM-Caps [9] model combines the strengths of convolutional neural networks (CNN) and capsule networks (CapsNet) for the detection of COVID-19 using X-ray images. Additionally, it integrates an attention mechanism module alongside a multi-branch lightweight module to enhance the model performance. They utilize the Contrast Adaptive Histogram Equalization (CLAHE) algorithm

Table 1
Summary of results from related works.

Model	Dataset/Task	Accuracy (%)
DenseCapsNet [5]	COVID-19 detection in chest radiographs	90.7
DenseCaps [6]	MNIST, Fashion-MNIST, CIFAR-10, SVHN	99.70, 94.93, 89.41, 95.99
ResCaps [7]	Papillary thyroid cancer classification	81.06
VGG-CapsNet [8]	Lung cancer classification	98.61
IRCM-Caps [9]	COVID-19 detection in X-ray images	99.00, 93.00
S-VCNet [10]	Lumbar spondylolisthesis diagnosis	98.00
Ensemble Model [11]	Skin cancer detection	93.50
RNNinCaps [12]	3D vertebral image recognition	46.2 better than original CapsNet.
XCapsNet [13]	Diabetic retinopathy diagnosis	83.06, 98.33
FixCaps [14]	Skin cancer diagnosis	96.49
BoostCaps [15]	Brain tumor classification	92.45
Res2Net+Caps [23]	BreakHis dataset	95.6

to preprocess the images and enhance image contrast. The model is evaluated using a test dataset consisting of 1200 X-ray images, including 400 COVID-19 cases, 400 viral pneumonia cases, and 400 normal cases. The experimental results demonstrate that the accuracy of the IRCM-Caps model is 0.99% while the CapsNet accuracy is 0.93%.

S-VCNet [10], a hybrid VGG and CapsNet, improves the accuracy of diagnosing lumbar spondylolisthesis identification in X-ray images. In this study, a total of 466 private radiographs were used. Among them, 186 images were of a spine with spondylolisthesis, while 280 images depicted a normal spine. The proposed model was evaluated and compared to other approaches to diagnose lumbar spondylolisthesis. The results demonstrate that our model overcoming in the performance of other models and achieves an accuracy of 98%. Ensemble Model [11], a skin cancer detection method, combines VGG, CapsNet, and ResNet to enhance detection accuracy. The study utilized the ISIC (International Skin Imaging Collaboration) dataset, comprising 25,000 skin sample images across various categories. However, the study focuses on binary classification, utilizing 3,000 malignant and 2,800 benign images. Experimental findings demonstrate that the combined model surpasses individual learners, achieving an accuracy of 93.5%. In comparison, the standalone accuracies of CapsNet, ResNet, and VGG models are 75%, 79%, and 69%, respectively.

RNNinCaps [12] integrates a modified CapsNet with an RNN module for recognizing 3D vertebral images, trained on a dataset comprising 4,000 such images alongside CIFAR-10. Its performance is compared with various models including CNN, CapsNet, Baseline-1, and Baseline-2, all trained on the same dataset, demonstrating superior accuracy over the other models.

XCapsNet [13] is a deep learning model combining Xception and Capsule networks to improve the accuracy and efficiency of diagnosing diabetic retinopathy (DR) from fundus images. On the APTOS2019 dataset, the method achieves a classification accuracy of 83.06% for multiclass classification and 98.91% for binary-class classification of DR images, while on the Messidor dataset, it achieves an accuracy of 98.33% for classifying fundus images into DR and Normal classes, overcoming existing methods in terms of both accuracy and efficiency.

FixCaps [14] is another improved capsule network for classification of dermoscopic images in skin cancer diagnosis. It improves detection accuracy and reduces computational complexity by using a high performance large kernel convolution layer with a kernel size of [31 x 31] instead of the commonly used [9 x 9] to incorporate a larger receptive field compared to traditional capsule networks. Experimental results show that FixCaps demonstrates superior performance compared to existing methods, including IRv2-SA, a leading model for dermatoscopic image classification. On the HAM10000 dataset, FixCaps achieves an accuracy of 96.49%. This exceeds the performance of IRv2-SA.

BoostCaps [15] is the first capsule network model to incorporate a boosting approach and takes uses both images of the brain and rough boundary boxes of the tumor as input, allowing access to both the primary target and the surrounding tissue. It incorporates an internal boosting mechanism to gradually boost weak learners and eliminate the need for an exhaustive architecture search. The performance of BoostCaps classification using the Jun Cheng brain dataset achieved an accuracy of 92.45%, outperforming the original capsule model's accuracy of 89.83%.

Khikani et al. [23] introduced an advanced capsule network for breast cancer classification using histopathologic images. The model integrates a Res2Net block and four additional convolutional layers for multi-scale feature extraction and parameter reduction. The model, trained and tested to evaluate it on the BreakHis dataset, achieved a promising accuracy of 95.6% and recall of 97.2%, outperforming previous methods. Notably, the research used augmentation techniques such as rotations and flips to increase the dataset size so as to improve the performance, suggesting a reliance on augmentation for optimal results, particularly on smaller datasets.

The above-mentioned studies demonstrate different approaches to improving capsule network performance by using various network architectures, combining pretrained models, utilizing attention mechanisms, and incorporating boosting techniques. Table 1 summarizes the results of this work.

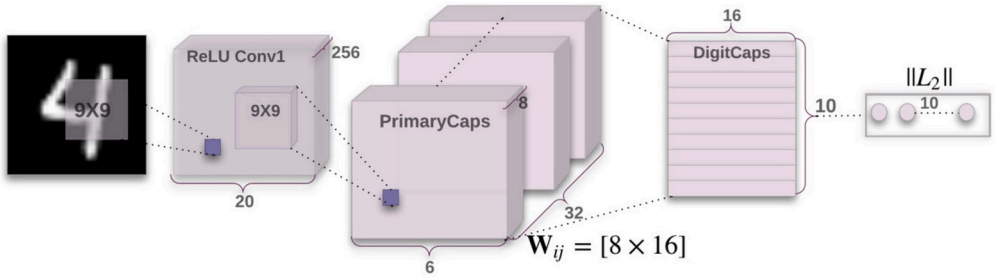


Fig. 2. Original Capsule Network Structure [16].

