DOI: 10.24996/ijs.2026.67.2.39

ISSN: 0067-2904

Evaluating the Impact of the Caputo Derivative on Deep Models for Hypertrophic Cardiomyopathy Classification Using CMR Images

Bushra K. Hilal [1, 2] , Ebtesam N. AlShemmary*[3]

[1]*Department of Computer Science College of Computer Science and Mathematics, University of Kufa, Kufa, Iraq.*
[2]*Department of Computer Information Systems, College of Computer Science and Information Technology, University Al- Qadisiyah, Al-Qadisiyah, Iraq.*
[3]*IT Research and Development Center, University of Kufa, Kufa, Iraq.*

Received:11/ 12/2024 Accepted: 25/ 2/2025 Published: xx

Abstract

To enhance patient outcomes, it is essential to diagnose hypertrophic cardiomyopathy (HCM) from cardiac magnetic resonance (CMR) images with precision, ensuring the process is swift and automated. This study investigates the impact of integrating Caputo derivatives into deep learning models to enhance their performance in classifying HCM. The study examines the performance of a tailored convolutional neural network (CNN), the advanced EfficientNetV2S architecture, and the improved CNN incorporating the Caputo derivative. Key pre-processing techniques included image resizing, normalization, and data augmentation. Caputo's CNN performed best with 92.47% accuracy, 93.57% precision, and 89.36% F1 score with a slightly reduced recall of 85.68%, while EfficientNetV2S achieved the highest accuracy (98.62%), demonstrating exceptional feature extraction capabilities. The results suggest that fractional calculus combined with deep learning can deepen diagnostic accuracy in CMR while providing more effective and interpretable HCM classification frameworks.

Keywords: Cardiac magnetic resonance (CMR), Caputo derivative, Deep learning models, EfficientNetV2S, Hypertrophic cardiomyopathy (HCM).

تقييم تأثير مشتق كابوتو على النماذج العميقة لتصنيف اعتلال عضلة القلب الضخامي باستخدام صور الرنين المغناطيسي القلبي

بشرى كامل هلال [2،1] ، ابتسام نجم الشمري *[3]

[1]قسم علوم الحاسوب، كلية علوم الحاسوب والرياضيات، جامعة الكوفة ، الكوفة، العراق.
[2]قسم نظم المعلومات الحاسوبية، كلية علوم الحاسوب وتكنولوجيا المعلومات، جامعة القادسية ، الديوانية، العراق.
*[3] مركز البحث والتأهيل المعلوماتي، جامعة الكوفة، الكوفة، العراق.

* Email: dr.alshemmary@uokufa.edu.iq

الخلاصة

لتحسين نتائج المرضى، من الضروري تشخيص اعتلال عضلة القلب الضخامي (HCM) من صور الرنين المغناطيسي للقلب (CMR) بدقة، مما يضمن سرعة العملية وأتمتتها. تبحث هذه الدراسة في تأثير دمج مشتقات كابوتو في نماذج التعلم العميق لتحسين أدائها في تصنيف اعتلال عضلة القلب الضخامي. تدرس الدراسة أداء الشبكة العصبية التلافيفية المصممة خصيصًا (CNN)، وهندسة EfficientNetV2S المتقدمة، وشبكة CNN المحسنة التي تتضمن مشتق كابوتو. تضمنت تقنيات المعالجة المسبقة الرئيسية تغيير حجم الصورة والتطبيع وزيادة البيانات. حققت شبكة CNN الخاصة بكابوتو أفضل أداء بنسبة دقة 92.47٪ ودقة 93.57٪ ودرجة F1 89.36٪ مع استرجاع منخفض قليلاً بنسبة 85.68٪، بينما حققت EfficientNetV2S أعلى دقة (98.62٪)، مما يدل على قدرات استخراج ميزات استثنائية. تشير النتائج إلى أن حساب الكسرات مع التعلم العميق يمكن أن يعمق دقة التشخيص في تصوير القلب بالرنين المغناطيسي مع توفير أطر تصنيف اعتلال عضلة القلب الضخامي أكثر فعالية وقابلية للتفسير.

1. Introduction

People with hypertrophic cardiomyopathy oftentimes protest of heart murmurs, increased heart rate, shortness of breath, abnormal heart rhythms, chest pain, and fainting. Hypertrophic cardiomyopathy is a condition in which the left ventricle muscle becomes enlarged, which can lead to sudden cardiac death [1]. Hypertrophic cardiomyopathy has a spread from 1 in a thousand cases to 200 cases per 100,000 subjects who need immediate diagnosis for the start of appropriate treatments, cardiovascular risks management, and periodically appropriate medical. A traditional diagnostic method is echocardiography or CMR imaging. From CMR images using specific pathological features, deep learning systems such as CNN can identify HCM patients [2]. Medicine uses cardiac CMR scans to diagnose complex heart defects and effectively separate healthy from abnormal heart parts. However, compared to computed tomography (CT) or echocardiography, it takes more time to produce images. Most patients with hypertrophic cardiomyopathy never receive a diagnosis, which increases their chances of developing serious health problems worldwide. Medical progress has improved diagnosis; however, patients struggle to obtain the appropriate treatment methods they need. Modern artificial intelligence advancements use deep learning to make medical imaging analysis more effective. Using deep learning constructs such as convolutional neural networks, raw visual data is processed to extract and classify features, authorizing accurate detection of clinical problems. Particularly, the analysis of cardiac CMR finds heart conditions that echocardiography would overlook. Treatment of hypertrophic cardiomyopathy soon after detection helps protect patients against sudden cardiac death [3].

This study explores the potential of deep learning in identifying hypertrophic cardiomyopathy by analyzing the interpretation of CMR images. The study analyses how three distinct CNN-based methods perform their tasks.

1. Model 1: A new five-block CNN structure is developed, incorporating diverse convolutional layers and skip connections, a characteristic also present in model 3. It improves feature extraction and the way input information connects to output results. Then utilizes a shallow artificial neural network (ANN) as the binary classifier.
2. Model 2: This model combines two neural networks - an ANN and EfficientNetV2S - to classify binary images using transfer learning. New analysis provides superior results compared to past research.
3. Model 3: This model retains the same structure as Model 1. The model structure includes a Caputo derivative layer placed after the second block. This additional experimental test was developed to enhance the capabilities of convolutional neural networks. To improve the model's ability to recognize and analyze complex patterns in data, the approach involves extending the model by incorporating the Caputo derivative.

The evaluation metrics were instrumental in identifying the optimal model, which was subsequently analyzed compared to others. The major contributions of this study:

- Through feature extraction and categorization, the novel CNN architecture can be improved.
- EfficientNetV2S integration: By the usefulness of transfer learning, the categorization accuracy can be highly improved.
- CNN model based on Caputo derivative layer: It makes the proposed model more capable of understanding and recognizing complex patterns in data.
- The suggested models demonstrate improvement across the following performance criteria: F1 score, precision, AUC, recall, MCC, and accuracy.

Related works, techniques and materials, proposed models, assessment criteria, analysis and results, and finally, conclusions are the sections of this study.

2. Related Works

One possible method for diagnosing hypertrophic cardiomyopathy (HCM) is through a cardiac MRI scan. Utilizing newly developed deep convolutional neural networks (CNNs), an updated dataset achieved a classification accuracy of 98.53% [1].

This paper sophisticated two machine learning models, LSTM and CNN, to be suitable for electrocardiogram (EKG) and cardiac magnetic resonance (CMR) scans, respectively. These models are designed to classify examinations into HCM and non-HCM classes. The LSTM model achieved an accuracy of 90.51%, precision of 60.31%, recall of 60.08, and F1 score of 60.19. The CNN model achieved an accuracy of 94.71%, precision of 96.97%, recall of 91.21%, and F1 score of 94.85%. These results can be used to apply the two models in the treatment of hypertrophic cardiomyopathy [5].

To evaluate left ventricular function in healthy persons and patients with hypertrophic cardiomyopathy and dilated cardiomyopathy, (2022) Guo et al. utilized a CNN model to analyze cardiac magnetic resonance (CMR) images. The model attains the ejection fraction sensitivity of approximately 92.31% in diagnosing HCM. However, in cardiomegaly cases, this degraded accuracy towards delineating cardiac boundaries in DCM. This study demonstrates a possible contribution of artificial intelligence (AI) in cardiac analysis and argues that models could be more accurate in assessing pathological cases [6].

For classifying gastrointestinal diseases from the Kvasir high-quality endoscopic image dataset in (2024), Demirbaş et al. suggested a new architecture depending on Spatial Attention ConvMixer (SAC). They successfully implemented this architecture in SAM by integrating the spatial attention mechanism with ConvMixer layers. In the study, data augmentation techniques were used to balance the data distribution and add model generalization. Finally, the proposed SAC model not only achieved higher accuracy compared to ResNet50 (87.44%) and Vanilla Vision Transformer (79.52%) models but also significantly outperformed them, achieving 93.37%. This also confirms that it performs better than traditional methods in medical image classification [7].

Using wireless capsule endoscopy (WCE), an innovative study was presented by Kim et al. (2024) for the utilization of deep learning for anatomical landmark marker classification of the upper gastrointestinal tract. The authors then applied color transfer techniques to improve the images studied in the study and create datasets resembling real WCE images. Applying the DenseNet169 model to such images, the research was able to achieve a classification accuracy of over 90%. This is significant as the use of image enhancement philters such as "Sharpen”, and "Detail" raised accuracy to 94.06% from 91.32%. The relevance of improving image quality, specifically to complement WCE diagnostics, is manifested in this work [8].

