Black Sea Journal of Engineering and Science
doi: 10.34248/bsengineering.1496991

Open Access Journal
e-ISSN: 2619 – 8991

Research Article
Volume 7 - Issue 5: 1050-1065 / September 2024

OPTIMIZING CAPSULE NETWORK PERFORMANCE WITH ENHANCED SQUASH FUNCTION FOR CLASSIFICATION LARGE-SCALE BONE MARROW CELLS

Amina FARIS ABDULLA AL RAHHAWI[1], Nesrin AYDIN ATASOY[2]*

[1]*Karabük University, The Institute of Graduate Studies, 78050, Karabük, Türkiye*
[2]*Karabük University, Faculty of Engineering, Department of Computer Engineering, 78050, Karabük, Türkiye*

Abstract: Capsule networks (CapsNet) have emerged as a promising architectural framework for various machine-learning tasks and offer advantages in capturing hierarchical relationships and spatial hierarchies within data. One of the most crucial components of CapsNet is the squash function, which plays a pivotal role in transforming capsule activations. Despite the success achieved by standard squash functions, some limitations remain. The difficulty learning complex patterns with small vectors and vanishing gradients are major limitations. Standard squash functions may struggle to handle large datasets. We improve our methodology to enhance squash functions to address these challenges and build on our previous research, which recommended enhancing squash functions for future improvements. Thus, high-dimensional, and complex data scenarios improve CapsNet's performance. Enhancing CapsNet for complex tasks like bone marrow (BM) cell classification requires optimizing its fundamental operations. Additionally, the squash function affects feature representation and routing dynamics. Additionally, this enhancement improves feature representation, preserves spatial relationships, and reduces routing information loss. The proposed method increased BM data classification accuracy from 96.99% to 98.52%. This shows that our method improves CapsNet performance, especially in complex and large-scale tasks like BM cells. Comparing the improved CapsNet model to the standard CapsNet across datasets supports the results. The enhanced squash CapsNet outperforms the standard model on MNIST, CIFAR-10, and Fashion MNIST with an accuracy of 99.83%, 73%, and 94.66%, respectively. These findings show that the enhanced squash function improves CapsNet performance across diverse datasets, confirms its potential for real-world machine learning applications, and highlight the necessity for additional research.

Keywords: Capsule networks, Enhanced squash function, Imbalanced dataset

***Corresponding author:** Karabük University, Faculty of Engineering, Department of Computer Engineering, 78050, Karabük, Türkiye
E mail: nesrinaydin@karabuk.edu.tr (N. AYDIN ATASOY)
Amina FARIS ABDULLA AL RAHHAWI https://orcid.org/0000-0001-9090-326X
Nesrin AYDIN ATASOY https://orcid.org/0000-0002-7188-0020

Received: June 6, 2024
Accepted: September 12, 2024
Published: September 15, 2024

Cite as: Faris Abdulla Al Rahhawi A, Aydın Atasoy N. 2024. Optimizing capsule network performance with enhanced squash function for classification large-scale bone marrow cells. BSJ Eng Sci, 7(5): 1050-1065.

1. Introduction

Bone marrow (BM) cells affect human health. They facilitate the correct functioning of human blood. Many cell types make up human blood. These cells support tissue repair and stop bleeding and infection resistance in the body (Girdhar et al., 2022). The human immune system depends critically on blood cells to defend the body against bacteria and infections (Çınar and Tuncer, 2021). Although making up just 1% of the blood volume overall, white blood cells (WBCs) are essential to immunity. Carrying oxygen, red blood cells (RBCs) make up a noteworthy 45% of the blood volume (Tamang et al., 2022). Besides, white blood cells shield the body from viruses and parasites. Regarding the human immune system, they are essential (Stock and Hoffman, 2000). WBCs are mostly of five types: monocytes, lymphocytes, basophils, neutrophils, and eosinophils (Yao et al., 2021). Every type serves a different purpose in shielding the body from various germs and promoting general health and well-being. The cytoplasm of these cells contains nuclei, which sets them apart. Often used to evaluate the effectiveness of radiation and chemotherapy (Kutlu et al., 2020), blood tests are an essential diagnostic tool for many diseases (Long et al., 2021). Polio, tuberculosis, measles, HIV, and chickenpox are only a few of the illnesses that can lower lymphocyte counts. On the other side, leukemia, brucellosis, liver disease, and Bordetella pertussis disease can all raise lymphocyte levels. Leukemia, the illness, polio, and pertussis-borne diseases can cause lymphocyte counts to rise; polio, tuberculosis, measles, HIV, and chickenpox can cause them to fall. WBCs, RBCs, and plasma, platelets are considered the peripheral blood cells (PBCs) (Balasubramanian et al., 2022). Anemia, malaria, and leukemia can all be diagnosed with a PBC test (Hegde et al., 2019a). Unlike more homogeneous forms of platelets and RBCs, WBCs are the subject of much research because of their varied shape and unique subtypes (Dhal et al., 2023). The diversity of WBCs makes them of special interest for medical image segmentation and classification (Agustin et al., 2021). Hematologists' workload has been much reduced by computer-aided automated white blood cell

classification. Masses of data can be processed quickly and accurately by these methods. These automated techniques can be categorized essentially into three groups. Known as TIP, the first technique extracts feature from WBC images by means of threshold functions derived from mathematical relationships and preset parameters. With the aid of these cutoffs, classification of various white blood cell types is facilitated. Over time, several TIP approaches have been honed and enhanced; these include pixel template matching techniques as in (Ghosh et al., 2010; Hegde et al., 2019b; Mohamed et al., 2012; Rezatofighi et al., 2009), fuzzy divergence, modified thresholding, Grayscale contrast, and Gram-Schmidt orthogonality. Second, machine learning: As computing power has increased and conventional image processing has shown flaws, researchers have started using machine learning (ML) algorithms such as support vector machines (SVM), Bayesian classifiers, and random forest models for WBC classification (Gautam et al., 2017; Mirmohammadi et al., 2021). These techniques enhance classification accuracy by making use of unique morphological features of WBCs. Statistical, geometric, wavelet and textural features are extracted from images through analysis. The most pertinent features are chosen through a feature selection process that runs through all these features. To achieve accurate identification, the features that were chosen are subsequently fed into classifiers like Bayesian and SVM. Third, DL: convolutional neural networks (CNNs) and other deep learning structures have recently seen a surge in popularity for WBC classification jobs. Deep learning techniques automatically extract features from images, leading to improved classification accuracy (Aydin Atasoy and Faris Abdulla Al Rahhawi, 2024; Dayı et al., 2023), in contrast to conventional ML methods that depend on human feature extraction.

Historically, the classification of WBCs has been carried out manually, resulting in a slow, laborious, and error-prone procedure. Hence, it is essential to develop algorithms and automated diagnostic systems to classify WBC quickly and accurately (Liu et al., 2019). The motivation behind this study is that Capsule networks (CapsNet) exhibits potential in capturing intricate data relationships but faces challenges when dealing with huge complex data sets due to limitations in traditional squash functions. This article introduced an enhanced squash function to improve CapsNet performance, especially when dealing with high-dimensional and complex datasets. We aim to improve feature representation and classification accuracy by optimizing the fundamental operations, focusing on BM cell classification tasks. This research represents a significant extension of our previous contributions (Aydın Atasoy and Faris Abdulla Al Rahhawi, 2024), improving the effectiveness and applicability of CapsNet in real-world applications. The contributions of the presented model are as follows:

- We present an enhanced squash function optimized for CapsNet. The improved function addresses the limitations of standard squash and improves feature representation.
- Our methodology significantly improved the classification of 21 BM cells, which increased the classification accuracy for bone marrow cells from 96.99% to 98.52%. This success is an essential step in the process of diagnosing hematological disorders.
- Imbalance classes issue have been solved using SMOTE oversampling.
- Extending evaluation to MNIST, Fashion MNIST, and CIFAR-10 datasets demonstrates that our CapsNet architecture with the enhanced squash function has consistently improved performance.
- A wide comparison is conducted between our proposed methods and several previous CapsNet architectures.

This paper is organized and arranged as follows: Section 2 presents the literature review. In Section 3, dataset details and selected CapsNet methods are introduced. The classification performance results of the proposed models are evaluated in Section 4. The conclusion and future works are discussed in Section 5.

1.2. Literature Review

Nowadays, deep learning has proven to be effective in many classification tasks (Somuncu and Aydın Atasoy, 2022; Taşdelen and Ugur, 2021). Further, it has been used to classify images using various fields and methods. A multilayer feedforward artificial neural network like CNN is one of the most popular deep learning models. It is easy to use, runs in parallel, and achieves high success levels. CNN comes in many types, including VGGNet, AlexNet, GoogleLeNet, ResNet, EfficientNet, and ConvNeXt (Arjun Ghosh et al., 2022). CNNs consist of convolution, pooling, and a fully connected layer (LeCun et al., 2015). Kernels filter input data in the convolution layer to extract feature maps and learn image regional patterns. The pooling layer reduces feature map dimensionality by applying pooling operations, but this can lose information (Somuncu and Aydın Atasoy, 2022). Finally, the fully connected layer's activation function processes weighted inputs from each neuron in the previous layer to determine output.

It is a significant challenge to train a model to perform all possible combinations of enlargement, rotation, inversion, cropping, and zooming; it requires a substantial amount of time and data. However, the resulting model may not yield optimal outcomes. The CNN model may cause data loss during the feature extraction when filters are applied to datasets created through rotation. CNNs can handle translational invariance but cannot do so with rotational invariance (Kwabena Patrick et al., 2022; Muhammad et al., 2023; Nair et al., 2021; Ren et al., 2019) and are not sufficiently successful in training large and unbalanced data (Singh et al., 2021). CapsNet employs dynamic routing methods to prevent data loss in the pooling layer (Zhao and Huang, 2019).