3. Methods and materials

3.1. Capsule network

The original Capsule network overcomes inherent shortcomings in CNNs, providing a new approach to object representation and recognition. Capsule network focuses on the effective representation of objects or object parts through the utilization of activity vectors. These vectors leverage the length to signify the existence of an entity and the orientation to encapsulate the construction parameters of objects [16]. A pivotal aspect of capsule networks involves ensuring that the output vectors fall from 0 to 1. This is achieved by implementing a squashing function, as shown in Equation (1).

$$\mathbf{v}_j = \frac{||\mathbf{s}_j||^2}{1 + ||\mathbf{s}_j||^2} \frac{\mathbf{s}_j}{||\mathbf{s}_j||} \tag{1}$$

Where the v_j represents the output vector of capsule j, and s_j denotes its total input. For all capsules, apart from the first one, s_j prediction value, as shown in Equation (2), is the weighted sum of the prediction vector $\hat{u}_j|_i$ from the preceding capsule. Furthermore, the vector of prediction $\hat{u}_j|_i$ is the product of the output of u_i from the lower-level capsule and the matrix of weight $\mathbf{W}ij$.

$$\mathbf{s}_j = \sum_i c_{ij} \hat{\mathbf{u}}_{j|i}, \quad \hat{\mathbf{u}}_{j|i} = \mathbf{W}_{ij} \mathbf{u}_i \tag{2}$$

Described model, the coupling coefficient, denoted as c_{ij}, plays a crucial role and is established through an iterative dynamic routing procedure. Specifically, these coupling coefficients, governing the connection strength between a lower-level capsule i and all subsequent higher capsules, are constrained by the SoftMax function to collectively sum to 1. The initial probability link between capsule i and capsule j, represented as b_{ij}, is a key factor in determining these coupling coefficients and is an integral part of the model's equation (Equation (3)).

$$c_{ij} = \frac{\exp\left(b_{ij}\right)}{\sum_k \exp\left(b_{ik}\right)} \tag{3}$$

The initial probability, which evolves in tandem with the weights during the learning phase, is contingent on the relative position and characteristics of the two capsules, rather than being solely influenced by the current input image. Subsequently, the initial coupling coefficient undergoes adjustments that mirror the alignment between the current output of the higher-level capsule $\mathbf{v}_j$ and the prediction emanating from capsule i, represented as $\hat{\mathbf{u}}_{j|i}$.

Capsules engage in communication through a dynamic routing algorithm, facilitating a collective consensus on the presence of distinct features and establishing hierarchical relationships among object parts. The previously elucidated consistency is quantified by a scalar value denoted as a_{ij}, computed as the dot product of vectors v_j and $\hat{u}_j$ normalized by |i. This calculated value is then added to b_{ij} to ascertain the updated coupling coefficient.

The architectural configuration of the original capsule network, as illustrated in Fig. 2, is structured as follows: The initial layer involves a Conv2D operation utilizing 256 filters with a $[9 \times 9]$ kernel size and a 1-dimensional stride. The activation function employed is Rectified Linear Unit (ReLU), resulting in feature maps of dimensions $[20 \times 20 \times 256]$. Subsequently, a convolutional layer is followed by the primary capsule layer, incorporating 256 filters with $[9 \times 9]$ convolutional kernels and a stride of 2. This produces 32 channels of convolutional output with 8-dimensional capsules. The outcomes of the primary capsule layer manifest as $[6 \times 6 \times 32]$ outputs, with each of the 32 units containing an 8-dimensional output capsule.

The third layer contains the digit capsule layer, fully connected to all capsules in the layer beneath. This layer contains 10 capsules, each dedicated to a specific digit. Every capsule within this layer features a 16-dimensional vector, accepting input from lower level layers of capsules and performing image classification. The concluding layer computes the magnitude of each capsule, indicating the likelihood of the entity's presence and thereby representing the probability of the classification outcome.

Extending beyond the classification layers, three fully connected reconstructing layers, referred to as the reconstruction loss, are added. These layers constitute a decoder network aimed at reconstructing inputs from the 16-dimensional capsule outputs. The first two layers contain 512 and 1024 units, respectively, employing the ReLU activation function, while the last layer contains 784 units,

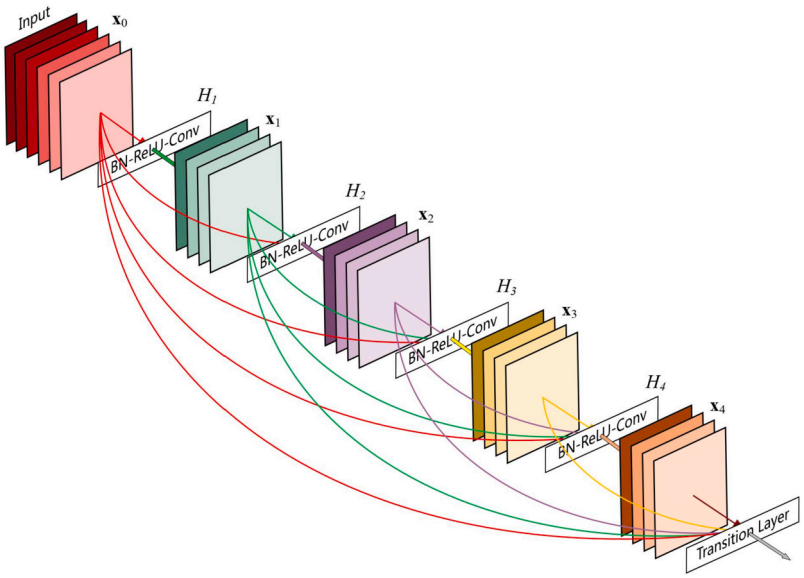


Fig. 3. The 5-layer depicts a dense block that has a growth rate of 4 (k = 4). In this block, every layer receives input from all the feature-maps generated by the preceding layers. [17].

consistent with the [28 × 28] input dimensions, and employs the sigmoid activation function to generate reconstructed images. It is important to highlight that a custom loss function is employed during training to ensure that the length of the instantiation vector of the correct capsule approximates 1, while the lengths of other vectors approach 0, given the binary nature of the inputs.

3.2. DenseNet201 model

DenseNet [17] optimizes information flow between layers by feed forwarding each layer to every other layer. Unlike traditional architectures, it utilizes dense connectivity patterns and feature reuse, resulting in improved gradient flow, reduced vanishing-gradient problem, and parameter efficiency. Each layer is connected to all other layers with matching feature-map sizes. The layer receives inputs from all preceding layers and passes its feature-maps to subsequent layers. The features are concatenated instead of summed, enabling the network to have L(L+1)/2 connections in an L-layer network. DenseNets uses a growth rate greater than 1 to enable dense connections between layers, giving each layer direct access to a rich set of feature maps from preceding layers. Fig. 3 displays the connectivity pattern of the 5-layer dense block with a growth rate of 4 (k = 4). Each layer in the block receives input from all the feature-maps generated by the preceding layers. The subsequent layers (H2, H3, H4) take all preceding feature maps (x0, x1, x2) as inputs and produce their own set of feature maps.