To improve the classification of gastrointestinal diseases through deep learning, Mari et al. (2023) conducted a study on how their technique could enable this. The researchers used the VGG-19 and ResNet-50 models to analyze and classify the Kvasir, using its dataset (3,500 images in 7 categories, that is, 500 images of each category). The study then fine-tuned the models' weights through transfer learning over the ImageNet dataset with improved accuracy. From this analysis, the ResNet50 model was found to be better performant because of a recall of 95.28%, accuracy of 96.81%, and precision of 95. This is contradictory to that recall, accuracy, and precision for the VGG-19 model were 94%, 94.21%, and 94.28%, respectively. During this research, it appears that the utilization of CNN architectures, such as ResNet50 and VGG-19, can give significant aid to clinicians in the process of accurate, efficient medical image classification, thereby improving the accuracy of diagnosis [9].

3. Materials and Methods

A detailed explanation of the methodology applied is provided in this section of the paper . How the classification network is developed, accompanied by a comprehensive description of the database and its processing.

3.1 Dataset Description

Cardiac magnetic resonance (CMR) images of 59,267 in patients with cardiomyopathy were used for classification in this study sourced from Kaggle [10]. The dataset consists of 37,421 images from healthy subjects and 21,846 images from patients with hypertrophic cardiomyopathy (HCM). The images in this dataset was collected between 2018 and 2020 at Omid Hospital, Tehran, under ethical approval, and underwent meticulous labelling by three cardiac imaging experts to attain high quality annotations. The average subject population age was 48.2 years (standard deviation 19.5 years and 53% female participation represents a diverse, representative sample. The dataset and sample images are shown in Figure 1, which includes some healthy samples and HCM.

In order to avoid any possible limitations on the level of the dataset size and to increase generalization, several data augmentation techniques like image rotation, flipping, and scaling were used. These methods attempted to improve model robustness without introducing bias or overfitting risk by introducing variability. Due to the size, diversity, and quality of the dataset it serves well as input for training convolutional neural network (CNN) models for CMR image classification. By including balanced and well labelled data, it is coupled with the fact that models do generalize well across different clinical settings and possible dataset insufficient and biased issues.

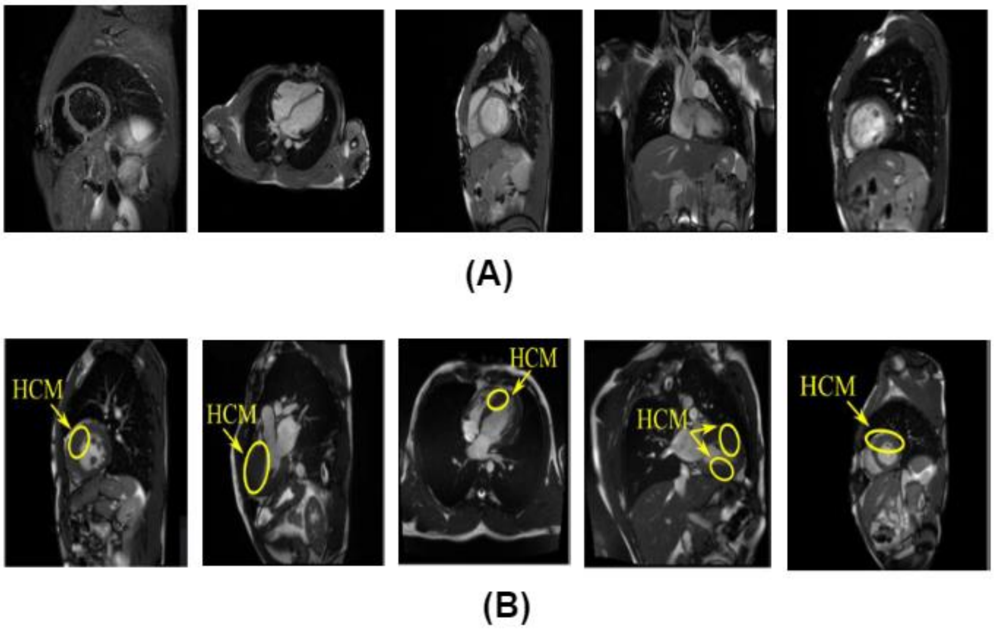


Figure 1: Sample of the dataset (CMR) images: (A) normal persons, (B) HCM patients

3.2 Convolution Neural Network

Consequently, deep learning algorithms, especially convolutional neural networks (CNNs), have become a keystone in medical image analysis. Raw image data is used by these models to learn hierarchical features for diagnosis, where diagnostic accuracy outperforms traditional methods. Normalizations and augmentation techniques further help increase the model performance of such model by reducing overfitting and providing better generalization [11, 12].

One such neural network (NN), also known as Convolution neural networks (CNNs), can accomplish semantic segmentation, object detection, and classification, among other things [13]. These networks are much used in deep learning (DL) applications such as facial recognition, voice recognition, and computer vision. A major contribution of CNNs is their recent capability, without any human supervision, to learn autonomously to identify important features, which provide advantages such as parameter sharing, sparse interactions, and equivalent representation [14]. Multiple convolutional layers, fully connected layers, and pooling layers are the common components of convolutional neural networks. They are very adaptable and train on very large datasets. However, they are readily available within the TensorFlow or the Keras frameworks, so they can be easily modified to serve any purpose. Figure 2 shows the CNN in action during picture. Categorization's input x for every layer in a CNN model is structured in three dimensions, each represented as ($m \times m \times r$), where the height (m) is the same as the width. The RGB image depth is represented by the three channels. Multiple kernels, or filters, denoted by k, are used by each convolutional layer, just as in the input image. The dimensions of these layers are also three-dimensional, measuring $n \times n \times q$. Where q could be less than or equal to r, but n can't be more than m. Additionally, the input is transformed into k feature maps h_k, where m is the minimum remoteness and n is the maximum dimension. The feature maps are created by combining the local connections made by the kernels with common parameters (bias b_k and weight w_k). In comparison to older, more traditional neural networks (NNs), (CNNs) have many advantages in computer vision (CV) tasks, such as the following [13, 14]:

1- The reduction number of trainable parameters due to the weight sharing function is the main advantage of using CNNs. As a result, generalization is improved, and overfitting is prevented.
2- When the feature extraction and classification layers are trained simultaneously, the model produces output that is both highly feature-dependent and well-structured.
3- It is simpler to deploy CNNs on a broad scale than other forms of neural networks.

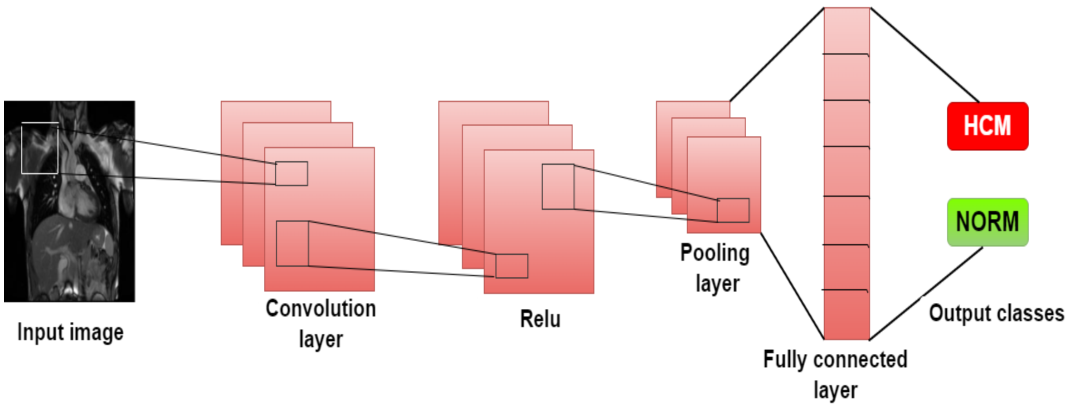


Figure 2: CNN architecture for image classification

3.2.1 Batch Normalization

By standardizing and normalizing the input obtained from the previous layer, batch normalization (BN) improves the stabilization and quickness of deep NNs. It is an unsupervised learning approach. It provides a "smart initialization" that enhances DN learning performance by adjusting its spline division to match the input without relying on the gradient-based learning or DN's weights does so. Training incredibly deep networks is where BN shines since it reduces the impact of internal covariate variations. Following convolutional or fully linked layers but before activation functions is batch normalization. The process of normalizing activations in intermediary layers aids in preventing overfitting and speeds up training. Learnable parameters known as gamma and beta are used to accomplish scaling and offsetting. A linear modification that is applied to the output of the previous layer during inference is the batch normalization; this layer is usually a convolution. Introducing randomness and noise to the layer inputs may also serve as a regularization for artificial neural networks. Normalization is the first step in BN's process, including rescaling and compensation.

Batch normalization has many benefits. It can minimize internal covariate shifts, which speeds up learning. It can regularize the model by adding a small amount of noise. It can enable greater learning rates and serve as a smart initialization for supervised learning tasks like classification and regression. Batch normalization (BN) increases margins by introducing a random jitter perturbation to the decision boundary, which helps the model learn boundaries with larger margins to the nearest training samples. This can improve batch normalization. For example, adjusting the noise strength and the standard deviation of batch normalization parameters can help further control the margin of the decision boundary. One of the main reasons for developing BN was to reduce internal covariate shift (ICS). This reduction is widely considered the key factor behind the success of batch normalization [15-17].