Recent advancements in BC classification have led to the

development of advanced deep-learning techniques. Girdhar et al. (Girdhar et al., 2022) developed a new CNN structure for the WBC classification. An investigation was conducted on the blood cell count and detection (BCCD) dataset to evaluate the effectiveness of the CNN method as presented. The CNN method achieved a classification accuracy of 98.55% after 20 epochs. Long and colleagues (Long et al., 2021) presented BloodCaps, a capsule-based model aimed at precisely categorizing a broad spectrum of blood cells for PBC classification. Eight classes of human peripheral blood cells made up a large dataset for which the BloodCaps method was used. Its remarkable 99.3% total accuracy beats that of convolutional neural networks like InceptionV3 (98.4%), ResNet18 (95.9%), VGG16 (97.8%), and AlexNet (81.5%). Sengur et al. (2019) presented in their work a hybrid method for the WBC count that combines deep learning methods with image processing. Among the image processing methods used by the WBC images are morphological procedures, thresholding, filtering, color-to-grayscale conversion, and RGB to HSV conversion. Later, the long-short-term memory (LSTM) approach is used to classify WBC. Analyzing the experiment findings on the BCCD dataset, the proposed hybrid approach produced a 92.89% classification accuracy. Patil et al. (2021) proposed a deep learning method for the WBC classification by combining LSTM and CNN through canonical correlation analysis. The official correlation analysis enhances the accuracy of the input image by extracting multiple overlapping features, surpassing other comparable deep learning methods. The classification accuracy achieved for the applications on the BCCD dataset is 95.89%. Baghel et al. (2022) designed a new CNN model to enhance the precision of WBC classification. After 1000 epochs, their analysis of the BCC dataset resulted in a classification accuracy of 98.91%. Basnet et al. (2020) introduced a technique for precise WBC classification by employing the CNN structure. The researchers employed a dataset of 10,000 images categorized into five classes to assess their deep CNN classification accuracy, achieving an impressive 98.92% accuracy. Baydilli and Atila (2020) proposed a new method that utilizes capsule networks to categorize white blood cells into five distinct types. The researchers evaluated the precision of their capsule networks by conducting tests on the Leukemia Image Segmentation Challenge (LISC) dataset. This dataset consists of 263 blood cell images and is considered relatively small. Their efforts resulted in an accuracy rate of 96.86%. Ha et al. (2022) introduced a new semi-supervised model called Fine-grained Interactive Attention Learning (FIAL) for the classification of WBC. This model involves two essential elements; firstly, the Semi-Supervised Teacher-Student modules utilize a limited number of labeled WBC images to train a network called the “teacher”. The “teacher” network subsequently directs the learning process of a “student” network using a vast collection of unlabeled WBC samples, resulting in the generation of estimated probability vectors for these samples. Secondly, the Fine-Grained Interactive Attention mechanism enhances the attention within the network by highlighting the informative regions of the WBC images, leading to improved accuracy in classification. they achieved an overall accuracy of 93.2% when evaluating the FIAL model on the publicly available BCCD dataset. Hosseini et al. (2022) proposed a new CNN structure that is specifically tailored for the classification of four different types of WBCs: neutrophils, monocytes, eosinophils, and lymphocytes. Their approach stands out because they utilize a mix of random and grid search optimization algorithms to adjust the model's hyperparameters precisely. Following evaluation using the BCC dataset, their CNN achieved an accuracy rate of 97%. Vigueras-Guillén et al. (2021) focused on developing an innovative “parallel CapsNets” framework tailored specifically for the classification of WBCs. By utilizing a branching strategy, this architecture can isolate capsules that are utilized to identify distinct cell types. When evaluated on a 15-class dataset of acute myeloid leukemia images, their parallel CapsNets demonstrated superior performance to the baseline CNN, owing to enhanced stability and rotational invariance. Although the precise accuracy was not explicitly stated, its performance exceeded that of the foundational CNN. Various publications have employed CapsNets to execute various tasks using medical images. Many researchers utilized Sabour et al.’s network (2017) on small patches to carry out various tasks. These tasks include detecting diabetic retinopathy in fundus images, identifying mitosis in histology images (Hand E), and classifying breast cancer (Anupama et al., 2019; Iesmantas and Alzbutas, 2018) (Hoogi et al., 2019).

As seen in the literature, many studies have tried to improve CapsNet using different methods, often pre-training techniques. However, much must be done to strengthen the core CapsNet components, particularly the squash function. This paper will focus on improving the squash function, an important part of CapsNet.

2. Materials and Methods

This section presents two implementations: one employing the standard squash function and the other utilizing an enhanced squash to conduct a comparative analysis of the performance of these two employments within the CapsNet architecture, as shown in Figure 1.

2.1. Baseline capsnet theory

The concept of CapsNet was initially proposed by Sabour et al. (2017) . These networks represent images in a whole vector format, which allows them to encode internal properties, including the pose of entities within an image. In contrast to CNNs, which rely on pooling for output routing, CapsNet aims to preserve information to achieve equivariance, particularly in handling viewpoint changes. This preservation is facilitated through dynamic routing, which replaces the pooling mechanism. Lower-level capsules representing specific features are hierarchically routed to parent capsules to capture part-

whole relationships via linear transformations. This approach is based on the concept of inverse graphics, which suggests that the neural system deconstructs images into their inherent hierarchical properties. A capsule is a group of neurons that perform a diverse range of internal computations and subsequently encode the results of these computations into an n-dimensional vector. The vector is the output of the capsule. The length of this output vector indicates the vector's probability and direction, indicating certain properties about the entity. Using initial convolutional layers in capsule networks permits the reuse and replication of learned knowledge across different parts of the receptive field. An iterative Dynamic Routing algorithm is employed to determine the inputs to the capsules. The output of each capsule is then compared with the actual production of the higher-level capsules. In the case of a match between the outputs, the coupling coefficient between the two capsules is increased. Let i represent a lower-level capsule and j a higher-level capsule. The prediction vector is calculated as follows in equation 1 (Sabour et al., 2017).

$$\hat{u}(j|i) = \text{Wijui} \tag{1}$$

The Wij trainable weighting matrix and the ui output pose vector from the $(i-th)$ capsule to the $(j-th)$ capsule are employed in this context. The coupling coefficients are calculated using a SoftMax function as follows in equation 2 (Sabour et al., 2017).

$$Cij = \text{SoftMax}(bij) = \frac{\exp(bij)}{\sum_k \exp(bik)} \tag{2}$$

The log probability of capsule i coupled with capsule j, denoted by bij, is initialized with zero values. The total input to capsule j is a weighted sum of the prediction vectors, calculated as follows in equation 3 (Sabour et al., 2017):

$$sj = \sum_i \text{Cij}\ \hat{u}\, j|i \tag{3}$$

In capsule networks, the length of the output vector is employed to represent the probability for the capsule. Consequently, a non-linear activation function, the squashing function, is used. The squashing function is defined as follows in equation 4 (Sabour et al., 2017):

$$Vj = Squash(sj) = \frac{||\text{sj}||^2}{1+||\text{sj}||^2}\frac{sj}{||\text{sj}||} \tag{4}$$

The dynamic routing algorithm can update the cij values in each iteration. In this case, the objective is to optimize the Vj vector. In the dynamic routing algorithm, the bij vector is updated in every iteration according to equation 5 (Sabour et al., 2017):

$$bij = bij + Vj\ \hat{u}j|i \tag{5}$$

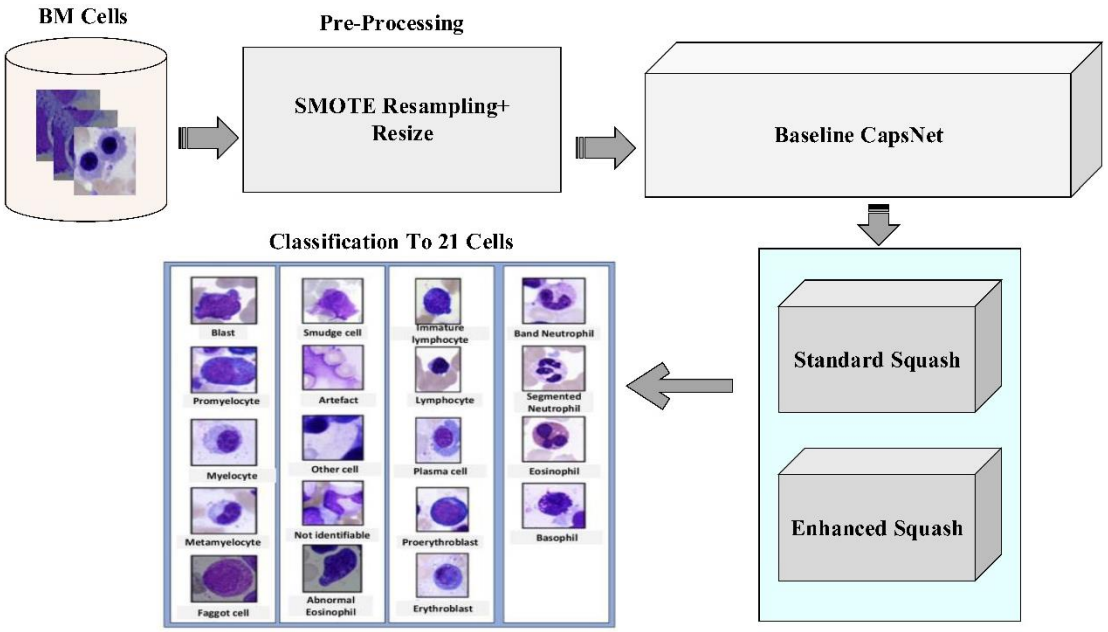

Figure 1. General steps for classification.