3.3. Proposed hybrid capsule network

The original CapsNet was initially developed and tested on the MNIST dataset, primarily composed of binary handwritten images [16]. This dataset has unique characteristics; it consists of binary pixels represented by values of 0 and 1. The dissimilarity between this dataset and medical images is not limited to their pixel representation but extends to the rich features and higher image quality inherent in medical images. In addition, unlike MNIST, medical datasets are often characterized by variations in the number and size of images, presenting a more complicated and diverse dataset composition [18]. The original capsule network was shown to be incompatible with medical datasets due to differences between these datasets and MNIST, particularly in terms of characteristics, number, size, and feature complexity. In particular, the increased complexity of pixel information in medical datasets with a relatively small number of images required a design adjustment of the capsule network to adapt to the unique characteristics of medical images. Although the original DenseNet201 model performs well in feature extraction, integrating it with the Capsule Network without modifications did not achieve the expected success and relied on data augmentation. Therefore, to overcome these challenges, we decided to enhance the feature extraction capabilities by combining modified DenseNet201 with the Capsule Network. This complementary approach aims to leverage the strengths of both architectures: the effectiveness of modified DenseNet201 in capturing hierarchical features from image data and the unique ability of capsule networks to handle spatial hierarchies. As seen in Fig. 4, this new hybrid architecture yields high accuracy without requiring any data augmentation techniques or a large dataset. The modifications are as follows:

1. Dense Block: The dense_block remains functionally the same in both the original DenseNet201 and the modified DenseNet model, except for differences in the number of hyperparameters and calling the function as described in the last point. The architectural

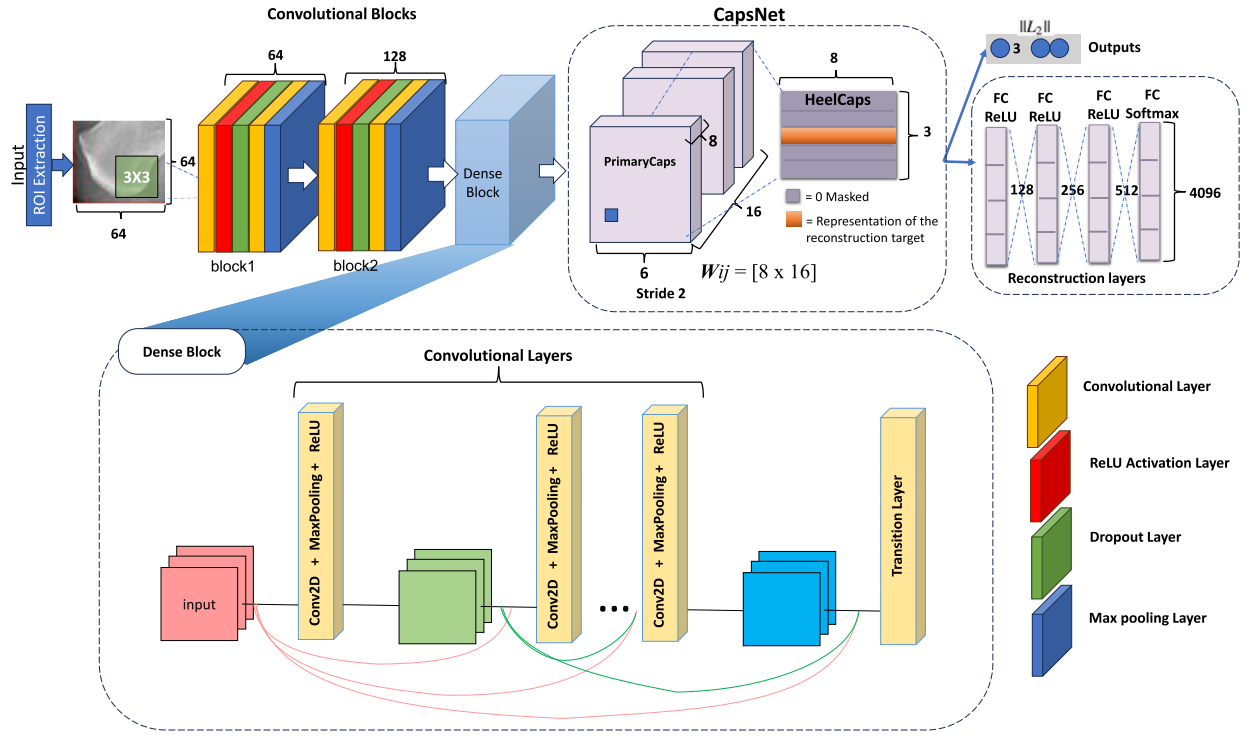


Fig. 4. Proposed hybrid capsule network architecture.

design includes a dense block composed of multiple densely connected blocks. Each layer within the dense block establishes feed-forward connections with all other layers, resulting in a dense connectivity pattern. This connectivity enhances the efficiency of information flow and facilitates gradient propagation throughout the network, enhancing the model's ability to capture complex patterns and features.

2. Dense Layer: In the original DenseNet201 architecture, the dense_layer function includes batch normalization, ReLU activation, and a single convolutional layer. However, in the modified architecture, the dense_layer consists of two Conv2D layers with different filter sizes to capture diverse and complex image features. A dropout layer is added to improve performance and avoid overfitting. The "he_normal" kernel initializer is used to effectively handle images with complex pixel patterns. This initializer addresses the limitation of using a standard normal distribution for weight initialization in conjunction with the ReLU activation function. The "he_normal" initializer draws samples from a truncated normal distribution centered at 0 with a standard deviation of $\sigma = \sqrt{2/n}$, where n denotes number of input units in the weight tensor [19]. By incorporating these adjustments, the modified DenseNet201 architecture aims to improve the model's ability to capture complex image features while addressing overfitting concerns.
3. Transition Layer: In the original DenseNet201 architecture, the transition_layer function is used to reduce the number of filters, which is achieved through a series of exciting operations including convolution, batch normalization, and activation followed by global average pooling for feature map pooling followed by global average pooling for subsequent pooling of the feature maps. On the other hand, in the modified DenseNet201 architecture, BatchNormalization is replaced with dropout, and GlobalAveragePooling2D is substituted with MaxPooling2D, enhancing the model's performance and adaptability to the task at hand.
4. Two convolutional blocks were added before feeding the input data into the DenseNet architecture. Each block is made up of two convolution layers: Conv1 and Conv2. Conv1 uses $[3 \times 3]$ kernel size, "same" padding, and "normal" kernel initializer, producing an output filtered with 64 channels for block1 and 128 channels for block2. After Conv1, a dropout layer is applied with a rate of 0.2 for block1 and 0.3 for block2. Next, Conv2 is applied with the same settings as Conv1, resulting in an output of 128 channels. Lastly, a MaxPooling2D layer with a kernel size of $[2 \times 2]$ is used to complete the blocks. These preliminary blocks serve as an initializer for the input to the DenseNet network, making it easier to receive normalized inputs with enriched feature maps instead of raw inputs.
5. As a final step, the original DenseNet201 architecture contains four blocks, each consisting of a different number of filters (6, 12, 32, 48) and a growth rate of 32, with a calling transition layer in each block iteration. However, the modified model contains only one block with two filters and a growth rate of 4. After the dense_block layer, a transition layer was introduced as the last layer of the modified DenseNet and input to the primary capsule.