3.2.2 Skip layer

With the use of shortcut connections, sometimes called skip connections, lower levels of a network may instantly join higher ones. The fact that this allows data to travel beyond defined levels is what the word "skipping" alludes to. To address the domain shift effectively at both the image and instance levels, the skip layer modifies the multilayer domain in Domain Adaptive (DA) Faster R-CNN9. The domain classifier takes advantage of the convolutional layer's top-level feature maps despite their low resolution. Additionally, the high-resolution feature maps at lower levels are ignored. Because the domain classifiers do not take into account lower and higher feature maps, the adaptive model can't generalize as well. Multi-label classification techniques enable the effective analysis of complex medical images, such as chest X-rays, for cardiomegaly prediction. These deep learning models identify multiple abnormalities within a single image, improving prediction accuracy and enabling personalized treatment strategies based on the multifaceted nature of cardiovascular diseases [18]. Classifiers for domains that use just high-level feature maps will not be able to mitigate the distribution bias between the two domains. The Skip-Layer Network was developed to augment the data used by domain classifiers in light of these difficulties. One of the primary methods used by modern neural networks to address the issue of disappearing gradients is the usage of skip connections. In backpropagation, when a network employs skip connections to provide shorter pathways, the loss calculated at the output may have a greater impact on the layers that came before it. The benefits of skip connections are, therefore, as follows [19, 20]:

1. ResNet improves deep neural networks' accuracy by avoiding the training-related vanishing gradient problem.
2. To re-create pictures at a super-resolution, CNNs use skip connections with groups of layers.
3. Skip connections with CNNs were implemented to address the issue of fading gradients.
4. Skip connections allow CNNs to need fewer filters and weights, in contrast to architectures like VGG that do not use them.

3.2.3 Dropout

To avoid the problem of overfitting while training neural network models, dropout regularization is widely utilized in academic settings. Among the many methods that make use of it are feedforward, convolutional, recurrent neural network, and transformer-based approaches, as well as AutoDrop Dropout (Attention Drop), Max Dropout, and Drop Block. Further study is necessary to thoroughly investigate the potential of dropout methods, and the selection of the best one is dependent on the particular situation and architecture. In deep learning models, dropout has been successful in reducing overfitting and enhancing performance [21]. Important difficulties in learning multi-layer neural networks, particularly in deep learning, include overfitting and lengthy training durations. It is well-known that regularization may be used to address these problems. Although dropout improves deep learning in its special way, it may cause convergence time to rise. Working with CNNs on dropouts requires care. One way to prevent overfitting, which (CNN) and many other DL models use, is the dropout layer. This layer is similar to that, where during the training of the network it randomly turns off some neurons, thus activating the network to interiorize more robust features [22].

3.3 The Proportional Caputo Derivative

Specifically, the Caputo derivative, a fractional derivative, is suitable for modeling processes having memory and hereditary properties and, consequently, is well matched to the complexity found in CMR image data. In contrast to integer order derivatives that focus only on local changes, the Caputo derivative can include non-local correlations in the data, yielding improved feature extraction through the retention of the information from the spatially distributed patterns in CMR images. In comparison to standard optimization heuristics such as the gradient descent or L-BFGS, the Caputo derivative enhances the sensitivity to small perturbations in image features that are fundamental in discriminating HCM vs. non HCM. Fractional calculus based method Caputo derivative improves the performance of neural networks, more specifically in image classification tasks. Inside neural network activation functions, especially MLPs scientists use Caputo and other fractional derivative operations. Training stability and model generalization performances improve using fractional derivatives, according to research. The Caputo derivative, a type of fractional derivative, is defined as [23]:

$$C\,D^{\alpha}{}_{\alpha}f(t) = \frac{d^{\alpha} f(t)}{dt^{\alpha}} = \frac{1}{\Gamma(n-\alpha)}\int_{\alpha}^{t}\frac{f^{(n)}(x)}{(t-x)^{\alpha-n+1}}dx \qquad \text{.....(1)}$$

Where,

- n=⌈α⌉(the ceiling of α),
- Γ(·) is the Gamma function,
- α is the fractional order (0<α≤1).

The importance of Caputo's derivative with neural networks:

- Enhanced Generalization: Fractional activation functions, such as those derived from Caputo's approach, enhance the generalization capabilities of neural networks, thereby reducing the risk of overfitting.
- Adaptability to Complex Patterns: Caputo activation functions enable networks to handle complex data patterns, such as medical images, more effectively than traditional activation functions.
- Accelerated Convergence: The application of the Caputo derivative facilitates faster model convergence, thereby reducing the training time and enhancing the model's classification efficiency.

3.3.1 Modified Convolution with Caputo Derivative

The best model performance, particularly in a medical image classification, combines Caputo's derivative and Neural Networks. Let X (i, j) represent a pixel in the input feature map and W (k, l) the corresponding filter weight. The fractional convolution operation can be expressed as [23]:

$$Y(i,j) = \sum_{k}\sum_{l} W(k,l)\frac{d^{\alpha}X(i-k,j-l)}{d^{\alpha}} \qquad \text{.....(2)}$$

Where $\frac{d^{\alpha}X}{d^{\alpha}}$ is the Caputo derivative applied along spatial dimensions. This operation allows the filter to consider non-local spatial dependencies, effectively capturing nuanced relationships in the input data.

3.3.2 Advantages in Feature Extraction

The proposed improvement stems from the derivative's capacity to regulate the activation function's behavior through the fractional parameter. The main highlights are:

- Control over Model Behavior: A fractional parameter is incorporated into the activation function behavior through the Caputo derivative. The control makes the model more responsive to data and improves generalization.
- The Caputo derivatives are demonstrated to reduce overfitting, thereby enhancing model performance across a broad range of datasets. This enhances the model's generalization ability.
- Improved Stability: The derivatives are subsequently utilized to enhance training stability, accelerate convergence, and improve classification accuracy within the model.
- Computational Efficiency: Comparison to traditional activation functions showed that Caputo based derivative models are computationally faster when finding optimal solutions.

Thus, it is quite preferable to construct robust and efficient neural network models for medical image classification using a Caputo derivative [23].

3.3.3 Numerical Implementation

The finite differences approximation of the Caputo derivatives is used for computational efficiency. For an input sequence x[n] [23]:

$$\frac{d^{\alpha}x[n]}{dt^{\alpha}} \approx \sum_{m=0}^{n} \frac{(-1)^{m}\Gamma(\alpha+1)}{\Gamma(m+1)\Gamma(\alpha-m+1)} x[n-m] \qquad \text{.....(3)}$$

This discrete representation is incorporated into the backpropagation algorithm, ensuring compatibility with standard gradient-based optimization techniques.

3.4 Transfer learning

The convolution neural network CNN based models are performing better in image processing and classification tasks, and thus, they are not only responsible for revolutionizing medical imaging but also critical to the diagnosis and treatment of sickness. Transfer Learning (TL) presents an effective option to overcome difficulties in working with small datasets because developers can use pre-trained CNN models. TL is a way to teach a model to solve one problem and then apply that to another. Training a new model is better than starting from scratch when extracting features for it because it can use the existing model. Despite the CNN significant success, CNNs exhibit inductive biases, such as equivariance of the regional responsive field in translation, which limits their ability to learn long-range information [24].

In general, TL models are trained with a large dataset, such as ImageNet. It is possible to use the model's parameters to build a custom neural network, which has several potential applications. These models may be used directly to make predictions about new tasks. Batch normalization is a popular approach to addressing the overfitting issue, dropout, and transfer learning. A synopsis of TL's key arguments is as follows:

1. When learning new information, it is vital to build on one's present skills.
2. A more efficient and accurate learning process is possible, or fewer training datasets are required.

While many more transfer learning strategies will certainly be explored, the optimal approach to utilize such models in image categorization will depend on the size and similarity of the dataset. CNN development requires substantial computing power yet merging pre-trained models with transfer learning makes diagnosis stronger while lowering cost. When TL influences the use of additional training and testing samples, faster and more efficient results are obtained. As an improvement tool, TL may assist with the modeling performance of a secondary task. There are three distinct contexts for TL approaches due to variations in the domains of origin, destination, and tasks.

Regarding the contributions of transfer learning in the medical sciences, medical imaging and MRI play a crucial role in daily clinical diagnosis and treatment. MRI distinguishes between healthy and diseased tissue. With the knowledge of machine learning and transfer learning, medical researchers can more effectively identify diseases. However, extensive training data can be costly to use. Therefore, transfer learning leverages medical imaging, making CNNs highly effective in analyzing medical images and refining medical imaging protocols [25, 26].

3.4.1 EfficientNetV2S

By adjusting depth, breadth, and resolution in equal measure, EfficientNet can achieve great accuracy while maintaining processing economy. The design is that of a CNN, or convolutional neural network. In terms of efficient feature extraction, this newer model shows better performance than EfficientNetV1 across all important benchmarks. According to the authors, EfficientNet models have several drawbacks:

- Training takes much longer when it works with big image files.
- Using depth-wise convolutions causes a delay in the first layers.
- Scaling consistently throughout all stages is not desirable.

The EfficientNetV2 network family aims to solve these problems. Training needs to optimize criteria for better performance accuracy, speed, and parameter size. These networks employ Fused-MBConv in combination with MBConv, they also make use of training-aware Neural Architecture Search (NAS) and scaling. Compared to EfficientNetV1 models, the resulting networks train four times faster and employ 6.8 times fewer parameters, as stated by the study's authors. In addition, as the size of the images increases, they use progressive learning techniques to systematically enhance data regularization and augmentation, leading to better model performance and efficiency. Unlike its predecessor, EfficientNetV2 extensively uses MBConv and Fused-MBConv in the first layers with a lower MBConv expansion ratio to minimize memory access demands. EfficientNetV2 has risen to the top of the convolutional classification network category thanks to its Fused-MBConv operation and small but painstaking optimization efforts. In the EfficientNetV2 family, there are three main models: EfficientNetV2 includes three network models: S, M, and L, they are: EfficientNetV2S, EfficientNetV2M, and EfficientNetV2L. The S model has two scaled-down versions, the M and the L. EfficientNetV2S is one in the family of EfficientNetV2 network models [27, 28].