2.2. Dataset

The BM cell dataset (Matek et al., 2021) includes various hematological disease cell microscopic images consisting of more than 170000 de-identified, expert-annotated cells from bone marrow smears of 961 patients stained using the May-Grünwald-Giemsa/Pappenheim stain. The institutional review board at the Munich Leukemia Laboratory (MLL) has given its permission to use this dataset. Sample microscopic images from the dataset are shown in Figure 2.

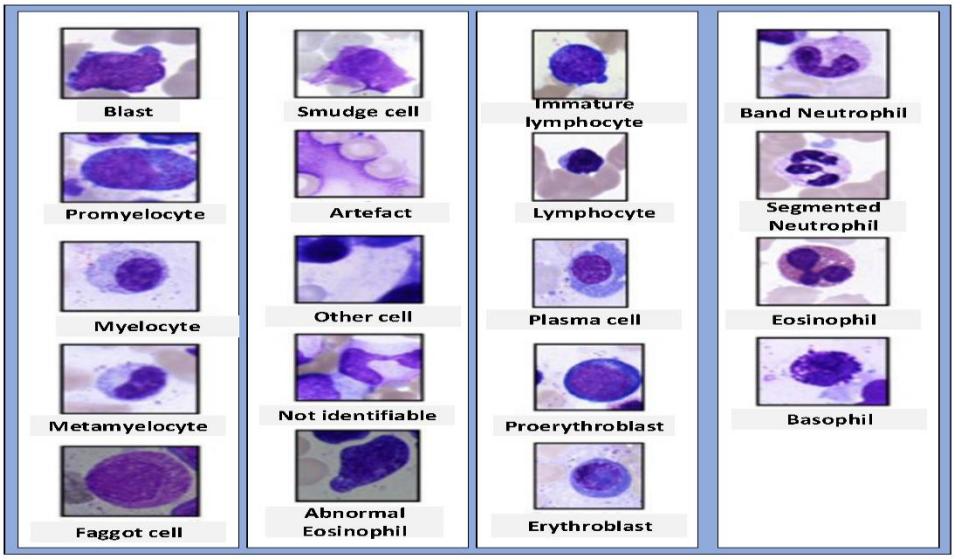

Figure 2. Morphological appearance of BM cell classes.

2.3. Pre-processing

As seen in Figure 3, imbalances in the distribution of some BM cell dataset classes cannot reach the expected target. SMOTE (Chawla et al., 2011), Adaptive Synthetic (ADASYN) (He et al., 2008), SMOTEBoost (Nitesh V. Chawla et al., 2003), and DataBoostIM (Guo and Viktor, 2004) sampling approach techniques are used to avoid these imbalances.

In this study, SMOTE, which is preferred over other sampling approximation techniques (Bajer et al., 2019; Maldonado et al., 2019) is used to create synthetic instances of the minority class by interpolating between class instances of an imbalanced dataset(Elreedy and Atiya, 2019) using equation 6 (Juanjuan et al., 2007; Reza and Ma, 2019). In this equation, t is the true minority value; k is the number of nearest neighbors of the true value; n is the nearest neighbor value $(n-t)$does Euclidean distance obtain the difference; $random(0,1)$Is the number value that allows the addition of different feature values. In this study, k=5 for each minority class.

$$\mathrm{smote(t,k)} = \mathrm{t} + \mathrm{random(0,1)} * (\mathrm{n} - \mathrm{t}) \quad (6)$$

Minority classes were rounded to 5000 images, and classes with over 5000 images were rounded up. This over-sampling technique improved overall classification performance and representation of the minority classes. Figure 4 shows the change in the number of data after the SMOTE process. In addition, all images are rescaled using Bicubic Interpolation (Keys, 1981). This technique is a non-adaptive interpolation algorithm and uses polynomial techniques to sharpen and enlarge digital images.

2.4. CapsNet Architecture with Standard Squash

The input images of the baseline CapsNet model were resized as 32 × 32 pixels to reduce the training time and then designed as shown in Figure 5.

The proposed model takes three arguments: the input image, the number of classes, and the number of routing iterations. After input images are preprocessed as in previous steps, fed to the convolutional layer to extract low-level features with filters is 256, kernel size is 9, and strides is 1. Then, these features are grouped into primary capsules with filters of 8 × 32, kernel of 9 × 9, and stride of 1 to reduce the model size, each representing a part or aspect of an object. Each capsule in the convolutional layer corresponds to a capsule in the primary capsule layer. A routing-by-agreement approach boosts learning capability and captures the relationships between different parts of an object. Each capsule in a higher-level layer sends its output to capsules in the layer above based on the agreement (compatibility) between their outputs. The routing process consists of iteratively updating the connection weights between capsules based on the match between their outputs. Thus, dynamic routing makes the capsules in higher layers focus on the most relevant capsules in the layer below; the ReLU activation function, a non-linear function in deep neural networks, is used to reduce dimension. The output of the capsule network is obtained by measuring the length of the output vectors of the capsules in the top layer. This length indicates the probability that a given class or object is present in the input image. Standard squash is applied to squash the vectors of the primary capsules as a non-linear activation function. It aims to ensure that short vectors are reduced to a length close to zero and

long vectors are reduced to a length close to 1. Then, adjusting the hyperparameter, the learning rate is initially determined to be 0.001, and the Adam (Kingma and Ba, 2015) optimization algorithm is used to change it dynamically. The model is run with a decay ratio of 0.9 and a routing of 3 for 100 epochs. The CapsNet model train accuracy is calculated at 96.99% on the BM dataset.

2.5. Squashing Vectors Improvement

Standard squash function proposed by (Sabour et al., 2017)represents a key component within the context of CapsNet. It refers to a non-linear activation function applied to the output of capsule layers to normalize the output vectors of capsules within the range [0, 1], ensuring consistent representation across different capsules. These layers process features and encode not only the presence of an entity but also its orientation. This normalization allows the magnitude of the output vector to be interpreted as the probability of the presence of an entity and its pose parameters. In addition, by scaling the vectors, the squash function prevents short vectors from vanishing and long vectors from becoming overly large. Additionally, there are still a few restrictions even with the succeeding standard squash function. It first has the vanishing gradient issue, which gets harder to ignore as vectors reach their maximum length. Training can proceed slowly or unstable because the squash function's gradient tends to get very small as the vectors reach 1. Squash functions aim to squash vectors into the range of 0 to 1. Moreover, very small norm vectors can be difficult for the traditional squash function to manage. Numerical instability or errors can arise when gradients must be computed precisely during training. Furthermore, the squash function has a sensitive behavior to initialization; a wrong initialization can cause problems like exploding, jeopardizing the training process's stability and convergence. Recording intricate relationships and patterns in the data is insufficient, to sum up. These drawbacks highlight the need to resolve these issues and improve the functionality of capsule networks in a range of applications by means of improvements or novel squash function formulations. This paper offers a better squash function that uses layer normalization to address these problems and offers more accurate and effective CapsNet training. Upon entering this normalization layer, the input vectors are scaled. Conversely, the commonly used squash function normalizes input vectors only by considering their squared norm. The normalization layer approach stabilizes the learning process by successively normalizing the activations of each capsule across the feature dimension. The computed mean and variance statistics are first applied by the improved squash function to normalize the input vectors. The following illustrates the mathematical procedures. The scaling factor is computed as (equation 7):

$$Scale = \frac{\alpha \left| \left| \frac{sj}{5} \right| \right|^2}{1 + \alpha \left| \left| \frac{sj}{5} \right| \right|^2} \quad (7)$$

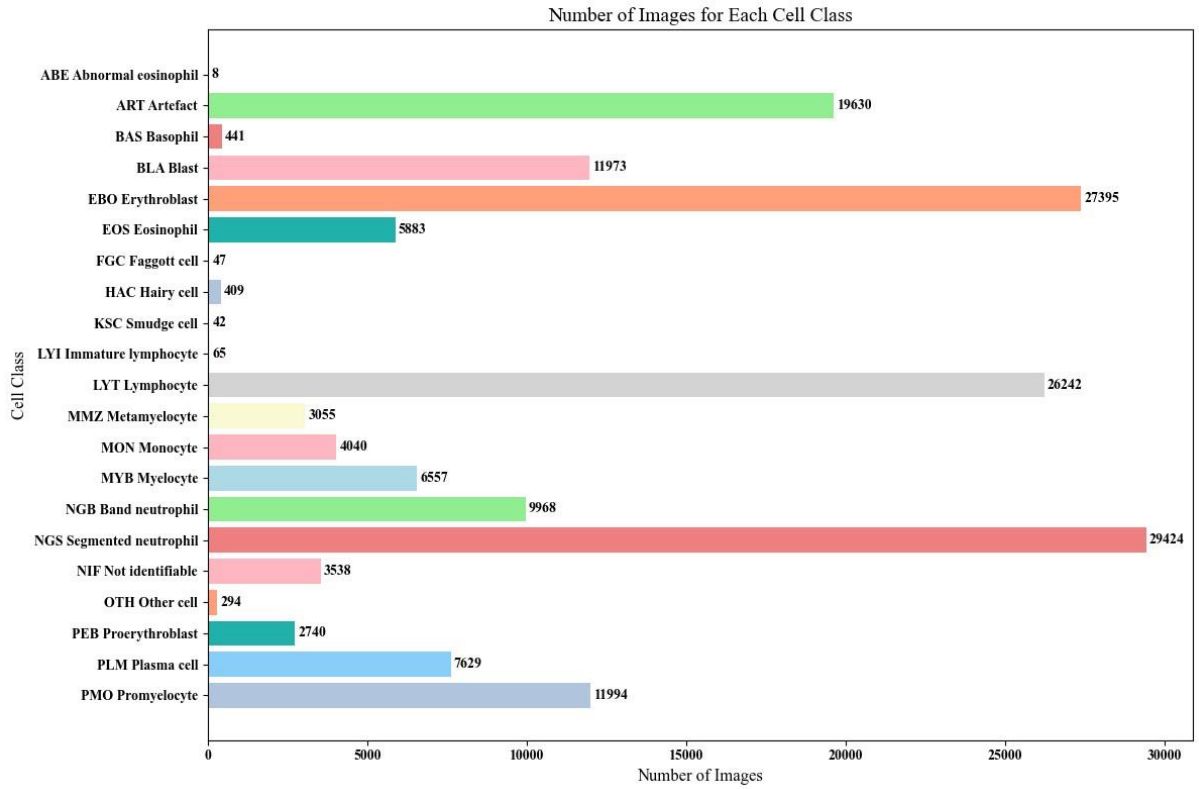

Figure 3. BM cell dataset features.