As a result, the modifications made to the DenseNet201 architecture reduced the number of used hyperparameters and simplified the model. Consequently, the modified architecture demonstrates improved training efficiency, particularly when handling smaller

Table 2
Classes in the dataset.

Classes	Number of Image
Normal	1,842
Heel spur	1,316
Sever	798

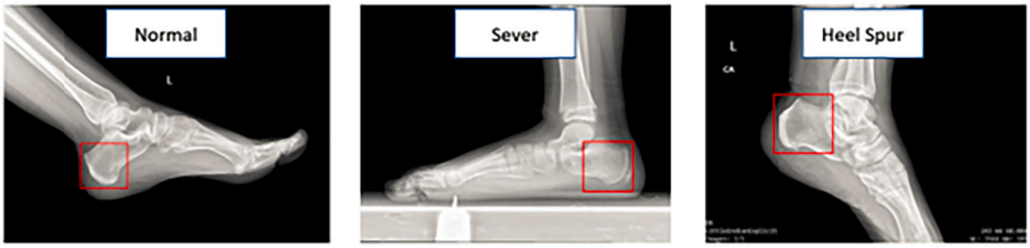


Fig. 5. Bounding-boxes for three different samples.

datasets and complex images with features, eliminating the need for data augmentation. These modifications improve performance while reducing the potential for overfitting. By reducing computational requirements and optimizing the model's ability to capture complex image patterns, the modified DenseNet201 architecture might provide a practical and effective solution for various image recognition tasks.

4. Experiments and results

We conducted a series of experiments on the proposed hybrid architecture with several datasets, including lateral X-ray foot image dataset (Heel dataset) [30] Breast_BreaKHis_v1, HAM10000 skin cancer dataset, and Jun Cheng Brain MRI dataset. Additionally, we used 100 epochs in each experiment to monitor the training and loss values in each iteration. All experiments were conducted using Google Colab environments, which included Python v3 as the runtime type, high RAM, and T4 GPU as the acceleration hardware. The performance results show that the model can achieve high accuracy with small and large medical image datasets. We explain each experiment according to the dataset used as follows.

4.1. MedcapsNet on heel dataset

The X-ray images dataset [30] contains three distinct classes: the "normal" class contained 1,842 images, the "heel spur" class contained 1,316 images, and the "severe" class contained 798 images, as given in Table 2. Thus, the dataset contained a total of 3,956 images. To facilitate disease identification in the heel bone images and to optimize network efficiency by excluding irrelevant areas, a bounding box method was used to create the region of interest (ROI).

The labelImg v1.8.6 software is used for bounding-box and annotation of the ROI. Fig. 5 shows samples of annotated X-ray images. Additionally, we save all the information of ROI in the comma-separated values (CSV) file. The CSV file contains the information of bounding-box, image name, xmin, ymin, xmax, ymax and label.

Before integration into the experimental setup, all images underwent preprocessing steps, including cropping to the ROI and subsequent rescaling to dimensions of $[64 \times 64]$. Given the imbalanced distribution of images across dataset classes, data balancing techniques, specifically oversampling, were applied to achieve a more balanced representation. We increased the number of instances in minority classes to match the number in majority classes to balance the dataset. In this study, the oversampling process was done by using the RandomOverSampler algorithm, which randomly duplicates samples in the minority class, from imblearn.over_sampling library. Consequently, the initial number of images was increased from 3,956 to 5,526 for balancing.

We have implemented original CapsNet [16] Res2Net+Caps [23], FixCapsNet [14], and BoostCaps [15] models according to the descriptions in their respective papers. To ensure the correct implementation of the existing models, we first trained the models using the datasets that employed in their studies. After achieving results consistent with those reported in the papers, we proceeded to train the models with heel dataset.

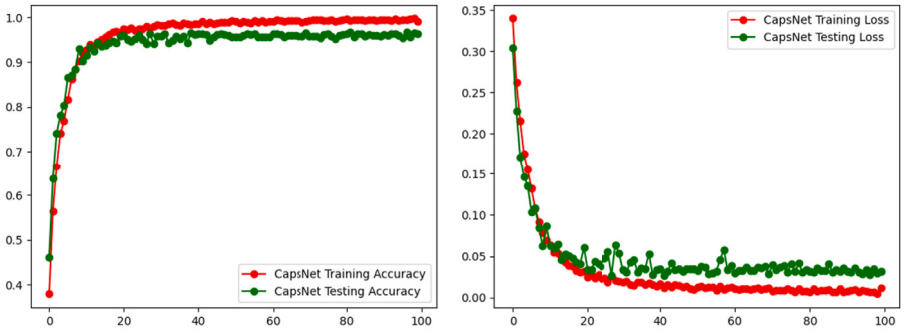
Table 3 shows the model accuracy values of original CapsNet, and the other capsule networks compared with MedCapsNet on the heel dataset. In our implementation, while original CapsNet showed an accuracy of 73.99%, Res2Net+Caps showed an accuracy of 34.81%, and FixCapsNet showed an accuracy of 73.33% on heel dataset. Regarding BoostCaps in [15], we encountered difficulties in method implementation due to some of the architecture components and parameters not being clearly specified.

MedCapsNet demonstrated 96.38% accuracy, as shown in Figs. 6a (Left) and (Right). Furthermore, the AUC score is 98.27%, as demonstrated in Fig. 6b (Right), along with the corresponding confusion matrix in Fig. 6b (Left).

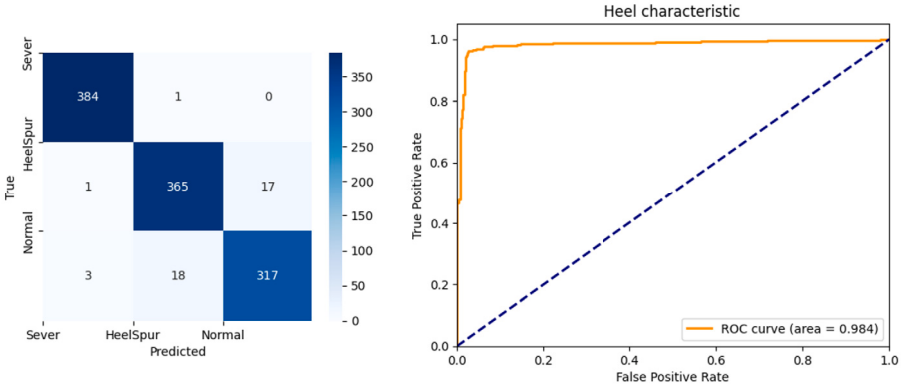
Furthermore, we implemented 5-fold cross-validation method on MedCapsNet. The full heel dataset of 5526 images has been splitted into 70% for training and 30% for testing. For training the model we used the part of 70%, while for the evaluation of model

Table 3
Results in comparison with base and recent capsule network models for Heel dataset.