4. Proposed Model

In this work, two novel CNNs based models and transfer learning architectures are presented. Then, present a CNN model based on Caputo's derivative, a fractional calculus style that grants the models more ability to recognize complex patterns and dependencies in the data. The application of the Caputo derivative leads to improved processing of non-local information, enabling the use of more robust models in addressing a number of challenges faced by traditional differential methods. This study explores the interaction between these advanced techniques in order to improve performance, efficiency, and accuracy of the proposed models, which can be applied to a variety of complex applications, including image processing and signal analysis, etc. Before constructing these architectures, the following mutual steps are included in the suggested system:

1- Reading images from the dataset.

2- Based on the directory for each image, create a class label.

3- In Python, utilize the TensorFlow bunch:

- Image files reading.
- Image decoding (so that the content of the image file is converted into a three-dimensional tensor: width, height, color channels (where color channel = three).

4- Pre-processing steps:

- Image resizing to (300 by 300) fix square size.
- Image augmentation using two methods: Randomly flip an image horizontally and vertically. These are medical images and any change in them affects the prediction decision. Therefore, the authenticity of the image must be ensured, and the images must be generated without changing the structure or content. Therefore, other methods cannot be used.

5- Dataset splitting using 80:20 training and testing datasets, respectively.

6- The two proposed architectural models are built one by one.

7- As part of the training process, compile, and fitting the model on the training data.

8- Predictions are made using the trained model, and assess the model's accuracy, recall, precision, F1-score, MCC, and AUC measurements.

- New CNN Architecture Based on Blocks Technique

The first proposed model for detecting hypertrophic cardiomyopathy (HCM) is based on CNN layers arranged in blocks, as shown in Figure 3. Each block consists of the following components:

1- Batch normalization layer followed by convolution layer with a filter size of (1, 1) and N filters.

2- Batch normalization layer followed by convolution layer with a filter size of (3, 3) and (1.5 * N) filters.

3- The batch normalization layer followed the convolution layer with a filter size of (1, 1) and N filters.

4- Skip the connection layer to avoid the vanishing gradient problem and extract additional features and knowledge at deeper layers.

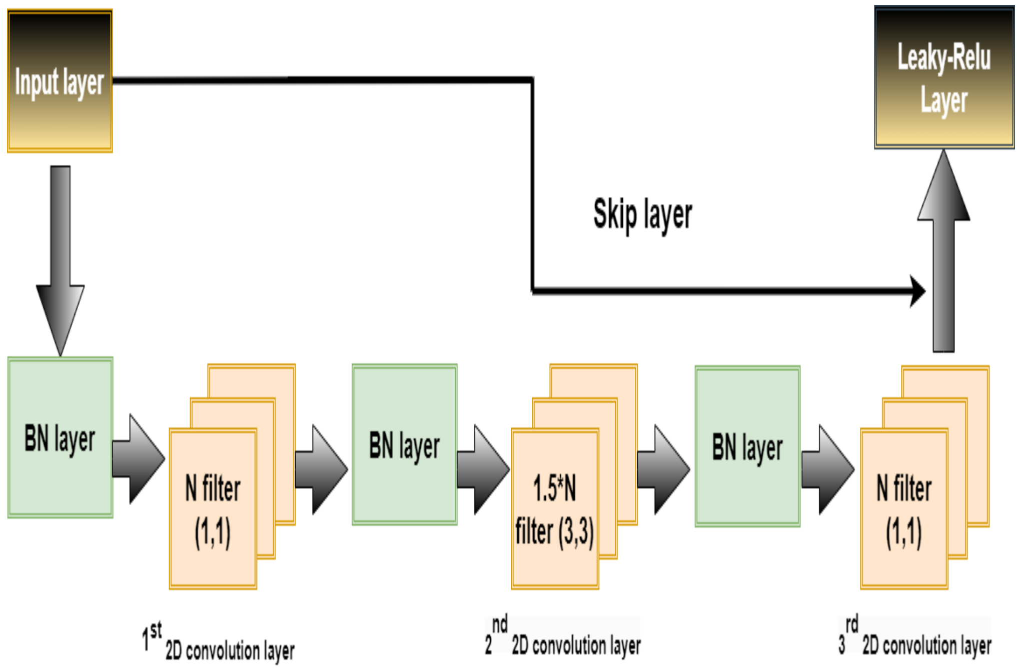


Figure 3: The CNN identity block

The general CNN model used for feature engineering based on the blocks shown in Figure 3 and constructed as shown in Figure 4, with the following steps:

1- 2D convolution layer with 32 filters and a size of (3, 3) followed by batch normalization layer.

2- 2D convolution layer with 64 filters followed by 2D max pooling layer with a size of (2, 2) and batch normalization layer.

3- 2D convolution layer with 128 filters.

4- Calling the CNN block with 128 filters followed by a 2D max pooling layer with a size of (2, 2) and batch normalization layer.

5- 2D convolution layer with 256 filters.

6- Calling the CNN block with 256 filters followed by a 2D max pooling layer with a size of (2, 2) and batch normalization layer.

7- 2D convolution layer with 512 filters.

8- Calling the CNN block with 512 filters.

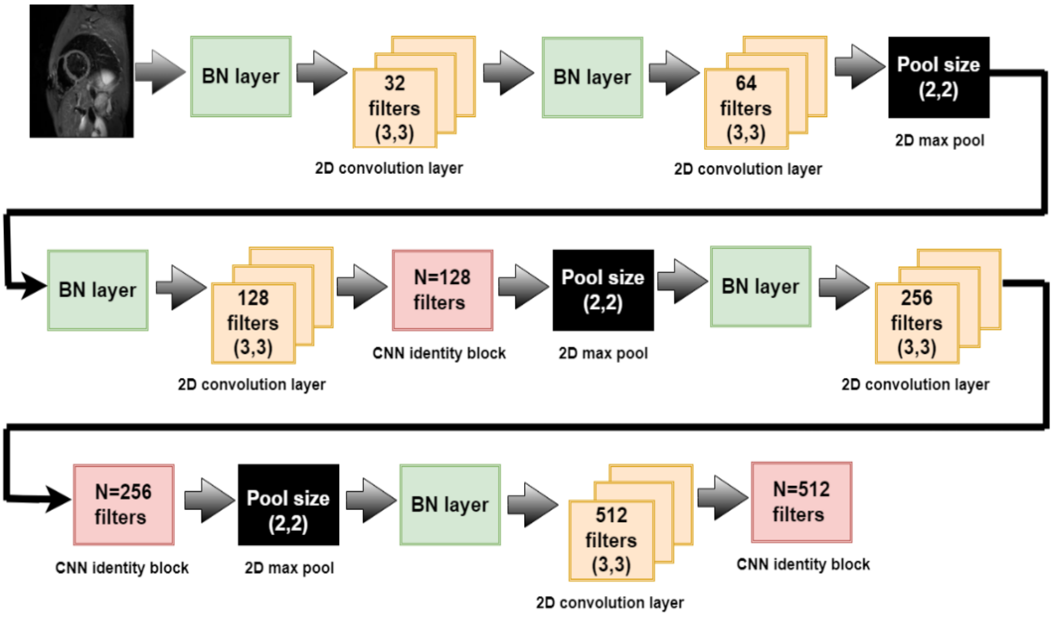


Figure 4: The proposed CNN architecture.

The feature engineering process of CNN is now complete. As depicted in Figure 5, the CNN architecture serves as the basis for the entire proposed classification model. It will be structured using the CNN feature extraction and an additional ANN for classification purposes. The steps are as follows:

1- Refer to the proposed block-based CNN in Figure 4 as the feature extraction network.

2- After adding a dense layer with 512 units, use leaky ReLU activation then add dropout at rate 0.5.

3- After adding another dense layer containing 256 units and a leaky ReLU activation function, a dropout layer with rate (0.3) is included.

5- Finally, add an output layer with a dense layer with one unit and a sigmoid activation function.

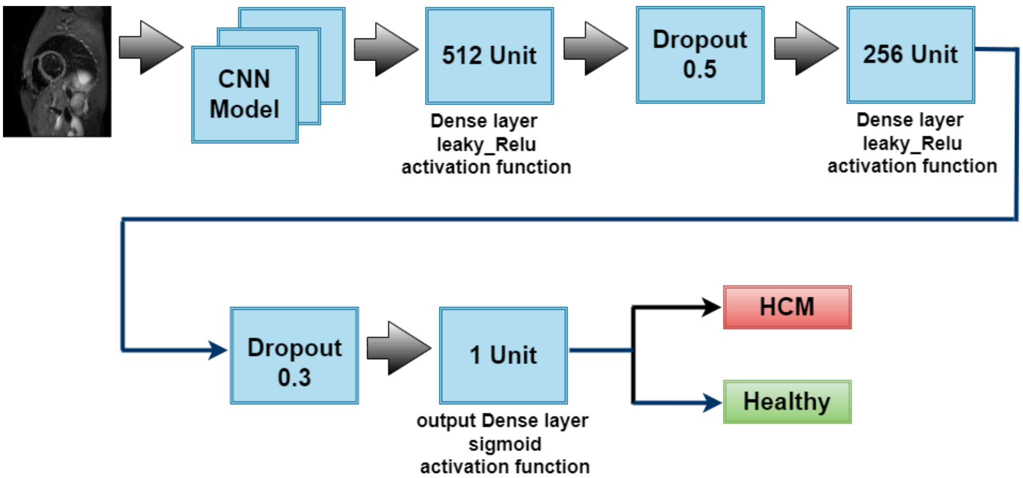


Figure 5: The proposed classification model using block-based CNN.