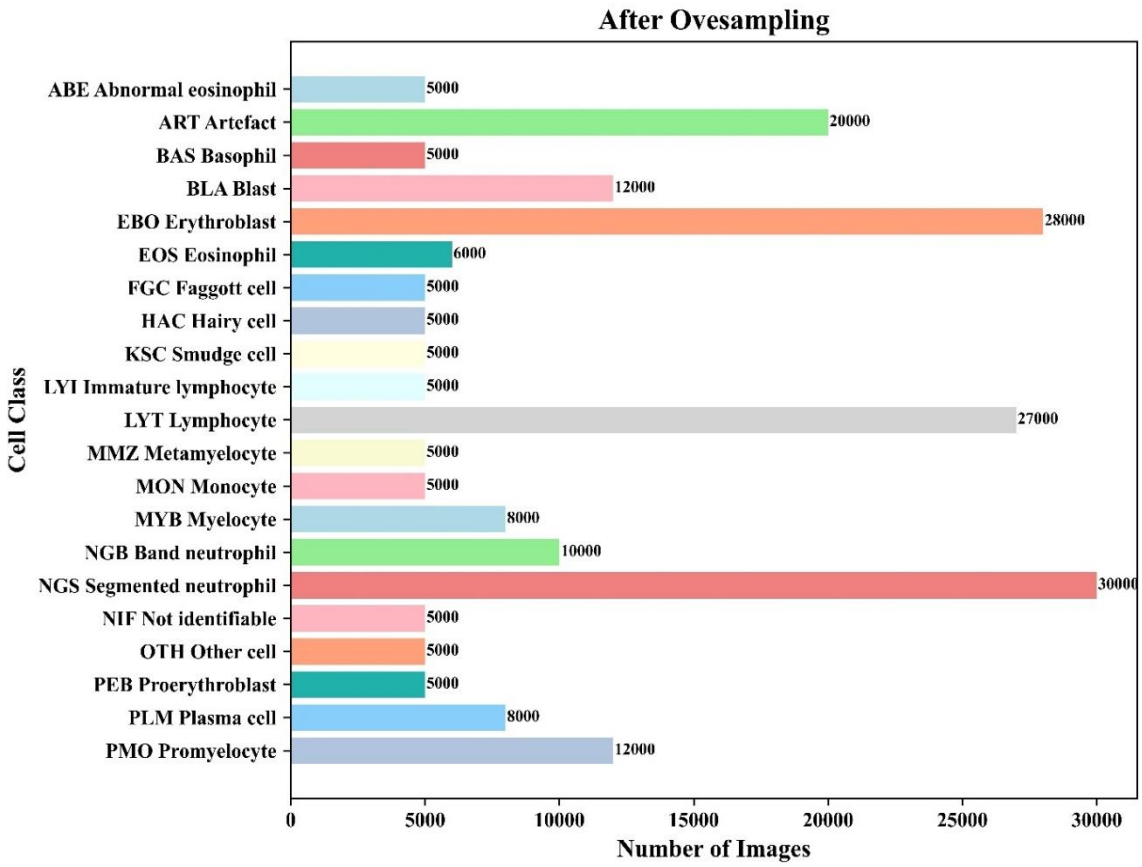

Figure 4. After Oversampling BM cell dataset features.

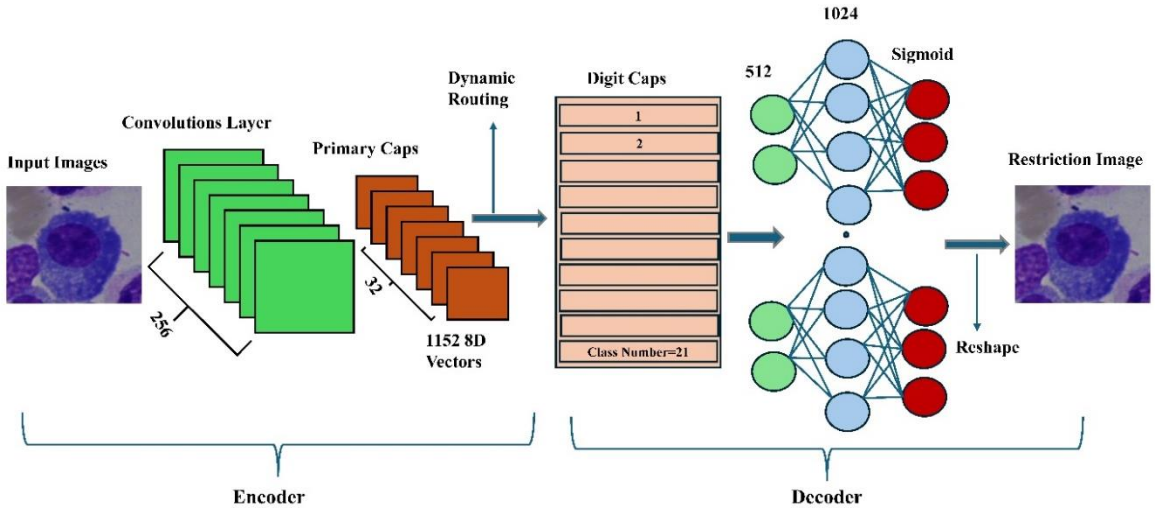

Figure 5. General structure of proposed CapsNet on BM dataset.

This equation defines the scale squash function; it transforms an input vector sj to extract informative features while discarding less informative ones. $\left|\left|\frac{sj}{5}\right|\right|^2$: This computes the input vector sj's squared L2 norm. The vector's length, or norm, equals its size. Input vectors can be stabilized by dividing them by 5 which will also improve optimization convergence and reduce issues like exploding or disappearing gradients. In addition, it indicates relative feature relationships over true values, so promoting scale invariance and strengthening the network against changes in feature magnitude. By focusing on features within a suitable range of magnitude, this normalization prevents the network from being unduly influenced by the values of extreme features. It allows it to extract meaningful patterns more successfully. Controlling the degree of the squashing is the scaling factor $\alpha > 0$, which is set to 0.5. Moreover, the model became more robust to noisy or unstable input data at α =0.5. The model demonstrated superior validation and tested dataset performance by capturing essential features and patterns. Thus, reduced bias,

improved gradients, and potentially more stable training often overcome the potential drawbacks. After calculating the scaling factor, the function performs layer normalization on the input vectors. It is a technique employed to normalize a layer's activations, ensuring that the mean is zero and the variance is one. This facilitates the stabilization of the learning process and accelerates convergence. The normalization layer is based on mean and variance instead of simply normalizing the input vectors based on their squared norm (as in the standard squash function); this step helps to capture more information about the distribution of the input vectors than just their squared norm. The equation for layer normalization is represented as (equation 8):

$$\text{vectors} = \frac{sj - \mu}{||\text{sj}||\sqrt{\sigma^2 + \varepsilon}} \quad (8)$$

First, layer normalization efficiently centers the input vectors around the zero mean by removing the mean (μ) from each one. This step can be used to lessen the variance of the input distribution to prevent too large or too small gradients during back-propagation. Every element of the input vectors has its distribution changed when the mean is subtracted from it. The effect of different mean values amongst samples is reduced by this method, which ensures that the transformed vectors mean close to 0. By focusing and scaling the activations around zero mean and unit variance, respectively, this normalization method improves the convergence and stability of the training process.

By normalizing activations within each layer, the improved squash function improves gradient flow by lowering variability across the feature dimension, and it additionally decreases the network's sensitivity to the layer addition of parameters. Through the enhancement of capsule networks' expressiveness and robustness, the squash function enables them to capture complex patterns and relationships more successfully in the data, which may lead to better performance on tasks like pose estimation, object detection, and image classification. The final enhanced equation is represented as (equation 9):

$$\text{Enhanced squash} = \text{scale} * \text{vector} \quad (9)$$

$$= \frac{\alpha \left|\left|\frac{\text{sj}}{5}\right|\right|^2}{1 + \alpha \left|\left|\frac{\text{sj}}{5}\right|\right|^2} \frac{sj - \mu}{||\text{sj}||\sqrt{\sigma^2 + \varepsilon}}$$

3. Results and Discussion

The experiment results in this work were obtained using an NVIDIA RTX 3070 GPU with 64 GB of VRAM. Our focus was on creating models specifically made to manage the difficulties of handling huge datasets and highly complex features. Our method focused mainly on adding a better squash function to the CapsNet core. On three more datasets MNIST, Fashion MNIST, and CIFAR-10, we carefully evaluated the performance of our proposed models. These tests showed how flexible our models could be and how they might be used for purposes other than the BM dataset that was first used.

Using extensive publicly available datasets, this section examines our improved models for BM cell classification. Utilizing over-sampling techniques, the SMOTE method was used to address the class imbalance in the BM dataset successfully. During training, the SMOTE application significantly improved our models' overall accuracy. Rebalancing the distribution of the data, SMOTE over-sampling methods guaranteed a strong evaluation and validation process. BM images totaling 216,000 were produced after SMOTE was applied to the original 171,374-image BM dataset. The dataset was next split into three subsets, as shown in Table 1, 70% for training, 10% for validation, and 20% for testing. Moreover, we investigated three more datasets: the MNIST, Fashion MNIST, and CIFAR-10 datasets, each of which presented important difficulties and revealed the effectiveness of our model in different domains. The handwritten numbers in the MNIST dataset are used as a standard for image categorization tasks. A further feature of the Fashion MNIST dataset is grayscale photos of accessories and clothing. With the more difficult classification task provided by the Fashion MNIST dataset than the MNIST dataset, our models can distinguish between minute details and minute variations in fashion items. Ten classification classes with a wider range of object classes including animals, cars, and commonplace objects represented in colored photos make up the CIFAR-10 dataset. The complexity and variation of CIFAR-10 present difficulties. Table 2 provides details of these three data sets.