Architecture	Accuracy
Original CapsNet [16]	73.99%
Res2Net+Caps [23]	34.81%
FixCaps [14]	73.33%
BoostCaps [15]	-
MedCapsNet	**96.38%**



(a) (Left) Training and Testing accuracy, (Right) Training and Testing loss.



(b) (Left) Confusion Matrix, (Right) AUC curve.

Fig. 6. Accuracy, loss, confusion matrix, and auc curve results for Heel dataset.

performance we used the part of 30% which splitted for the test and never exposed to the model training, also not the part that k-fold that splitting it for internal evaluation in iterations steps. The result shows the model performance evaluation done by cross validation accuracy 95.69% as shown in Figs. 7a (Left) and (Right) and ROC AUC 98.87% as demonstrated in Fig. 7a (Right), along with the corresponding confusion matrix in Fig. 7a (Left).

4.2. MedCapsNet on various medical datasets

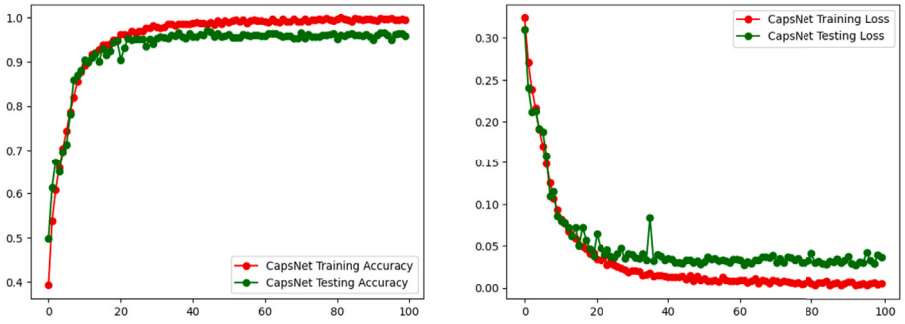
We evaluated the performance of MedCapsNet for various medical datasets to demonstrate its superiority compared to existing studies. For this, we tested the model on the datasets used by the studies in the related work section and compared the results. The evaluation results of their performances are given in the Sections 4.2.1, 4.2.2, and 4.2.3. In these experiments, the 5-fold cross-validation method was implemented using the kf.split algorithm to evaluate the performance, and the full data was split into 80% for training and 20% for testing. Table 4 presents the cross-validation accuracy results of MedCapsNet across various datasets, comparing its performance with accuracy of the other models.

4.2.1. MedCapsNet with Breast_BreaKHis_v1 dataset

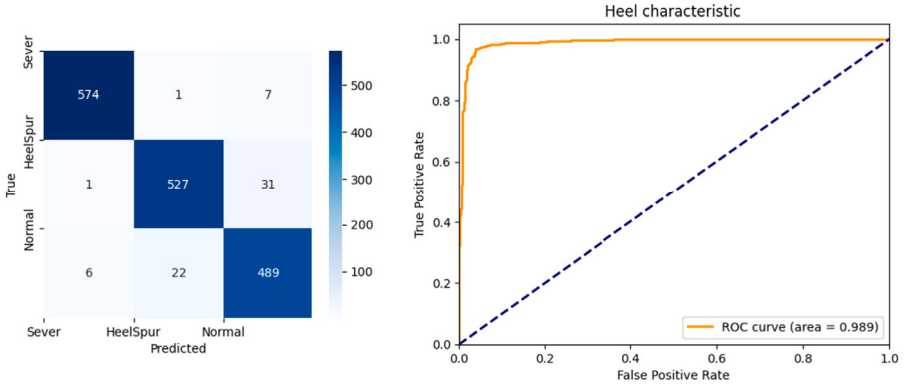
The BreakHis dataset, known as the Breast Cancer Histopathological Image Classification, is a collection of 9,109 microscopic images of breast tumor tissue. These images were obtained from 82 patients and were captured at various magnification scales, namely 40X, 100X, 200X, and 400X. The dataset consists of 2,480 benign samples and 5,429 malignant samples. Each image has

Table 4
Results in comparison for various medical datasets with MedCapsNet vs. other models.

Dataset	MedcapsNet (5-fold CV Accuracy)	Other Models (Accuracy)
BreakHis_v1 [20]	98.40% (without augmentation)	Res2Net+Caps [23]: 95.6% (with augmentation)
HAM10000 [21]	98.29%	FixCaps [14]: 96.49%
Jun Cheng [22]	97.67%	BoostCaps [15]: 92.45%



(a) (Left) Training and Testing accuracy, (Right) Training and Testing loss.



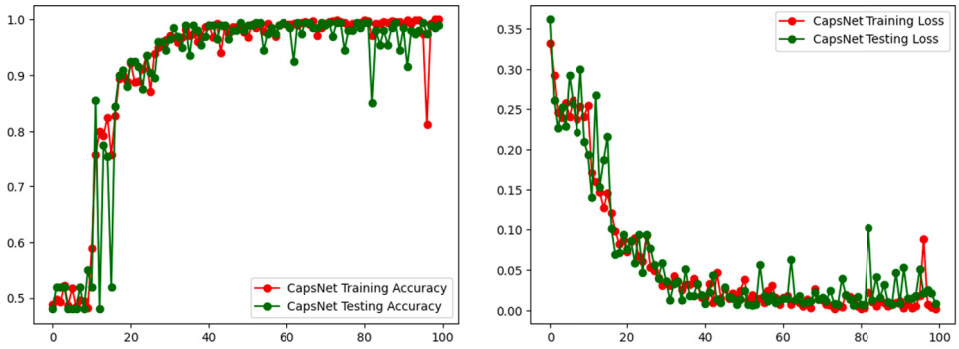
(b) (Left) Confusion Matrix, (Right) AUC curve.

Fig. 7. Cross-validation accuracy, loss, confusion matrix, and auc curve results for Heel dataset.