4.2 A New CNN Architecture Utilizing Transfer Learning

In this work, the EfficientNetV2S is used as a feature-engineering network followed by an ANN (Artificial Neural Network) classifier for HCM detection, as in Figure 6. The architecture of this model consists of:

1- Utilizing EfficientNetV2S with convolution layers, excluding the top classifier. Only the top 400 layers are trainable.

2- Additional layers are added and connected to the EfficientNetV2S shallow neural network as follows:

- Add a dense layer with 512 units and a leaky ReLU activation function, then a dropout layer with a rate of 0.5.
- A dense layer for output using one unit and a sigmoid activation function.

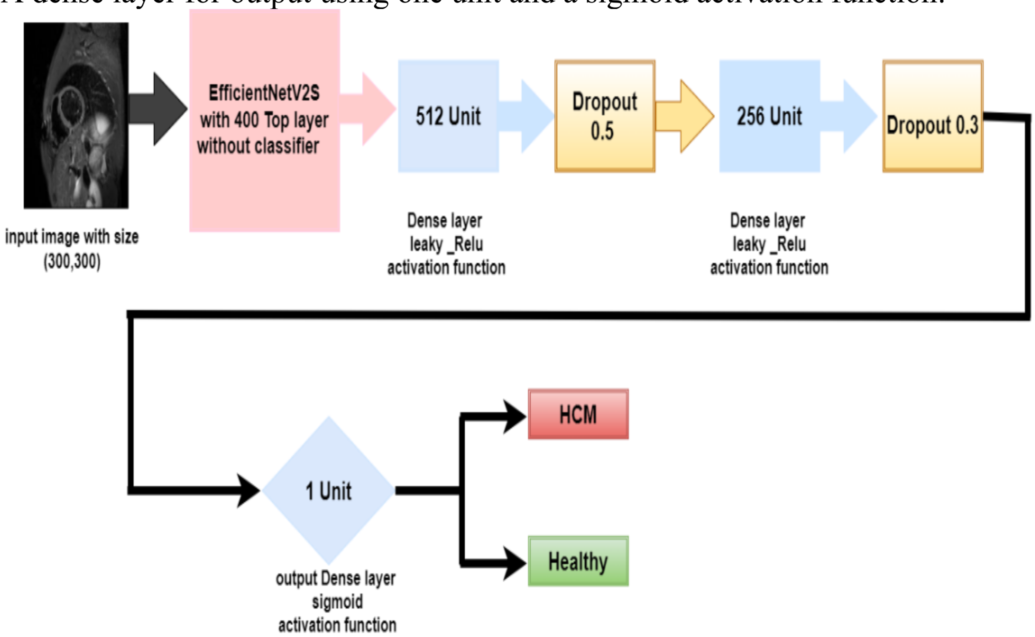


Figure 6: The second proposed classification model based on efficientNetV2S**.**

4.3 The Proposed CNN Based on a Caputo Derivative Layer Model

The third model maintains the same architecture as the first, with one significant modification: a Caputo derivative layer is integrated after the second block, as illustrated in Figure 7. To test new methods to improve the performance of the CNN, this layer was added to the network. The incorporation of Caputo derivative technology helps the model better recognize and process complex patterns within the data.

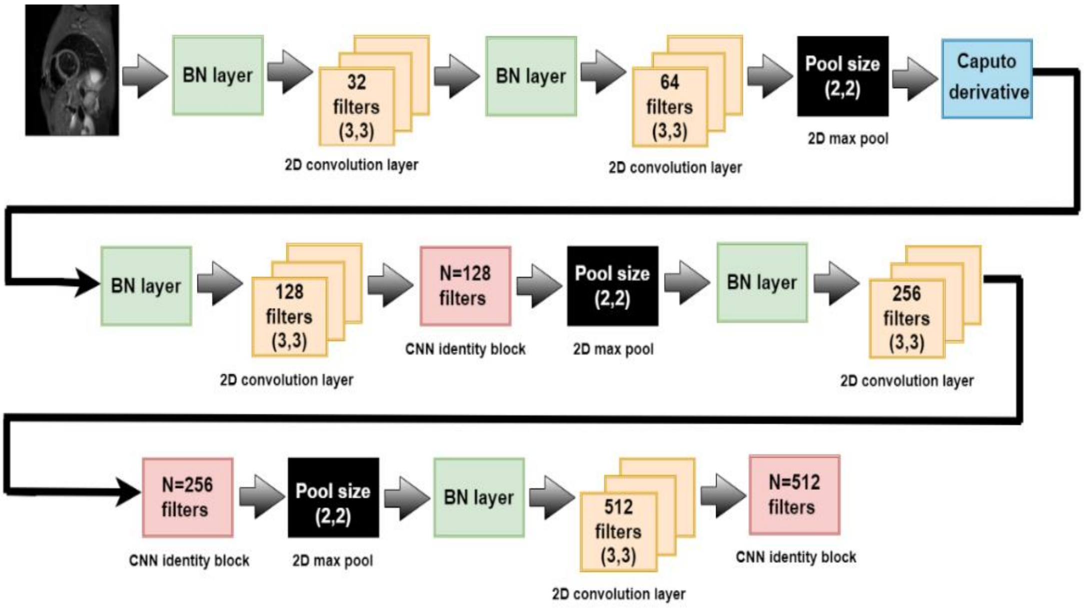


Figure 7: The suggested architecture for the convolutional neural network incorporates the Caputo derivative.

5. Evaluation Metrics

Binary classification is a typical application of machine learning and computational statistics. As researchers use confusion matrices to quantify binary classification issues, it is vital to provide a consistent statistical rate that appropriately depicts the quality of a binary prediction. Depending on the goal of the experiment, researchers may use various statistical rates to assign values to binary classifications and the confusion matrices that go along with them. Although it is a significant problem in ML, no consensus has been achieved about this. Accuracy and F1 scores computed using confusion matrices are two often used metrics for binary classification issues. Matthews Correlation Coefficient (MCC) rates items correctly while considering the number of positives and negatives in available data. It earns a high score only if the prediction does well in all four categories of the confusion matrix: true positives, true negatives, false positives, and false negatives. An important evaluation tool, especially when we only know one confusion matrix threshold is the area under the receiver operating characteristic curve, which is also called ROC AUC or AUROC [28-30]. To produce a forecast about the category of the data case, the classification model assigns a positive or negative label to each sample. After the classification procedure is finished, there are four potential results for each sample, as shown in Figure 8.

• Actual positives that are correctly predicted; true positives (TP).
• Actual negatives that are correctly predicted; true negatives (TN).
• Actual negatives that are wrongly predicted; false positives (FP).
• Actual positives that are wrongly predicted; false negatives (FN).

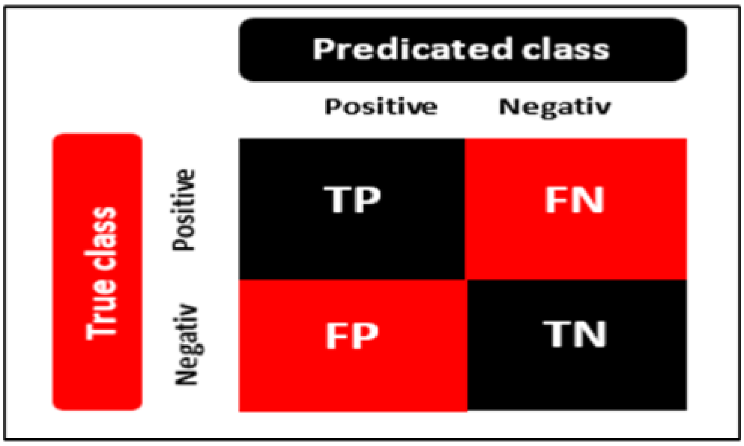


Figure 8: The basic structure of a confusion matrix.

Accuracy_ score: $ACC = \frac{TP+TN}{TP+FN+FP+TN}$ …..(4)

Precision score: $Precision = \frac{TP}{TP+FP}$ …..(5)

Recall score: $Recall = \frac{TP}{TP+FN}$ …..(6)

F1_score: $F1_{\_Score} = 2 * \frac{Precision*Recall}{Precision+Recal}$ …..(7)

$$\boldsymbol{AUC} = \int_0^1 TPR(FPR)d(FPR)$$

Where:

TPR*(True Positive Rate) is calculated as:* $TPR = \frac{TP}{TP+FN}$ …..(8)

FPR *(False Positive Rate) is calculated as:* $FPR = \frac{FP}{FP+TN}$

MCC: $\boldsymbol{MCC} = \frac{\boldsymbol{TN\times TP - FN\times FP}}{\sqrt{\boldsymbol{(TP+FP).(TP+FN).(TN+FP).(TN+FN)}}}$ …..(9)

This evaluation metrics in Eques. (4 to 9), including accuracy, precision, recall, F1-score, AUC, and MCC, are critical in understanding the model's performance, particularly for hypertrophic cardiomyopathy (HCM) diagnosis. Each metric offers unique insights, and their significance in the context of HCM is as follows:

1. Accuracy: To understand model performance accuracy, results can lose accuracy when used with datasets that contain an unbalanced number of classes. However, in conjunction with other metrics, it offers a baseline understanding of prediction success
2. Precision: Precision highlights the proportion of correctly identified HCM cases among all samples predicted as HCM. At the level of clinic workflow, this corresponds directly to the minimization of false positives (FP), which is imperative to avoid unnecessary anxiety, further examinations as well as potential overtreatment of HCM negative patients.
3. Recall (Sensitivity): The model's ability to correctly identify actual HCM cases is called recall. This is important to ensure that the false negatives (FN), where true HCM cases are missed, are minimized, which can delay a diagnosis and treatment, leading to worsening patient outcomes.
4. F1-Score: is the harmonic mean between precision and recall, which makes a reasonable form of tradeoff between false positives and false negatives a good metric. This is

especially true when under diagnosis (FN) as well as over diagnosis (FP) has severe subsequent consequences.