Table 1. Splitting BM dataset

Total BM cells	After applying SMOTE	Training data (70%)	Validation (10%)	Test data (20%)
171374	216000	151200	21600	43200

Table 2. Details of the dataset

Dataset	Image Size	Channels	Classes	Train Set	Test Set
MNIST	(28, 28)	1	10	60,000	10,000
Fashion-MNIST	(28, 28)	1	10	60,000	10,000
CIFAR10	(32, 32)	3	10	50,000	10,000

3.1. Evaluation Process of Models

We evaluated the performance of our models using an array of metrics to get an extensive understanding of their effectiveness. F1-score was one of the metrics applied, together with specificity, recall, and precision. Much of evaluating the model's ability to accurately reduce false positives while identifying true positive instances was precision, which indicates the accuracy of positive predictions. Conversely, recall gauged how sensitively the model caught each positive occurrence in the data. Conversely, specificity assessed how well a model might recognize instances of negativity. Furthermore, the general performance of the model was fairly evaluated by the harmonic mean of recall and precision, or F1-score. We were driven to enhance and optimize the performance of our models for different applications by our thorough knowledge of their benefits and drawbacks, which we attained by combining these metrics. Equations (10, 11, 12 and 13) are applied as follows (Bharathi, 2024):

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

$$Specificty = \frac{TN}{TN + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

$$F1 - score = \frac{2 * (Precision * Recall)}{Precision + Recall} \tag{13}$$

4. Discussion

The CapsNet model's outcomes using the conventional squash function are shown in Table 3 for a variety of classes in the BM dataset. It reports the precision, recall, specificity, and F1-score of the model, which demonstrates its correct classification of various cell types. The model is robust in identifying positive and negative instances, demonstrated by the high precision and recall it shows across most classes.

It is shown in Table 4 and Figure 6 that combining the suggested CapsNet architecture with an "enhanced squash" function works well. More precisely, the improved squash function outperforms CapsNet versions using standard squash functions from previous studies with an excellent classification accuracy of 98.52% on the BM dataset. This remarkable development highlights how important the improved squash function is to the model's capacity to identify and classify intricate patterns in the dataset. To optimize the performance of CapsNet, we have also methodically included several improved squash functions into our model architecture and carefully included enhancements recommended by previous studies. Through conducting extensive testing and comparison of the BM dataset, we evaluated the efficacy of each of these variations. Our results clearly demonstrate the superiority of our model over alternative approaches, confirming its ability to identify the most efficient squash function for capsule networks.

Table 3. Performance metrics of the CapsNet model using standard squash

Class Name	Precision (%)	Recall (%)	Specificity (%)	F1-score (%)
Abnormal eosinophil (ABE)	100	100	100	100
Artefact (ART)	100	100	100	100
Basophil (BAS)	100	100	100	100
Blast (BLA)	99.92	100	100	99.96
Erythroblast (EBO)	99.89	99.96	99.98	99.93
Eosinophil (EOS)	100	99.5	100	99.75
Faggott cell (FGC)	100	100	100	100
Hairy cell (HAC)	100	100	100	100
Smudge cell (KSC)	100	100	100	100
Immature lymphocyte (LYI)	100	100	100	100
Lymphocyte (LYT)	99.17	99.59	99.88	99.38
Metamyelocyte (MMZ)	95.03	89.8	99.89	92.34
Monocyte (MON)	89.01	89.1	99.74	89.06
Myelocyte (MYB)	92.57	93.37	99.71	92.97
Band neutrophil (NGB)	91.74	85	99.63	88.24
Segmented neutrophil (NGS)	94.19	97.47	99.03	95.8
Not identifiable (NIF)	91.86	86.9	99.82	89.31
Other cell (OTH)	97.35	99.1	99.94	98.22
Proerythroblast (PEB)	95.35	92.3	99.89	93.8
Plasma cell (PLM)	93.28	86.75	99.76	89.9
Promyelocyte (PMO)	91.83	97	99.49	94.35

Table 4. Comparison of our proposed with the previous optimized squash functions for the CapsNet on the BM dataset

References	Year	Equations	Accuracy %	Number of epochs
(Sabour et al., 2017)	2017	$Squash(sj) = \frac{\|\|sj\|\|^2}{1+\|\|sj\|\|^2}\frac{sj}{\|\|sj\|\|}$	96.99	100
(Xi et al., 2017)	2017	$F(x) = (1 - \frac{1}{\exp(\|\|x\|\|)})\frac{x}{\|\|x\|\|}$	91	100
(Afriyie et al., 2022a)	2022	$Squash(sj) = \frac{\|\|sj\|\|^2}{1+\|\|sj\|\|^2}\frac{sj}{2\|\|sj\|\|^2}$	94.05	100
Proposed (Enhanced squash)	2024	$Squash(sj) = \frac{\alpha \left\|\frac{sj}{5}\right\|^2}{1+\alpha \left\|\frac{sj}{5}\right\|^2}\frac{sj-\mu}{\|\|sj\|\|\sqrt{\sigma^2+\varepsilon}}, \alpha = 0.5$	98.52	100

Figure 6. Accuracy comparison of standard vs enhanced squash functions on BM dataset.

A comprehensive comparison between the baseline CapsNet, which employs conventional squash functions, and our enhanced model is shown in Table 5. Multiple datasets are utilized for the comparison, such as MNIST, Fashion-MNIST, and CIFAR-10. The results show that, on various image classification tasks, the improved squash function significantly enhances CapsNet's performance. On the MNIST dataset, our proposed model achieved 99.83% accuracy compared to the baseline CapsNet accuracy of 99.23%. This shows that the improved squash function successfully optimized CapsNet's performance by a significant 0.5% percent on this commonly used dataset for handwritten digit classification. With a 94.66% accuracy on the Fashion-MNIST dataset, our improved model outperformed the foundational CapsNet's 92.49% accuracy. The significant accuracy increase (2.17%) observed is strong evidence that the squash function enhancement improves CapsNet's performance on tasks involving fashion item categorization. Moreover, the performance of our suggested model was much enhanced when it was applied on the CIFAR-10 dataset, which is a very difficult classification problem because of its large number of real-life images. At 73%, the CapsNet model we improved outperformed the baseline CapsNet's accuracy (67.81% by 5.19%). The substantial accuracy increase shows the robustness and adaptability of the improved squash function, which allowed CapsNet to classify complicated images correctly in various datasets. The results demonstrate the flexibility and promise of the enhanced squash function to support CapsNet's performance in a range of image categorization tasks. We show that our model is flexible and capable of handling different data types and challenges by routinely outperforming the original CapsNet on a variety of datasets. The results show how our improved CapsNet architecture can be useful in practical situations requiring precise and efficient image categorization.

Table 5. Comparison baseline CapsNet and enhanced our model on different datasets

Dataset	Standard Squash (%)	Enhanced Squash (%)
MNIST (Sabour et al., 2017)	99.23	99.83
Fashion MNIST (Afriyie et al., 2022b)	92.49	94.66
Cifar10 (El Alaoui and Gadi, 2021)	67.81	73

Figure 7, Figure 8 and Figure 9 show how the enhanced squash function on the training data improved over 100 epochs. This shows how the model learns and how its accuracy improves.

Figure 10 shows a confusion matrix of the model's performance for Bone Marrow, Fashion MNIST, and CIFAR-10 dataset. The actual and predicted classes of a classification model are shown in a confusion matrix. Each row of the matrix represents actual class instances, while each column represents predicted class instances. The matrix cells show the number of true and false positives. Analyzing the confusion matrix allows us to evaluate the model's accuracy and detect misclassification patterns across classes in each dataset as shown Table 6.

4.1. Comparison of Our Models with State-of-The-Art Methods on Same Dataset

Table 7 briefly compares different methods on the BM dataset the studies on this BM dataset are limited because is new and huge unbalanced data, but in our proposed models these issues have been solved, showing the performance metrics achieved by different approaches. It shows that our proposed methods outperform existing state-of-the-art methods in terms of accuracy. These results highlight the effectiveness and superiority of our proposed methods for biomedical image classification tasks on the BM dataset.