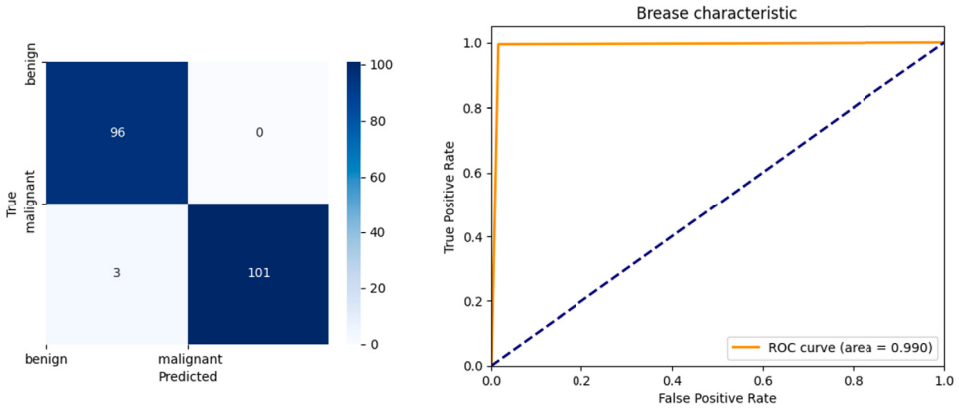
dimensions of 700 x 460 pixels and is represented in RGB color space with 3-channel information. The pixel depth in each channel is 8 bits and the images are stored in PNG format [20]. This experiment used a small subset of images from a breast cancer dataset, consisting of 1000 images rescaled to $[65 \times 65]$ pixels and converted to 1 color channel in grayscale mode. 800 images were used for training, while the remaining 200 images were used to test the performance with a limited sample size. The experiment yielded a 5-fold cross-validation accuracy of 98.40% and an AUC of 98.96% as demonstrated in Figs. 8a (Left) and 8b (Right), along with the corresponding confusion matrix in Fig. 8b (Left). Our proposed model demonstrates superior performance with a small sample of the dataset without using data augmentation compared to the model presented in [23], which achieved an accuracy of 95.6% using data augmentation.

4.2.2. MedCapsNet with HAM10000 dataset

The HAM10000 dataset, provided by the Harvard Dataverse Organization, addresses the limited size and diversity of existing dermatoscopic image datasets for training neural networks in the diagnosis of pigmented skin lesions. It consists of 10,015 dermatoscopic images collected from different populations and acquisition modalities. The dataset covers important diagnostic categories such as melanoma, basal cell carcinoma, and benign keratosis-like lesions [21]. The dataset is provided in 3 color channels on which the proposed model was trained in RGB mode. In addition, since the provided dataset is imbalanced, the oversampling method has been used to balance the dataset and also make the training number of images in large numbers, including 37548 images. In this experiment, the oversampling process was conducted in accordance with the methodology outlined in Section 4.1.



(a) (Left) Training and Testing accuracy, (Right) Training and Testing loss for breast BreaKHis v1 dataset experiment.



(b) (Left) Confusion Matrix, (Right) AUC curve for breast BreaKHis v1 dataset experiment

Fig. 8. Results for BreakHis dataset.

The result shows the high performance of the model, achieving cross-validation accuracy for 5-fold of 98.29%, given in Fig. 9a, and AUC of 99.00% as demonstrated in Fig. 9b (Right), along with the corresponding confusion matrix in Fig. 9b (Left). Our proposed model shows superior performance compared to the model presented in [14], which achieved an accuracy of 96.49%.

4.2.3. MedCapsNet with Jun Cheng brain MRI dataset

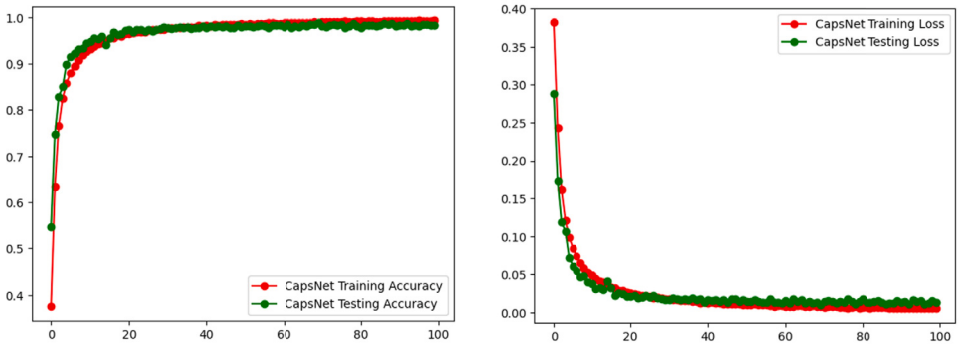
The brain tumor dataset has 3064 images representing three different categories of brain tumors: meningioma, glioma, and pituitary tumor published by Jun Cheng [22]. The provided dataset was imbalanced; therefore, oversampling was used to balance three classes. The total number of images after balancing is 4246. The proposed model was trained on the dataset with $[64 \times 64]$ image size in 3 channel RGb mode. The result shows the high performance of the model, achieving cross-validation accuracy for 5-fold of 97.67%, given in Fig. 10a, and AUC of 99.80% as demonstrated in Fig. 10b (Left), along with the corresponding confusion matrix in Fig. 10b (Right). Our proposed model achieved a higher accuracy than the boosted capsule network presented by [15], which reported an accuracy of 92.45%.

It's important to note that we couldn't compare our study to all cited studies in the "Related Work" section because they relied on proprietary datasets that are unavailable online. For instance, regarding the models of DenseCaps in [5] and DenseCapsNet in [6], we couldn't do comparison due to the model in [5] using a private Covid-19 dataset which is not accessible online. While the model in [6] using Street View House Numbers (SVHN) images which is not a medical images dataset and it may give good results with medical images or may not.

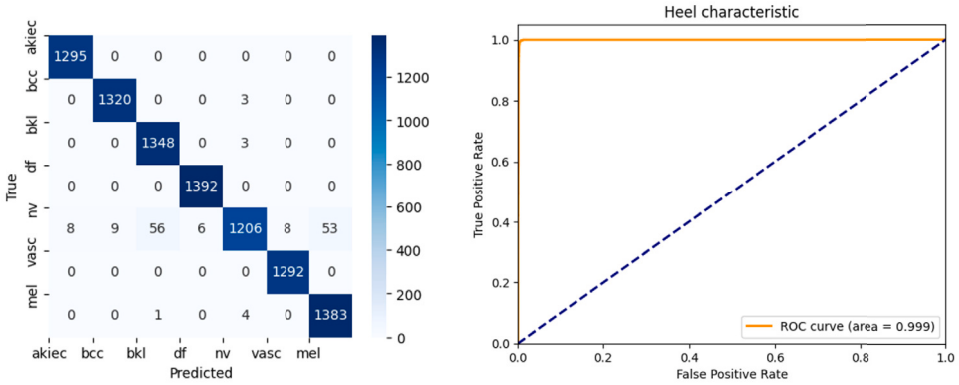
5. Discussion

Table 3 in Section 4.2 provides a comprehensive comparison of the performance of the proposed model with that of other models. From the results, we can infer that the existing models do not perform well on heel datasets with their current architectures and parameters. In contrast, our proposed model achieved a higher performance than the other models, without needing a different architecture and hyperparameters for each type of image and disease.