5. AUC (Area Under the ROC Curve): The model's ability to classify HCM and non HCM cases across different decision thresholds is quantified by the AUC. It means a high AUC results in a strong discriminative ability, thus maintaining robustness in clinical settings.

6. MCC (Matthews Correlation Coefficient): The MCC is a holistic measure of all four components of confusion matrix (TP, FP, TN, FN). Especially for imbalanced datasets, it gives a reliable way to understand how well the model can predict correctly.

Clinical implications of false positives and false negatives must be carefully managed in HCM diagnosis since the tradeoffs between them are different. Missed diagnoses, including false negative results (FN), are correlated with miss diagnosis and can lead to delayed treatment and worsened outcomes for patients with HCM. This indicates that the recall has to be high. False positives (FP), however, can result in uselessly pursued follow-up tests and procedures, which increases healthcare costs and leads to an increase in patient stress. As mentioned, trading these off is balanced with metrics like F1 score and MCC, as these metrics give us a better picture of how the model performs. This points towards the need for fine tuning the thresholds to bridge the gap between model predictions and clinical priorities and comparative diagnostic sensitivity and specificity.

6. Results and Discussion

The remainder of this paper includes a detailed analysis of the test results from the HCM dataset. The evaluation of the effectiveness and performance of the proposed classification models is the main objective. Each classification model's results are explained in detail for a complete comparison. The importance of adding a Caputo derivative layer to the convolutional neural network (CNN) based model will be drawn out in this comparison. This analysis shows how this novel contribution changes the overall classification accuracy and robustness of the approach.

The classification employed in this research paper is binary classification. The dataset utilized comprises 59,145 images of the human heart, divided into 37,290 images from healthy individuals and 21,855 images from patients with hypertrophic cardiomyopathy. Figure 9 illustrates the distribution of the dataset.

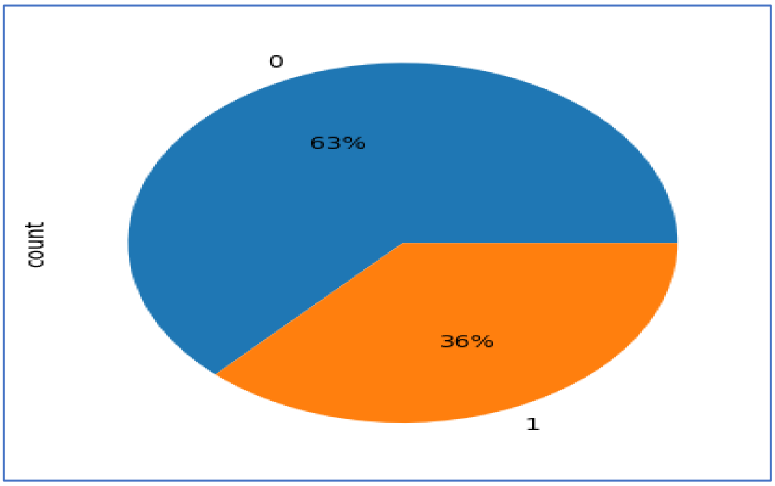
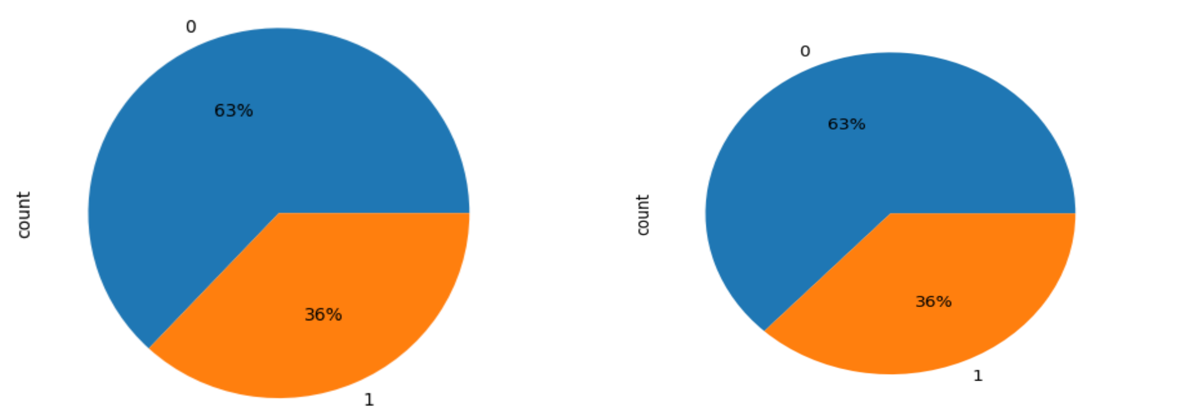


Figure 9: Distribution of samples on two classes of dataset.

The data set was separated into training and testing, with a ratio of (8:2), and the function (stratify=dataset) is used to be able to maintain the division ratio for healthy people (label 0 for health) and patients (label 1 for sick) in the data set, in both, as shown in Figure 10.



(a) Training data set. Label 0:29832, and Label 1:17484.

(b) Testing data set. Label 0:7458, and Label 1:4371.

Figure 10: (a) Ratio of healthy and sick people in the training dataset, (b) Ratio of healthy and sick people in the testing dataset.

6.1 Performance Results of Block-Based CNN Model

Based on the presented results, it is clear that the CNN-based model achieved 91.60% accuracy in epoch 10, as in Figure 11, demonstrating the stability and superior performance of the model in less time. Figure 12 shows the confusion matrix to visualize the performance results of the proposed model, where the true positive (TP) achieved by the proposed block-based CNN model is 7000 samples and the true negative (TN) is 3800, the analysis summary of the model in this work is shown in Table 1.

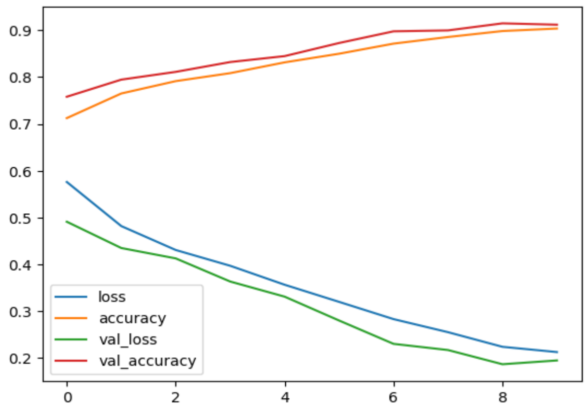


Figure 11: CNN training results with augmentation method

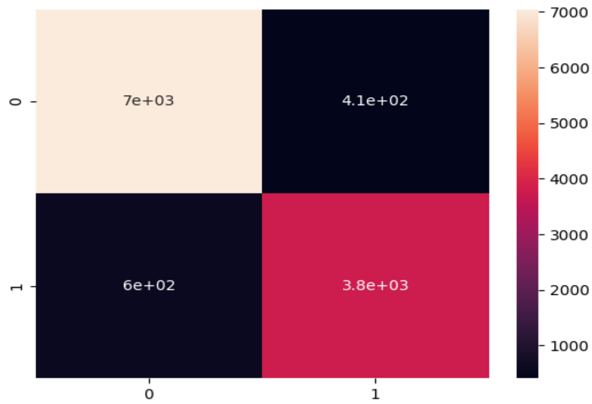


Figure 12 : Confusion matrix of the first proposed CNN-based block model

The results shown in Table 1 were achieved by:

• Performing experiments under consistent computational conditions.

• Ensuring that the model is tested on data that was not used for training to avoid any potential bias.

Table 1: Results of analysis of the proposed CNN model.

Model	Accuracy	Precision	Recall	F1 Score	AUC	MCC
CNN	91.60	90.25	86.40	88.36	90.45	81.71

6.2 Performance Results of EfficientNetV2S Model

For the second model based on EfficientNetV2S, the training was performed for 5 epochs, which was enough for the model to reach a stable state with minimal error and achieve its highest accuracy of 98.62%. The same augmentation techniques were applied for the first CNN-based model to make a fair comparison. Figure 13 shows the training results of the EfficientNetV2S model. Based on the results provided, it is evident that the EfficientNetV2S model achieved the highest accuracy of 98.62%. It reached a stable state with this peak accuracy and minimal error by epoch 5. In contrast, the initial CNN-based model attained an accuracy of 91.60% by epoch 10, highlighting the stability and superior performance of the EfficientNetV2S model in a shorter timeframe. Figure 14 shows the confusion matrix to visualize the performance results of the proposed model EfficientNetV2S, where the true positive (TP) achieved by the EfficientNetV2S model is 7400 samples and the true negative (TN) is 4300.