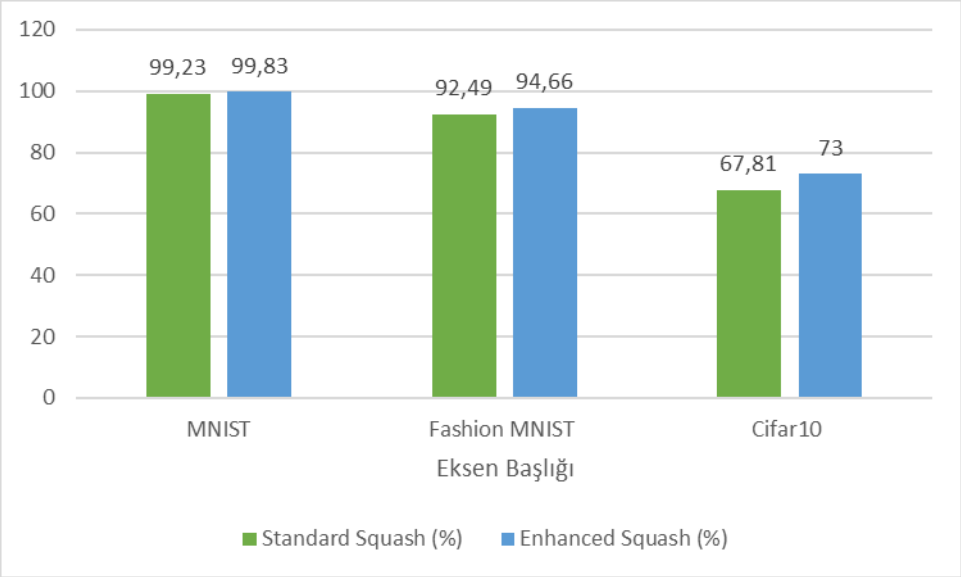

Figure 7. Accuracy comparison of standard vs enhanced squash functions on different datasets

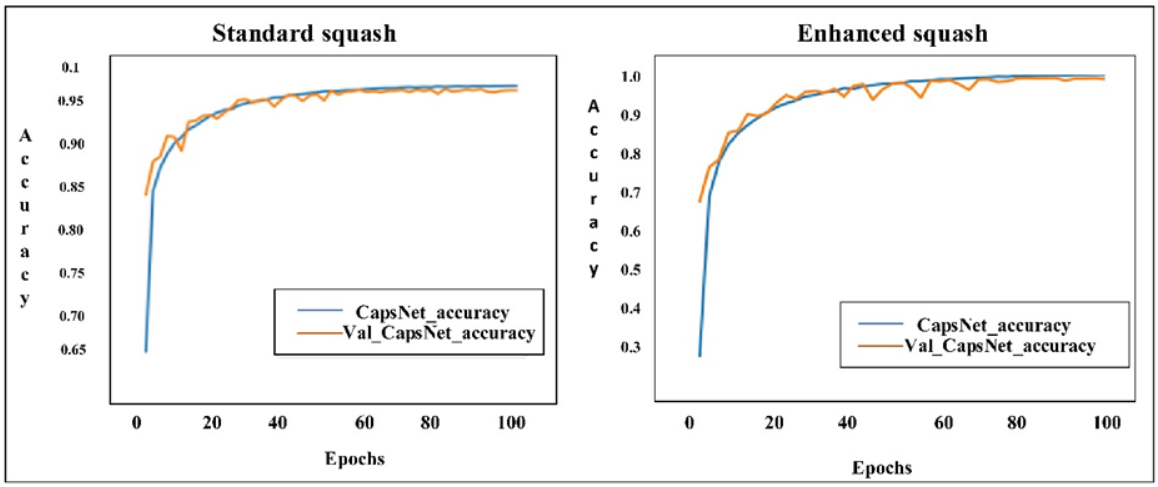

Figure 8. Accuracy curve comparison: standard vs enhanced squash functions on BM dataset

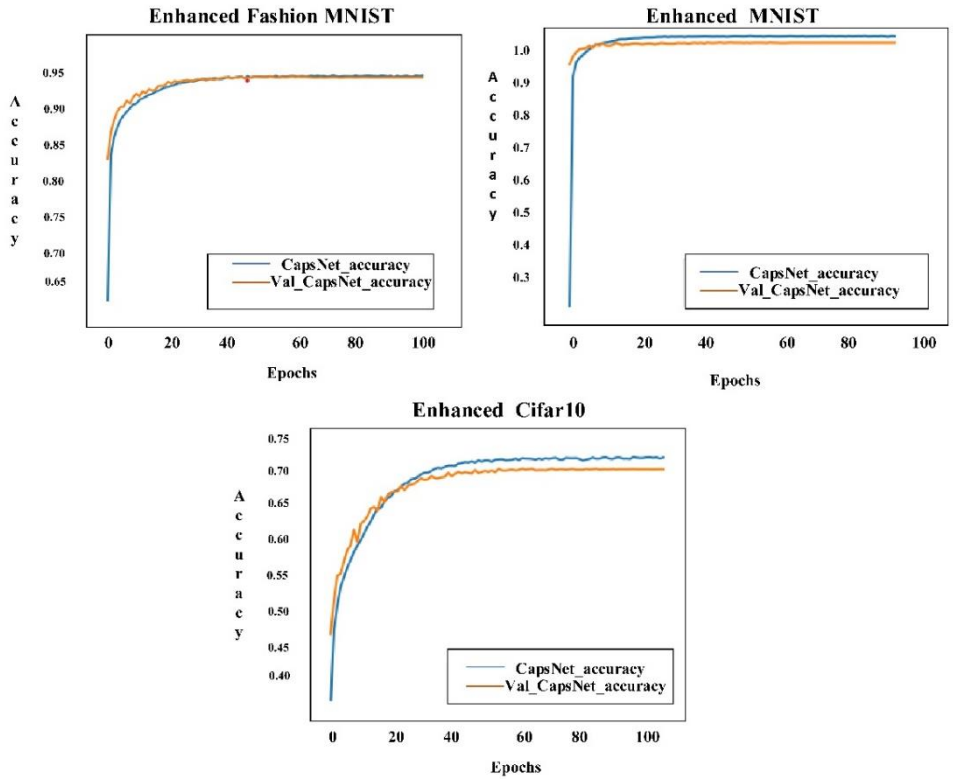

Figure 9. Accuracy comparison: standard vs enhanced squash functions on a different dataset.

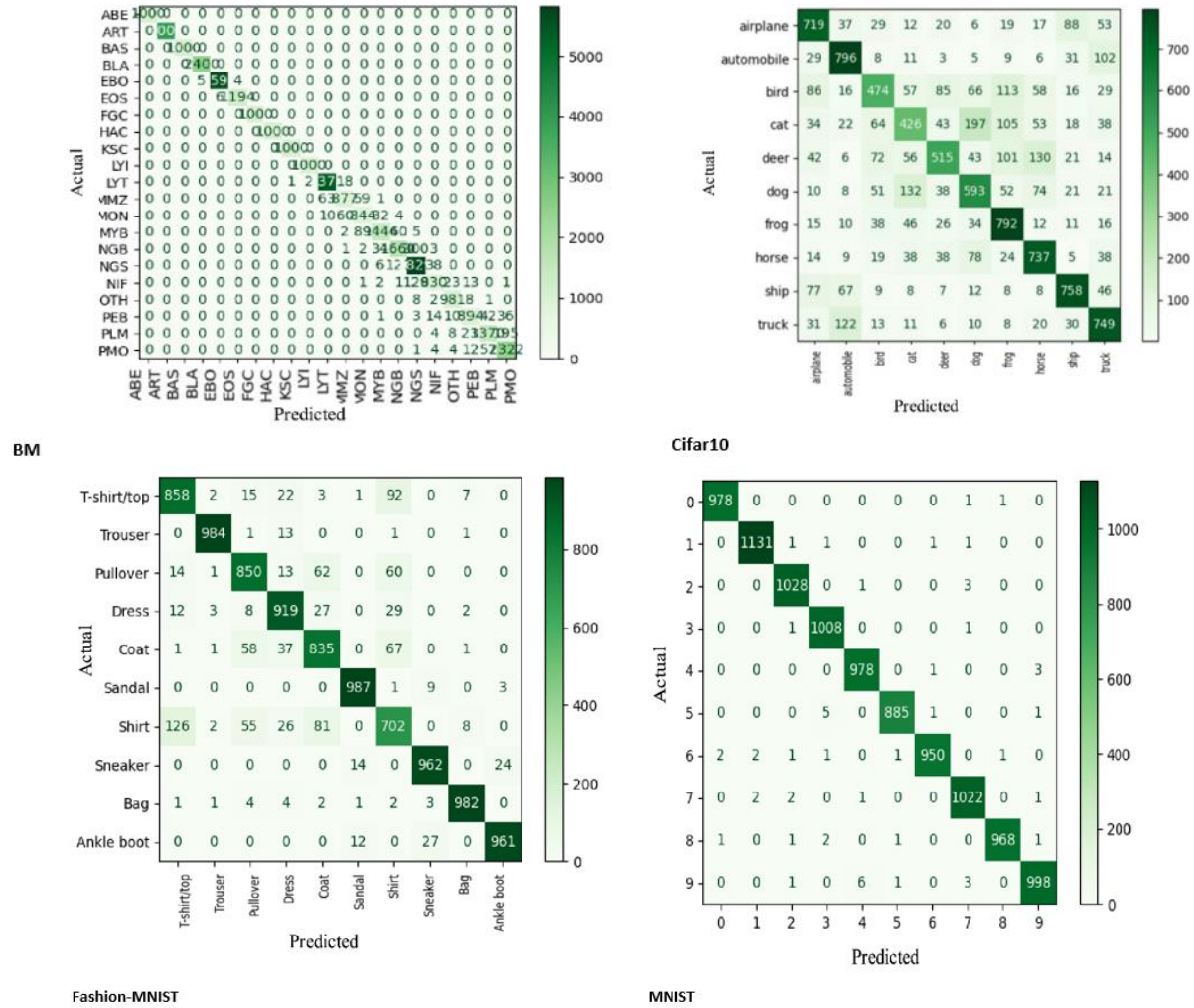

Figure 10. Confusion matrix analysis across four types of datasets

Table 6. Performance Comparison of Various CapsNet Architectures on MNIST, Fashion MNIST, CIFAR-10, and BM Datasets.