Our results suggest that the original CapsNets may have weak performance on medical images for two reasons. First, the original CapsNets have limited convolutional layers, which may need to be revised for effective feature extraction from complex images.



(a) (Left)Training and Testing accuracy, (Right) Training and Testing loss for HAM10000 skin lesion dataset experiment.



(b) (Left) Confusion Matrix, (Right) AUC curve for HAM10000 skin lesion dataset experiment.

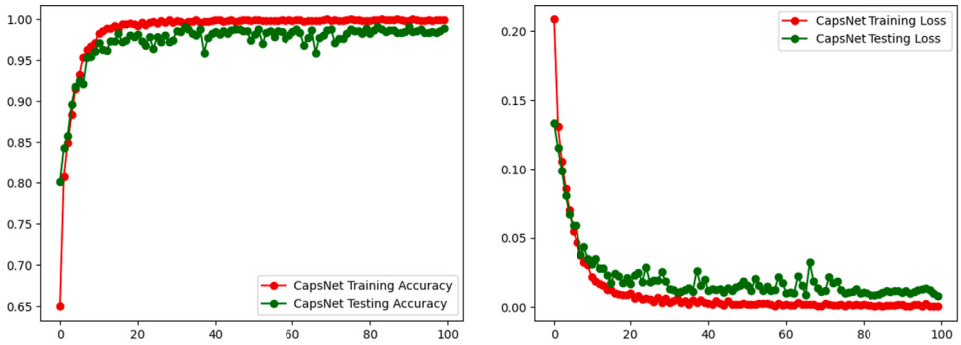
Fig. 9. Results for HAM10000 dataset.

Therefore, increasing the depth of the convolutional layer can significantly improve the classification accuracy. Second, the standard kernel initializer used for the ReLU activation function may not perform optimally for images with complex pixels. To address this, we utilized the kernel initializer of "he_normal", which is help to initializes the weights by drawing samples from a truncated normal distribution centered at 0 with a standard deviation of $\sigma = \sqrt{2/n}$, n is representing the number of input units in the weight tensor [19]. This technique addresses the vanishing/exploding gradient problem by utilizing the he_normal weight initialization in a modified DenseNet. The proposed initialization strategy enhances gradient flow, thereby aiding the convergence and training of deep rectified models, resulting in improved performance on complex medical images by mitigating this problem.

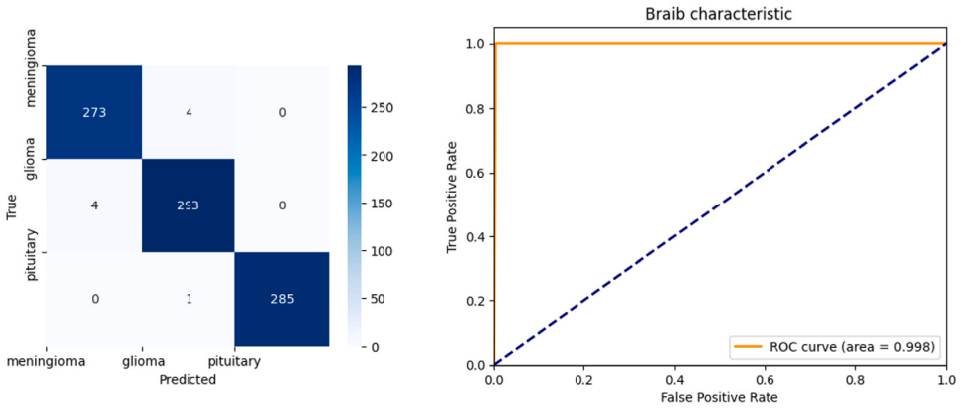
Placing a modified DenseNet201 before the capsules provides several benefits, including the following:

- First, it allows more robust and discriminative features to be extracted from the input data. This improvement in feature extraction contributes to improved accuracy by providing more relevant information for subsequent processing.
- Second, the architectural changes in the dense network facilitate the establishment of a structured representation of features. This allows the network to capture hierarchical relationships within the input data, effectively encapsulating spatial relationships and compositionality of objects. As a result, the network achieves better accuracy by leveraging these relationships after passing them to the capsule network.
- Third, the integration of the modified DenseNet201 adds increased non-linearity to the network. This increased non-linearity allows the network to detect and represent more complex patterns from the input images. By capturing these complex patterns, the network is better equipped to handle the complexity present in medical images.
- Finally, the modified architecture serves as a form of regularization. Techniques such as dropout and max-pooling2D reduce overfitting and facilitate better generalization to unseen data. Regularization helps prevent the network from relying too heavily on specific features or patterns during training, thereby improving its ability to learn and generalize from complex patterns in the data.

Both dropout and batch normalization are regularization techniques whose effectiveness can depend on the specific task, the architecture of the model, and other hyperparameters [24]. Replacing batch normalization with dropout improved the performance of our proposed model. The experiments conducted here could be supported by [15] which have also reported improvements when employing dropout. Replacing GlobalAveragePooling2D with MaxPooling2D also improved the performance of the proposed model



(a) (Left) Training and Testing accuracy, (Right) Training and Testing loss for Jun Cheng Brain MRI dataset experiment.



(b) (Left) Confusion Matrix, (Right) AUC curve for Jun Cheng Brain MRI dataset experiment.

Fig. 10. Results for Jun Cheng Brain MRI dataset.

[25]. MaxPooling2D was effective in our case because it involves taking the maximum value from a group of values, typically a $[2 \times 2]$ window, which compresses multiple feature maps into one, helps to capture the essential feature in a region, and reduces the possibility of overfitting, making it robust to small translations and distortions in the input [26–28]. Also, the use of max-pooling can be helpful in early layers to capture distinctive local features [29], as the dense network will be the early stage before the capsule in our proposed model. A study in [25] compared several methods that generalize max- and average-pooling and found that none significantly outperformed standard max-pooling in a classification task.

6. Conclusion

AI-based detection and classification can improve the efficiency and accuracy of diagnosing conditions such as heel spurs and Sever's disease, thereby improving patients' quality of life. To help diagnose heel spurs and Sever's disease, we propose MedCapsNet, an improved capsule network by integrating a modified DenseNet201 and addressing the limitations of the original CapsNets, we improve the accuracy and demonstrate the effectiveness of our method on various medical image datasets. We evaluated our model's performance on various medical datasets to demonstrate its performance on various medical image datasets, including lateral foot X-ray images dataset (Heel dataset), Breast_BreaKHis_v1, HAM10000 skin cancer dataset, and Jun Cheng Brain MRI dataset. In the first experiment we evaluated MedCapsNet for heel disease detection and classification. The other experiments evaluate the model's performance on various medical datasets to demonstrate its improvements over existing studies. The findings indicated that MedCapsNet achieved a higher performance compared to the other models. This was achieved without the necessity for a distinct architectural configuration or hyperparameters for each type of image and disease, and the use of any data augmentation. We also implemented YOLO V8 for an automated ROI detection in foot X-ray images. These results could contribute to the field of automated medical image analysis. In addition, by reducing computational requirements and optimizing the model's ability to capture complex image patterns, MedCapsNet might provide a practical and effective solution for various image recognition tasks.