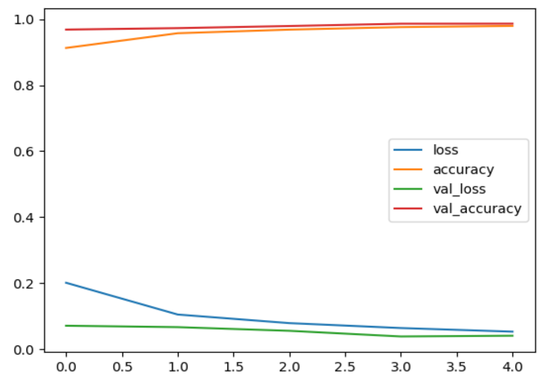


Figure 13: EfficientNetV2S training results with augmentation method

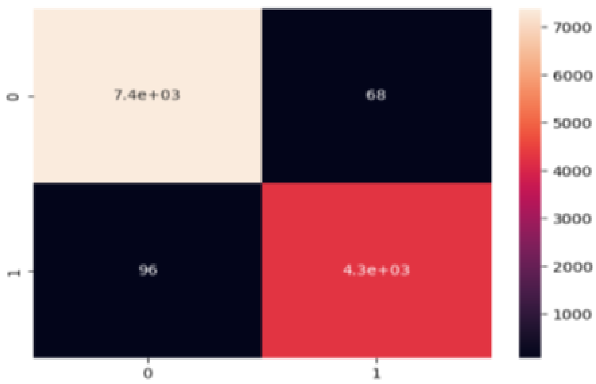


Figure 14 : Confusion matrix of the second proposed EfficientNetV2S model

Table 2: Results of the analysis of the second proposed model EfficientNetV2S.

Model	Accuracy	Precision	Recall	F1 Score	AUC	MCC
EfficientNetV2S	98.62	98.50	97.82	98.21	98.45	97.03

Table 3 presents a comparison between the CNN model and the EfficientNetV2S model. The EfficientNetV2S model's ability to extract deeper and more complex features through its pre-trained network architecture accounts for its superior performance compared to the simpler CNN-based model. In other words, the second proposed model resulted in fewer misclassified samples, leading to improvements in all evaluation metrics utilized. This enhancement is attributed to the EfficientNetV2S model's capacity to extract deeper features and patterns, which were instrumental in decision-making through the robust pre-trained EfficientNetV2S network.

Table 3: Summary of the comparative analysis of the two proposed models (CNN and EfficientNetV2S)

Model	Accuracy	Precision	Recall	F1 Score	AUC	MCC
CNN	91.60	90.25	86.40	88.36	90.45	81.71
EfficientNetV2S	98.62	98.50	97.82	98.21	98.45	97.03

6.3 Performance Results of Block-Based CNN Model with Caputo Layer

This section presents a comprehensive analysis of the results obtained from the proposed model following the introduction of the Caputo layer. This addition has significantly improved the model's performance. Specifically, the effectiveness of the original CNN model will be highlighted, showing how this modification enhances its performance. Table 4 presents the results of incorporating the Caputo layer into the original CNN model. The results of this model are visualized in Figure 15.

Table 4: Summary of the analysis results of the proposed CNN model with the Caputo layer.

Model	Accuracy	Precision	Recall	F1 Score	AUC	MCC
CNN with Caputo	92.47	93.57	85.68	89.36	91.14	83.76

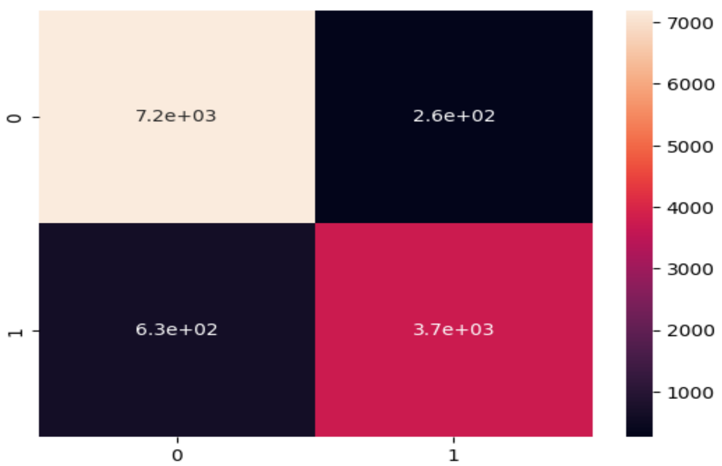


Figure 15: Confusion matrix of the proposed CNN model with Caputo layer

Table 5 presents a comparison between the traditional CNN model and the modified CNN model that utilizes the relative Caputo derivative. The results indicate that the CNN model incorporating the Caputo derivative achieved superior accuracy, precision, F1 score, AUC, and MCC compared to the traditional CNN model. However, it is noteworthy that the recall was slightly lower in the modified CNN model than in the traditional model. This comparison demonstrates that incorporating the Caputo derivative into the CNN model led to improvements in most metrics, suggesting that this modification is effective in enhancing the model's accuracy in classifying hypertrophic cardiomyopathy. To optimize the randomly initialized weights of neural networks (NNs), the fractional-order derivative technique can be applied to activation functions. Innovative adaptation strategies aimed at improving convergence, enhancing generative characteristics, and accelerating training are emerging in computer-assisted research fields due to the development of hybrid fractional-order derivative definitions. Signal processing and machine learning increasingly utilize gradient descent algorithms and fractional-order derivatives iteratively.

Table 5: Comparison of CNN and CNN with Caputo Derivative for HCM Classification

Model	Accuracy	Precision	Recall	F1 Score	AUC	MCC
CNN without Caputo	91.60	90.25	86.40	88.36	90.45	81.71
CNN with Caputo	92.47	93.57	85.68	89.36	91.14	83.76

The results indicated that incorporating the Caputo derivative with the convolutional neural network (CNN) significantly improved the model's performance across various evaluation metrics compared to the model that did not utilize the Caputo derivative. The Caputo derivative is particularly effective at capturing temporal and structural relationships within the data, thereby enhancing the model's capacity to represent complex patterns in cardiac CMR images.

In order to identify the more effective model, Table 6 presents the results of all proposed models and works as a comparative analysis. After checking, it was found that the better performing model via all assessment metrics used in this study is the second model, EfficientNetV2S.

Table 6: Comparison analysis of all proposed models.

Model	Accuracy	Precision	Recall	F1 Score	AUC	MCC
CNN	91.60	90.25	86.40	88.36	90.45	81.71
EfficientNetV2S	98.62	98.50	97.82	98.21	98.45	97.03
CNN with Caputo	92.47	93.57	85.68	89.36	91.14	83.76

6.4 Evaluation and Comparison

In this section, a comparison study of the proposed classification models and the hypertrophy detection formerly studied is presented. It also discusses the utilized classifiers and the classification accuracy. Table 7 compares the classification performance of the suggested model with that of similar studies. While some models demonstrate competitive performance, the EfficientNetV2S model outperforms most others in terms of accuracy and generalizability. It is important to note that some referenced studies employed augmentation techniques on the entire dataset before splitting the data, which may introduce bias in testing performance. In contrast, the proposed approach strictly applied augmentation after the dataset was split, ensuring that the test set remained unseen by the model. This method provides a more accurate reflection of real-world performance and generalizability. The newly proposed second classification technique demonstrates improved accuracy when compared to the methods referenced in Table 7. While Model 1 achieves a higher recall than the third proposed model and has an accuracy level comparable to that of the third model, it employs data augmentation on the original dataset before splitting the data into training and testing sets. Consequently, the testing dataset includes augmented images, which introduces bias since the model is evaluated on samples it has already encountered during training. As a result, Technique 1 lacks generalizability and realism in its testing performance. Therefore, the proposed technique is superior in terms of generalizability.

Table 7: The comparison between the proposed classification models and previous works

Ref.	Year	Classifier	Accuracy	Precision	Recall	F1 Score	AUC	MCC
[1]	2021	Deep CNN	98.53	/	98.70	/	/	/
[6]	2022	Deep CNN	/	/	78.05	/	0.695	/
[9]	2023	CNN/T.L(VGG-19, ResNet-50)	(94.21, 96.81)	(94.28, 95.00)	(94.00, 95.28)	(94.00, 94.85)	/	/
[5]	2024	CNN	94.71	96.97	91.21	94.85	/	/
[7]	2024	CNN/T.L	93.37	93.66	93.37	93.42	/	/
[8]	2024	CNN	/	/	/	/	94.06	/
Proposed Models		CNN-block	91.60	90.25	86.40	88.36	90.45	81.71
		EfficientNetV2S	98.62	98.50	97.82	98.21	98.45	97.03
		CNN with Caputo	92.47	93.57	85.68	89.36	91.14	83.76

6.4 Practical Implications and Real-World Deployment

The integration of deep learning models, such as the Caputo derivative-based CNN and EfficientNetV2S, into clinical workflows offers promising avenues for advancing healthcare diagnostics, particularly in the detection of hypertrophic cardiomyopathy (HCM) using CMR imaging. These models provide accurate and efficient classification, potentially serving as supplementary diagnostic tools for cardiologists.

These models will make real-world deployment possible by integrating into electronic health record (EHR) systems to access patient data and associated diagnostic results. Furthermore, these models could be easily included in picture archiving and communication systems (PACS) so that the analysis of medical images is streamlined within existing clinical infrastructure. Nevertheless, these concepts need to be addressed in real-world implementation. It is essential to ensure that the model remains robust to various imaging protocol variants, different scanner types, and patient demographics and adheres to regulations, including Health Insurance Portability and Accountability Act (HIPAA) for data security and patient privacy, General Data Protection Regulation (GDPR), and so on. However, more validation is needed with multicenter trials to confirm AI models for clinical adoption, as they need to be validated across various populations and imaging set up.

These models can achieve high accuracy and handle augmented datasets, indicating they have the potential to minimize diagnostic errors, to discriminate between subtle features of HCM. These models integrate advanced preprocessing techniques and employ the Caputo derivative for improved generalization and form a building block to design scalable and interoperable solutions in healthcare. Future research will further extend their versatility to serve other medical imaging modalities and further expand disease classification capabilities for wider clinical impact.