Methods	MNIST (%)	Fashion MNIST (%)	Cifar10(%)	BM (%)
ShallowNet (Mensah et al., 2021)	-	92.70	75.75	-
CapsNet (Sabour et al., 2017)		90.72	62.91	-
64 Capsule Layers (Xi et al., 2017)	68.93	-	64.67	-
Multi-lane (Chang and Liu, 2020)	99.73	92.63	76.79	-
MS-CapsNet (Xiang et al., 2018)		92.70	75.70	-
ResCapsNet (Goswami, 2019)		-	78.54	-
CFC-CapsNet (Shiri and Baniasadi, 2021)	-	92.86	73.15	-
Fast Inference (Zhao et al., 2019b)	99.43	91.52	70.33	-
Max–min (Zhao et al., 2019a)	99.55	92.07	75.92	-
MLSCN (Chang and Liu, 2020)	99.73	-	76.79	-
MLCN (do Rosario et al., 2021)	-	92.63	75.18	-
Quick-CapsNet (QCN) (Shiri et al., 2020)	99.28	88.84	67.18	-
(Afriyie et al., 2022a)	-	92.80	75.42	-
Proposed Model Enhanced Squash CapsNet	99.83	94.66	73	98.52

Table 7. Performance comparison of our proposed method with the existing state-of-the-art on the same BM dataset

Reference	Year	Method	Performance Metric	Result Rate
(Ananthakrishnan et al., 2022)	2022	CNN + SVM , CNN + XGB.	Accuracy	32%,28%
		Siamese neural	Accuracy	91%
		Supervised method (ResNeXt-50)	(Avg) F1-Score	62 %
			Precision	68.24%
			Recall	72.92%
(Fazeli et al., 2022)	2022	Self-supervised method (SW AV)	(Avg) F1-Score	73%
			Precision	78%
			Recall	70%
		Supervised Contrastive	(Avg) F1-Score	69 %
			Precision	75%
			Recall	66%
Our proposed Methods	2024	Baseline CapsNet	Accuracy	96.99 %
		Enhanced Squash CapsNet	Accuracy	98.52 %

4. Conclusion

Our research demonstrates that a CapsNet architecture with a novel function designated “enhanced squash” is highly efficacious. This yielded superior outcomes compared to preceding CapsNet methodologies on the selected BM dataset. This evidence substantiates the pivotal role of the Enhanced Squash function in enabling the model to effectively learn and classify intricate patterns within the data. The enhanced squash function is also effective in other datasets. The enhanced squash function facilitated superior performance of the model on datasets such as MNIST, CIFAR-10 and Fashion-MNIST, demonstrating its versatility in handling diverse image types. The enhanced squash function handles complex data well. It improves the accuracy of CapsNet models in real-world applications. Implementing our enhanced squash function improved the accuracy of bone marrow cell classification. Our experiments have shown that the Enhanced squash feature improves accuracy on various datasets, including MNIST, CIFAR-10 and Fashion-MNIST. The model achieved 99.83% accuracy on the MNIST dataset, 73% on the CIFAR-10 dataset, and 94.66% on the Fashion MNIST dataset.

In the future, we can look at the math behind the enhanced margin loss function in more detail. This will help us understand how it affects the model and improves classification accuracy. We can also compare CapsNet with other image classification architectures to see how the enhanced squash function affects its performance and whether it can be used for other image classification tasks.

Author Contributions

The percentages of the author contributions are presented below. The author reviewed and approved the final version of the manuscript.

	A.F.A.A.R.	N.A.A
C	50	50
D	60	40
S		100
DCP	100	
DAI	100	
L	100	
W	50	50
CR	50	50
SR		100
PM		100

C=Concept, D= design, S= supervision, DCP= data collection and/or processing, DAI= data analysis and/or interpretation, L= literature search, W= writing, CR= critical review, SR= submission and revision, PM= project management.

Conflict of Interest

The author declared that there is no conflict of interest.

Ethical Consideration

Ethics committee approval was not required for this study because of there was no study on animals or humans.

References

Afriyie Y, Weyori BA, Opoku AA. 2022a. Classification of blood cells using optimized capsule networks. Neural Process Lett. 54: 4809–482.

Afriyie Y, Weyori BA, Opoku AA. 2022b. Comparative evaluation performances of capsule networks for complex image classification. J. Data Inf. Manag, 4(3–4): 267-276.

Agustin RI, Arif A, Sukorini U. 2021. Classification of immature white blood cells in acute lymphoblastic leukemia L1 using neural networks particle swarm optimization. Neural Comput Appl, 33(17): 10869-10880. doi: 10.1007/S00521-021-06245-7/TABLES/5.

Ananthakrishnan B, Shaik A, Akhouri S, Garg P, Gadag V, Kavitha MS. 2022. Automated bone marrow cell classification for haematological disease diagnosis using siamese neural network. Diagnostics (Basel), 13(1): 3390. doi: 10.3390/DIAGNOSTICS13010112.

Anupama MA, Sowmya V, Soman KP. 2019. Breast cancer classification using capsule network with preprocessed histology images. In: Proceedings of the IEEE International Conference on Communication and Signal Processing, ICCSP, April 4-6, Melmaruvathur, India, pp: 143-147.

Aydın Atasoy N, Al Rahhawi AFA. 2024. Examining the classification performance of pre-trained capsule networks on imbalanced bone marrow cell dataset. Int J Imaging Syst Technol, 34(3): e23067. doi: 10.1002/IMA.23067.

Balasubramanian K, Ananthamoorthy NP, Ramya K. 2022. An approach to classify white blood cells using convolutional neural network optimized by particle swarm optimization algorithm. Neural Comput Appl, 34(18): 16089-16101. doi: 10.1007/S00521-022-07279-1/TABLES/13.

Baghel N, Verma U, Nagwanshi KK. 2022. WBCs-Net: type identification of white blood cells using convolutional neural network. Multimed Tools Appl, 81(29): 42131-42147. doi: 10.1007/S11042-021-11449-Z/TABLES/11.

Bajer D, Zonc B, Dudjak M, Martinovic G. 2019. Performance analysis of SMOTE-based oversampling techniques when dealing with data imbalance. In: Proceedings of the International Conference on Systems, Signals, and Image Processing, June 5-7, Osijek, Croatia, pp:265-271.

Basnet J, Alsadoon A, Prasad PWC, Al Aloussi S, Alsadoon OH. 2020. A novel solution of using deep learning for white blood cells classification: enhanced loss function with regularization and weighted loss (ELFRWL). Neural Process Lett, 52(2): 1517-1553. doi: 10.1007/S11063-020-10321-9.

Baydilli YY, Atila Ü. 2020. Classification of white blood cells using capsule networks. Comput Med Imaging Graph, 80: 101699. doi:10.1016/J.COMPMEDIMAG.2020.101699.

Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. 2011. SMOTE: synthetic minority over-sampling technique. J Artif Intell Res, 16: 321-357.

Chawla NV, Lazarevic A, Hall LO, Bowyer KW. 2003. SMOTEBoost: improving prediction of the minority class in boosting. In: Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, September 22-26, Cavtat-Dubrovnik, Croatia, pp: 107–119.

Chang S, Liu J. 2020. Multi-lane capsule network for classifying images with complex background. IEEE Access, 8: 79876-79886.

Confusion Matrix for Multi-Class Classification 2024. URL: https://www.analyticsvidhya.com/blog/2021/06/confusion-matrix-for-multi-class-classification/ (accessed date: June 13, 2024).

Çınar A, Tuncer SA. 2021. Classification of lymphocytes, monocytes, eosinophils, and neutrophils on white blood cells using hybrid Alexnet-GoogleNet-SVM. SN Appl Sci, 3(4): 1-11. doi: 10.1007/S42452-021-04485-9/TABLES/4.

Dayı B, Üzen H, Çiçek İB, Duman ŞB. 2023. A novel deep learning-based approach for segmentation of different type caries lesions on panoramic radiographs. Diagnostics, 13(2): 202. doi: 10.3390/DIAGNOSTICS13020202.

Dhal KG, Rai R, Das A, Ray S, Ghosal D, Kanjilal R. 2023. Chaotic fitness-dependent quasi-reflected Aquila optimizer for superpixel based white blood cell segmentation. Neural Comput Appl, 35(21): 15315-15332. doi: 10.1007/S00521-023-08486-0/TABLES/10.

do Rosario VM, Breternitz M, Borin E. 2021. Efficiency and scalability of multi-lane capsule networks (MLCN). J Parallel Distrib Comput, 155: 63-73. doi: 10.1016/J.JPDC.2021.04.010.

El Alaoui-Elfels O, Gadi T. 2021. EMG-CapsNet: Elu multiplication gate capsule network for complex images classification. In: Proceedings of the 13th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2021), Online, December 15 – 17, pp: 97–108.

Elreedy D, Atiya AF. 2019. A comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance. Inf Sci (N Y), 505: 32-64.

Fazeli S, Samiei A, Lee TD, Sarrafzadeh M. 2022. Beyond labels: visual representations for bone marrow cell morphology recognition, ArXiv, 111-117. doi: 10.1109/ichi57859.2023.00025.

Gautam A, Singh P, Raman B, Bhadauria H. 2017. Automatic classification of leukocytes using morphological features and Naïve Bayes classifier. In: Proceedings of the IEEE Region 10 Annual International Conference, TENCON, November 5-8, Penang, Malaysia, pp: 1023-1027. doi: 10.1109/TENCON.2016.7848161.

Girdhar A, Kapur H, Kumar V. 2022. Classification of white blood cells using convolution neural network. Biomed Signal Process Control, 71: 103156. doi: 10.1016/J.BSPC.2021.103156.

Ghosh M, Das D, Chakraborty C, Ray AK. 2010. Automated leukocyte recognition using fuzzy divergence. Micron, 41(7): 840-846. doi: 10.1016/J.MICRON.2010.04.017.

Ghosh A, Jana ND, Mallik S, Zhao Z. 2022. Designing optimal convolutional neural network architecture using differential evolution algorithm. Patterns, 3(9):100567. doi: 10.1016/j.patter.2022.100567.

Guo H, Viktor HL. 2004. Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach. SIGKDD Explor, 6(1): 30-39.

Goswami D. 2019. Application of capsule networks for image classification on complex datasets.URL: https://hdl.handle.net/2142/105694 (accessed: May 26, 2024).