Ethics statement

The Medical Ethics Committee of the Kirkuk Provincial Health Department and Kirkuk General Hospital approved the acquisition of numerous X-ray images for research purposes with approval document number [28386]. No informed consent was required because this is a retrospective study, and there is no information about the participants.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

CRediT authorship contribution statement

Osamah Taher: Writing – original draft, Validation, Software, Methodology, Investigation, Data curation, Conceptualization. **Kasım Özacar:** Writing – review & editing, Validation, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgements

We would like to express our sincere thanks to Dr. Ümit Özgür GÜLER, Specialist in Orthopedics and Traumatology, for his efforts and time in marking the foot x-ray images.

References

[1] E. Agyekum, K. Ma, Heel pain: a systematic review, Chin. J. Traumatol. 18 (2015) 164–169.
[2] American Academy of Orthopaedic Surgeons (AAOS), Plantar Fasciitis and Bone Spurs, OrthoInfo, 2000 [Online].
[3] S. Toraman, T. Alakus, I. Turkoglu, Convolutional capsnet: a novel artificial neural network approach to detect COVID-19 disease from X-ray images using capsule networks, Chaos Solitons Fractals 140 (2020) 110122.
[4] E. Goceri, Analysis of capsule networks for image classification, in: International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing, 2021.
[5] H. Quan, X. Xu, T. Zheng, Z. Li, M. Zhao, X. Cui, DenseCapsNet: detection of COVID-19 from X-ray images using a capsule neural network, Comput. Biol. Med. 133 (2021) 104399.
[6] G. Sun, S. Ding, T. Sun, C. Zhang, W. Du, A novel dense capsule network based on dense capsule layers, Appl. Intell. 52 (2022) 3066–3076.
[7] X. Ai, J. Zhuang, Y. Wang, P. Wan, Y. Fu, ResCaps: an improved capsule network and its application in ultrasonic image classification of thyroid papillary carcinoma, Complex Intell. Syst. (2021) 1–9.
[8] A. Bushara, R. Kumar, S. Kumar, An ensemble method for the detection and classification of lung cancer using computed tomography images utilizing a capsule network with visual geometry group, Biomed. Signal Process. Control 85 (2023) 104930.
[9] S. Qiu, J. Ma, Z. Ma, IRCM-Caps: an X-ray image detection method for COVID-19, Clin. Respir. J. 17 (2023) 364–373.
[10] D. Saravagi, S. Agrawal, M. Saravagi, S.K. Jain, B. Sharma, A. Mehbodniya, S. Chowdhury, J.L. Webber, Predicting lumbar spondylolisthesis: a hybrid deep learning approach, Intell. Autom. Soft Comput. 37 (2) (2023) 2133–2151.
[11] A. Imran, A. Nasir, M. Bilal, G. Sun, A. Alzahrani, A. Almuhaimeed, Skin cancer detection using combined decision of deep learners, IEEE Access 10 (2022) 118198–118212.
[12] H. Wang, K. Shao, X. Huo, An improved CapsNet applied to recognition of 3D vertebral images, Appl. Intell. 50 (2020) 3276–3290.
[13] M. Gour, S. Jain, S. Kaushal, XCapsNet: a deep neural network for automated detection of diabetic retinopathy, Int. J. Imaging Syst. Technol. 33 (2023) 1014–1027.
[14] Z. Lan, S. Cai, X. He, X. Wen, Fixcaps: an improved capsules network for diagnosis of skin cancer, IEEE Access 10 (2022) 76261–76267.
[15] P. Afshar, K. Plataniotis, A. Mohammadi, BoostCaps: a boosted capsule network for brain tumor classification, in: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 2020, pp. 1075–1079.
[16] S. Sabour, N. Frosst, G. Hinton, Dynamic routing between capsules, Adv. Neural Inf. Process. Syst. 30 (2017).
[17] G. Huang, Z. Liu, L. Van Der Maaten, K. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.
[18] A. Jiménez-Sánchez, S. Albarqouni, D. Mateus, Capsule Networks against Medical Imaging Data Challenges, CVII-STENT/LABELS@MICCAI, 2018.
[19] A. Kumar, N. Upadhyay, P. Ghosal, T. Chowdhury, D. Das, A. Mukherjee, D. Nandi, CSNet: a new DeepNet framework for ischemic stroke lesion segmentation, Comput. Methods Programs Biomed. 193 (2020) 105524.
[20] F. Spanhol, L. Oliveira, C. Petitjean, L. Heutte, A dataset for breast cancer histopathological image classification, IEEE Trans. Biomed. Eng. 63 (2016) 1455–1462.
[21] P. Tschandl, The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions, Harvard Dataverse, 2018 https://doi.org/10.7910/DVN/DBW86T.
[22] J. Cheng, W. Yang, M. Huang, W. Huang, J. Jiang, Y. Zhou, R. Yang, J. Zhao, Y. Feng, Q. Feng, W. Chen, Retrieval of brain tumors by adaptive spatial pooling and Fisher vector representation, PLoS ONE 11 (2016) e0157112.
[23] H. Khikani, N. Elazab, A. Elgarayhi, M. Elmogy, M. Sallah, Breast cancer classification based on histopathological images using a deep learning capsule network, ArXiv Preprint, arXiv:2208.00594, 2022.

[24] C. Garbin, X. Zhu, O. Marques, Dropout vs. batch normalization: an empirical study of their impact to deep learning, Multimed. Tools Appl. 79 (2020) 12777–12815.
[25] F. Bieder, R. Sandkühler, P. Cattin, Comparison of methods generalizing max- and average-pooling, ArXiv Preprint, arXiv:2103.01746, 2021.
[26] B. Graham, Fractional max-pooling, ArXiv Preprint, arXiv:1412.6071, 2014.
[27] L. Cheng, D. Chang, J. Xie, R. Ma, C. Wu, Z. Ma, Channel max pooling for image classification, in: International Conference on Intelligent Science and Big Data Engineering, 2019, pp. 273–284.
[28] Y. Zheng, B. Iwana, S. Uchida, Discovering class-wise trends of max-pooling in subspace, in: 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2018, pp. 98–103.
[29] Y. Zheng, B. Iwana, S. Uchida, Mining the displacement of max-pooling for text recognition, Pattern Recognit. 93 (2019) 558–569.
[30] O. Taher, K. Özacar, HeCapsNet: an enhanced capsule network for automated heel disease diagnosis using lateral foot X-ray images, Int. J. Imaging Syst. Technol. 34 (3) (2024) e23084.