7. Conclusion

The objective of this research paper is to classify hypertrophic cardiomyopathy using cardiac magnetic resonance (CMR) imaging. The system was developed and evaluated using a deep learning architecture. Image scaling, normalization, and data augmentation techniques were applied in the pre-processing phase to improve the quality of the dataset. Three basic models were used to classify CMR images of cardiomyopathy: A Caputo derivative-based model and a CNN-based model for comparison with the EfficientNetV2S-based model. The results show that the CNN model optimized with Caputo performed better compared with the standard CNN model. Key metrics for the CNN model without the Caputo derivative were as follows: Accuracy = 91.60, Precision = 90.25, Recall = 86.40, F1-Score = 88.36, AUC = 90.45, and MCC = 81.71. In comparison, the CNN model with the Caputo derivative achieved improved metrics: Accuracy = 92.47, Precision = 93.57, Recall = 85.68, F1-Score = 89.36, AUC = 91.14, and MCC = 83.76. In this work, it is shown that the use of fractional calculus, e.g., Caputo derivative increases the performance of deep learning-based models for medical image classification. Overall, the EfficientNetV2S-based model had very good performance, achieving metrics such as Accuracy = 98.62, Precision = 98.50, Recall = 97.82, F1-Score = 98.21, AUC = 98.45, and MCC = 97.03, surpassing both CNN-based variants. In future work, CNNs will be generalized to other medical imaging modalities such as electrocardiograms (ECGs), echocardiography, computed tomography, X-ray, and results will be increased by using Caputo derivative developments to increase the number of patients the models can be applied to. Real integration of these models with electronic health records and other diagnostic tools is further suggested for the complete patient evaluation. Yet, the trend will extend to other kinds of cardiomyopathy (restrictive, dilated cardiomyopathy).

Conflict of Interest: No conflict of interest has been declared by the authors.

Acknowledgments

To all those who contributed to the achievement of this research paper, the authors would like to exhibit their heartfelt gratitude.

References

[1] D. Sharifrazi *et al.*, "Hypertrophic cardiomyopathy diagnosis based on cardiovascular magnetic resonance using deep learning techniques," *Colour Filtering, Research Square,* Oct. 26, 2021. [Online]. Available: https://doi.org/10.21203/rs.3.rs-1005999/v1

[2] V. Sangha *et al.*, "Identification of hypertrophic cardiomyopathy on electrocardiographic images with deep learning," *medRxiv,* Dec. 28, 2023. [Online]. Available: https://doi.org/10.1101/2023.12.23.23300490

[3] H. Wu *et al.*, "The predictive value of deep learning-based cardiac ultrasound flow imaging for hypertrophic cardiomyopathy complicating arrhythmias," *European Journal of Medical Research,* vol. 28, no. 1, p. 36, Jan. 19, 2023. [Online]. Available: https://doi.org/10.1186/s40001-022-00975-7

[4] R. K. Hughes *et al.*, "Improved diagnostic criteria for apical hypertrophic cardiomyopathy," *Journal of Cardiovascular Imaging,* vol. 17, no. 5, pp. 501–512, May 1, 2024. [Online]. Available: https://doi.org/10.1016/j.jcmg.2023.07.012

[5] S. Kolluri and S. Hathwar, "Diagnosing hypertrophic cardiomyopathy using machine learning models on CMRs and EKGs of the heart," *Journal of Emerging Investigators,* vol. 7, pp. 1–7, 2024. [Online]. Available: https://doi.org/10.59720/23-209

[6] J. Guo *et al.*, "Artificial intelligence study on left ventricular function among normal individuals, hypertrophic cardiomyopathy, and dilated cardiomyopathy patients using 1.5T cardiac cine MR images obtained by SSFP sequence," *British Journal of Radiology,* vol. 95, no. 1133, Article 20201060, 2022. [Online]. Available: https://doi.org/10.1259/bjr.20201060

[7] A. Demirbaş, H. Üzen, and H. Fırat, "Spatial-attention ConvMixer architecture for classification and detection of gastrointestinal diseases using the Kvasir dataset," *Health Information Science and Systems,* vol. 12, Article 32, 2024. [Online]. Available: https://doi.org/10.1007/s13755-024-00290-x

[8] H.-S. Kim *et al.*, "Color-transfer-enhanced data construction and validation for deep learning-based upper gastrointestinal landmark classification in wireless capsule endoscopy," *Diagnostics,* vol. 14, Article 591, 2024. [Online]. Available: https://doi.org/10.3390/diagnostics14060591

[9] X. A. Mary *et al.*, "Multi-class classification of gastrointestinal diseases using deep learning techniques," *Open Biomedical Engineering Journal,* vol. 17, no. 1, Article e187412072301300, 2023. [Online]. Available: https://doi.org/10.2174/18741207-v17-e230215-2022-HT27-3589-11

[10] D. Sharifrazi and R. Alizadeh, "Hypertrophic Cardiomyopathy Dataset," *Kaggle,* 2021. [Online]. Available: https://www.kaggle.com/danialsharifrazi/hypertrophic-cardiomyopathy-datase

[11] R. Razaq, E. N. AlShemmary, and Z. Lu, "A survey on bladder cancer detection and classification using deep learning algorithms," AIP Conf. Proc., vol. 3232, no. 1, Oct. 2024, pp. 1-10. AIP Publishing. [Online]. Available: https://doi.org/10.1063/5.0236313

[12] Abdulhamed MA, Hashim AN. A Survey on Detecting Deep Fakes Using Advanced AI-Based Approaches. Iraqi Journal of Science, Vol. 65, No. 9, pp: 5254-5269, Sep 30 (2024). Available: https://ijs.uobaghdad.edu.iq/index.php/eijs/article/view/9726

[13] K. C. Tan and D. Tao, *Machine Learning: Foundations, Methodologies, and Applications.* Springer, 2021. [Online]. Available: https://link.springer.com/bookseries/16715

[14] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data,* vol. 8, p. 53, Dec. 2021. [Online]. Available: https://doi.org/10.1186/s40537-021-00444-8

[15] R. Balestriero and R. G. Baraniuk, "Batch normalization explained," *arXiv preprint,* arXiv:2209.14778, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2209.14778

[16] S. Saxena, "Introduction to batch normalization," *Analytics Vidhya,* Mar. 2021. [Online]. Available: https://www.analyticsvidhya.com

[17] S. Santurkar *et al.*, "How does batch normalization help optimization?" in *Advances in Neural Information Processing Systems,* vol. 31, 2018. [Online]. Available: https://papers.nips.cc/paper/2018

[18] Z. C. Oleiwi, E. N. AlShemmary, and S. Al-Augby, "Multi-label Classification Technique of Chest X-Rays Image Based Cardiomegaly Disease Prediction," 2023 International Conference on Information Technology, Applied Mathematics and Statistics (ICITAMS), Mar. 2023, pp. 134-139. IEEE. [Online]. Available: https://doi.org/10.1109/ICITAMS57610.2023.10525488

[19] Q. Xu *et al.*, "Skip-layer network with optimization method for domain adaptive detection," *PLoS One,* vol. 17, no. 3, p. e0263748, 2022. [Online]. Available: https://doi.org/10.1371/journal.pone.0263748

[20] H. Ahn and C. Yim, "Convolutional neural networks using skip connections with layer groups for super-resolution image reconstruction based on deep learning," *Applied Sciences,* vol. 10, no. 6, p. 1959, 2020. https://doi.org/10.3390/app10061959

[21] A. Salehin and D. K. Kang, "A review on dropout regularization approaches for deep neural networks within the scholarly domain," *Electronics,* vol. 12, no. 14, p. 3106, 2023. https://doi.org/10.3390/electronics12143106

[22] C. Garbin, X. Zhu, and O. Marques, "Dropout vs. batch normalization: An empirical study of their impact on deep learning," *Multimedia Tools and Applications,* vol. 79, no. 19, pp. 12777–12815, 2020. https://doi.org/10.1007/s11042-019-08453-9

[23] M. Kumar *et al.*, "Enhancing neural network classification using fractional-order activation functions," *AI Open,* vol. 5, pp. 10–22, 2023. https://doi.org/10.1016/j.aiopen.2023.12.003

[24] L. Fahem, et al., "Integrating Swin Transformer with Fuzzy Gray Wolve Optimization for MRI Brain Tumor Classification", International Journal of Intelligent Engineering and Systems, Vol.17, No.6, 2024. https://doi.org/ 10.22266/ijies2024.1231.30

[25] A. Hosna *et al.*, "Transfer learning: A friendly introduction," *Journal of Big Data,* vol. 9, no. 1, p. 102, Oct. 2022. [Online]. Available: https://doi.org/10.1186/s40537-022-00652-w

[26] P. Grd *et al.*, "Transfer learning with EfficientNetV2S for automatic face shape classification," *Journal of Universal Computer Science,* vol. 30, no. 2, Feb. 2024. [Online]. Available: https://doi.org/10.3897/jucs.104490

[27] Pacal *et al.*, "Enhancing EfficientNetV2 with global and efficient channel attention mechanisms for accurate MRI-based brain tumor classification," *Cluster Computing,* pp. 1–26, May 2024. [Online]. Available: https://doi.org/10.1007/s10586-024-04532-1

[28] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics,* vol. 21, pp. 1–3, Dec. 2020. [Online]. Available: https://doi.org/10.1186/s12864-019-6413-7

[29] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," *BioData Mining,* vol. 16, no. 1, p. 4, Feb. 2023. [Online]. Available: https://doi.org/10.1186/s13040-023-00322-4

[30] M. Issa *et al.*, "Detection of arrhythmias and myocardial infarction using SVM and ANN algorithms," *Baghdad Science Journal,* vol. 21, no. 8, pp. 2757–2770, 2024. [Online]. Available: https://doi.org/10.21123/bsj.2024.8168