Ha Y, Du Z, Tian J. 2022. Fine-grained interactive attention learning for semi-supervised white blood cell classification. Biomed Signal Process Control, 75: 103611. doi: 10.1016/J.BSPC.2022.103611.

He H, Bai Y, Garcia EA, Li S. 2008. ADASYN: adaptive synthetic sampling approach for imbalanced learning. In: Proceedings of the International Joint Conference on Neural Networks, June 1-8, Hong Kong, China, pp:1322-1328.

Hegde RB, Prasad K, Hebbar H, Singh BMK. 2019a. Comparison of traditional image processing and deep learning approaches for classification of white blood cells in peripheral blood smear images. Biocybern Biomed Eng, 39(2): 382-392. doi: 10.1016/J.BBE.2019.01.005.

Hegde RB, Prasad K, Hebbar H, Singh BMK. 2019b. Feature extraction using traditional image processing and convolutional neural network methods to classify white blood cells: a study. Australas Phys Eng Sci Med, 42(2): 627-638. doi: 10.1007/S13246-019-00742-9/FIGURES/11.

Hosseini M, Bani-Hani D, Lam SS. 2022. Leukocytes image classification using optimized convolutional neural networks. Expert Syst Appl, 205: 117672. doi: 10.1016/J.ESWA.2022.117672.

Hoogi A, Wilcox B, Gupta Y, Rubin DL. 2019. Self-attention capsule networks for object classification. URL: https://arxiv.org/abs/1904.12483v2 (accessed: May 03, 2024).

Juanjuan W, Mantao X, Hui W, Jiwu Z. 2007. Classification of imbalanced data by using the SMOTE algorithm and locally linear embedding. In: Proceedings of the International Conference on Signal Processing, ICSP, November 16-20, Guilin, China, pp: 1741- 1745.

Keys RG. 1981. Cubic convolution interpolation for digital image processing. IEEE Trans Acoust, 29(6): 1153-1160.

Kingma DP, Ba JL. 2015. Adam: a method for stochastic optimization. In: Proceedings of 3rd International Conference on Learning Representations, ICLR, May 7-9, San Diego, pp:13.

Kutlu H, Avci E, Özyurt F. 2020. White blood cells detection and classification based on regional convolutional neural networks. Med Hypotheses, 135: 109472. doi: 10.1016/J.MEHY.2019.109472.

LeCun Y, Bengio Y, Hinton G. 2015. Deep learning. Nature, 521(7553): 436-444. doi: 10.1038/nature14539.

Liu Y, Fu Y, Chen P. 2019. WBCaps: a capsule architecture-based classification model designed for white blood cells identification. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, July 23-27, Berlin, Germany, pp:7027-7030. doi: 10.1109/EMBC.2019.8856700.

Long F, Peng JJ, Song W, Xia X, Sang J. 2021. BloodCaps: a capsule network-based model for the multiclassification of human peripheral blood cells. Comput Methods Programs Biomed, 202: 105972. doi: 10.1016/J.CMPB.2021.105972.

Iesmantas T, Alzbutas R. 2018. Convolutional capsule network for classification of breast cancer histology images. In: Proceedings of the 15th International Conference, ICIAR 2018, June 27–29, Póvoa de Varzim, Portugal, pp 869–876 DOI: 10.1007/978-3-319-93000-8_85

Maldonado S, López J, Vairetti C. 2019. An alternative SMOTE oversampling strategy for high-dimensional datasets. Appl Soft Comput, 76: 380-389.

Matek C, Krappe S, Münzenmayer C, Haferlach T, Marr C. 2021. Highly accurate differentiation of bone marrow cell morphologies using deep neural networks on a large image data set. Blood, 138(20): 1917-1927. DOI: 10.1182/blood.2020010568

Mensah PK, Weyori BA, Ayidzoe MA. 2021. Evaluating shallow capsule networks on complex images. Int J Inf Technol (Singapore), 13(3): 1047-1057. doi: 10.1007/S41870-021-00694-Y/FIGURES/6.

Mirmohammadi P, Ameri M, Shalbaf A. 2021. Recognition of acute lymphoblastic leukemia and lymphocytes cell subtypes in microscopic images using random forest classifier. Phys Eng Sci Med, 44(2): 433-441. doi: 10.1007/S13246-021-00993-5/TABLES/2.

Mohamed M, Far B, Guaily A. 2012. An efficient technique for white blood cells nuclei automatic segmentation. In: Proceeding of IEEE International Conference on Systems, Man, and Cybernetics (SMC), October 14-17, Seoul, Korea (South), pp: 220-225. doi: 10.1109/ICSMC.2012.6377703.

Muhammad A, Arserim M, Ömer T. 2023. Compare the classification performances of convolutional neural networks and capsule networks on the Coswara dataset. DUJE, 14(2): 265-271. doi: 10.24012/dumf.1270429.

Nair P, Doshi R, Keselj S. 2021. Pushing the limits of capsule networks. URL: http://arxiv.org/abs/2103.08074 (accessed: May 03, 2024).

Patil AM, Patil MD, Birajdar GK. 2021. White blood cells image classification using deep learning with canonical correlation analysis. IRBM, 42(5): 378-389. doi: 10.1016/J.IRBM.2020.08.005.

Patrick MK, Adekoya AF, Mighty AA, Edward BY. 2022. Capsule networks – a survey. J King Saud Univ Comput Inf Sci, 34(1): 1295-1310. doi: 10.1016/J.JKSUCI.2019.09.014.

Ren H, Su J, Lu H. 2019. Evaluating generalization ability of convolutional neural networks and capsule networks for image classification via Top-2 classification. URL: https://arxiv.org/abs/1901.10112v4 (accessed: May 10, 2024).

Reza MS, Ma J. 2019. Imbalanced histopathological breast cancer image classification with convolutional neural network. In: Proceedings of the 14th IEEE International Conference on Signal Processing (ICSP), August 12-16, 8, Beijing, China, pp: 619-624.

Rezatofighi SH, Soltanian-Zadeh H, Sharifian R, Zoroofi RA. 2009. A new approach to white blood cell nucleus segmentation based on gram-schmidt orthogonalization. In: Proceedings of the international Conference on Digital Image Processing, ICDIP, March 7 – 9, Bangkok, Thailand, pp:107-111. doi: 10.1109/ICDIP.2009.19.

Sabour S, Frosst N, Hinton GE. 2017. Dynamic routing between capsules. In: preceding of the 31st Conference on Neural

Information Processing Systems (NIPS 2017), December 5 - 7, Long Beach, CA, USA, pp: 3859–3869. doi.org/10.48550/arXiv.1710.09829

Sengur A, Akbulut Y, Budak U, Comert Z. 2019. White blood cell classification based on shape and deep features. In: Proceeding of International Conference on Artificial Intelligence and Data Processing Symposium, IDAP, September 21-22, Malatya, Turkey**,** pp 1-4.

Singh R, Ahmed T, Kumar A, Singh AK, Pandey AK, Singh SK. 2021. Imbalanced breast cancer classification using transfer learning. IEEE/ACM Trans Comput Biol Bioinform, 18(1): 83-93. doi: 10.1109/TCBB.2020.2980831.

Shiri P, Baniasadi A. 2021. Convolutional fully connected capsule network (CFC-CapsNet). In : Proceeding of the International Conference of Workshop on Design and Architectures for Signal and Image Processing (DASIP) - 14th edition, January 18-20, Budapest, Hungary, pp:19.

Shiri P, Sharifi R, Baniasadi A. 2020. Quick-CapsNet (QCN): a fast alternative to capsule networks. In: Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, November 2-5, Antalya, Turkey, pp:1-7. doi: 10.1109/AICCSA50499.2020.9316525.

Somuncu E, Atasoy Aydın N. 2021. Realization of character recognition application on text images by convolutional neural network. J FAC ENG ARCHIT GAZ, 37(1):17-27.doi: 10.17341/gazimmfd.866552.

Stock W, Hoffman R. 2000. White blood cells 1: non-malignant disorders. Lancet, 355(9212): 1351-1357. doi: 10.1016/S0140-6736(00)02125-5.

Tamang T, Baral S, Paing MP. 2022. Classification of white blood cells: a comprehensive study using transfer learning based on convolutional neural networks. Diagnostics, 12(12).

Tasdelen A, Ugur AR. 2021. Artificial intelligence research on COVID-19 pandemic: a bibliometric analysis. In: Proceedings of 5th International Symposium on Multidisciplinary Studies and Innovative Technologies, ISMSIT, October 21-23, pp: 693-699. doi: 10.1109/ISMSIT52890.2021.9604573.

Vigueras-Guillén JP, Patra A, Engkvist O, Seeliger F. 2021. Parallel capsule networks for classification of white blood cells. In: preceding of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), September 27 to October 1, Strasbourg, France, pp: 743-752. doi.org/10.48550/arXiv.2108.02644

Xi E, Bing S, Jin Y. 2017. Capsule network performance on complex data. URL: https://arxiv.org/abs/1712.03480v1 (accessed: May 16, 2024).

Xiang C, Zhang L, Tang Y, Zou W, Xu C. 2018. MS-CapsNet: a novel multi-scale capsule network. IEEE Signal Process Lett, 25(12): 1850-1854.

Yao X, Sun K, Bu X, Zhao C, Jin Y. 2021. Classification of white blood cells using weighted optimized deformable convolutional neural networks. Artif Cells Nanomed Biotechnol, 49(1): 147-155. doi: 10.1080/21691401.2021.1879823.

Zhao L, Huang L. 2019. Exploring dynamic routing as a pooling layer. In: Proceeding of IEEE/CVF International Conference on Computer Vision Workshop (ICCVW),October 27-28, Seoul, Korea, pp: 738 – 742.

Zhao Z, Kleinhans A, Sandhu G, Patel I, Unnikrishnan KP. 2019. Capsule networks with max-min normalization. URL:http://arxiv.org/abs/1903.09662 (accessed: May 26, 2024).