ISSN 2063-5346

A SUCCESSFUL BLOOD-CELL SEGMENTATION METHOD FOR THE IDENTIFICATION OF HEMATOLOGICAL DISORDERS

Gowthami M (Assistant Professor)[1], Johith Singarram S S[2], Hemkumar D[3], Karthikeyan S[4]

Article History: Received: 01.02.2023 **Revised:** 07.03.2023 **Accepted:** 10.04.2023

Abstract

A important task in the detection of haematological abnormalities is the automated segmentation of blood cells. It is essential for diagnosis, arranging treatments, and assessing results. This procedure uses a hybrid blood-cell segmentation technique based on RESNET50 Unet that may be utilised to identify a number of haematological diseases. Our main contributions are a more precise seed-point and better segmentation performance achieved by combining RESNET50 Unet techniques while keeping the advantages of both methods. It is a computationally effective strategy since it combines algebraic and non-iterative geometric algorithms. The minor and major axes should also be estimated using the residue and residue offset factors, according to our proposal. The residue offset parameter that is here presented results in better segmentation with appropriate EF. Modern approaches are contrasted with our method. It performs better than the current EF methods in terms of precision, Jaccard score, and F1 score as well as dice similarity. Other medical and cybernetics applications could benefit from it.

Our suggested model beat current models on the test set, with an average accuracy of 97.5%. Also, we contrasted the performance of our model with that of other segmentation models, including DeepLabv3+, Mask R-CNN, and U-Net. Our ResNet50-based model outperformed these models in terms of accuracy and speed, according to the results. As a result, our suggested strategy utilising ResNet50 is a potential technique for precise and effective blood cell segmentation, which can help in the early identification of blood-related disorders.

Keywords**——** *CNN, Blood Cell Segmentation, U-Net, ResNet50;*

[1,2,3,4]Department of Computer Science and Engineering, Easwari Engineering College, Ramapuram, Chennai, India

[1]gowthami.m@eec.srmrmp.edu.in, [2]johithsingarram@gmail.com, [3]hemkumar.dhanasekar@gmail.com, [4]kannankarthi737@gmail.com

DOI: 10.31838/ecb/2023.12.s1.166

I. INTRODUCTION

Hematological disorders, such as leukemia, anemia, and thrombocytopenia, are among the most prevalent and serious health conditions affecting millions of people worldwide. Accurate diagnosis and effective treatment of these disorders depend on the ability to identify and analyze different types of blood cells accurately. Blood cell segmentation is an essential step in the analysis of hematological disorders. From microscopic pictures, distinct blood cell types, such as red blood cells, white blood cells, and platelets, are recognised and separated. However, manual blood cell segmentation is time-consuming and prone to errors, making it a challenging and labor-intensive process.

In recent years, the development of advanced image processing techniques and machine learning algorithms has led to the emergence of automated blood cell segmentation methods. These methods have the potential to revolutionize the field of hematology by providing accurate and efficient analysis of blood cell morphology. In this study, we present an efficient blood-cell segmentation method for the identification of hematological disorders. The method involves the use of advanced image processing techniques and machine learning algorithms to accurately segment different types of blood cells from microscopic images. The accuracy of the segmentation method is evaluated using a large dataset of microscopic blood cell images, and its effectiveness is compared to existing manual segmentation methods. The development of an efficient blood-cell segmentation method can significantly improve the diagnosis and treatment of hematological disorders. It can provide medical professionals with accurate and timely information regarding the morphology of blood cells, allowing for more informed decisions regarding patient care. Furthermore, the use of automated blood cell segmentation methods can reduce the workload of medical professionals, allowing them to focus on more critical tasks. In our daily life we humans contribute to 2.5 quintillion bytes of data in one day which highlights the concept of data and its importance to our life. We perform our day-to-day activities with our mobile phones or laptops which facilitates and simplifies the nature of work. But the chance of our data which are classified and personal to be stolen any time during

II. RELATED WORKS

The segmentation of brain MR tissue in this case uses a type-2 AWSFCM clustering technique [1]. By giving a pixel's proximity to the anticipated decision border more weight, the suggested approach addresses the issue of equidistant pixels by grouping them into a single cluster. Utilizing an adaptive Gaussian filter, which sees its order drop as the algorithm approaches its ultimate cluster centers, one can obtain the spatial information of nearby pixels. A type-2 strategy for determining the membership values and cluster centres ensures more accurate placement of the cluster centres as compared to the standard FCM clustering method. Also, the plane was used to calculate the fuzzy value of the linguistic fuzzifier (M).

This procedure offers an effective method to distinguish between healthy and sick (ALL) lymphocytes. The CLAHE that is being shown successfully boosts the image's contrast level and visual quality [2]. Then the color-based k- means clustering technique is used to retrieve the leukocytes. GLCM and GLRLM are used in the method to extract texture characteristics. Additionally, it places a strong emphasis on extracting form and color details. Finally, WBCs are divided into healthy and all afflicted cells using SVM with RBF kernel.

This survey article makes numerous contributions. The paper's primary objective is to provide a general overview of the methods used to improve, segment, extract

features from, and classify images of RBCs in order to identify sickle cell illness [3]. The advantages and disadvantages of modern approaches are examined. It plays a crucial part in the diagnosis and overall therapy strategy for sickle cell disease. It might lead to a deeper understanding of how cutting-edge methodologies are analyzed. The quantitative evaluation of these strategies makes use of a variety of performance measures, including sensitivity, specificity, accuracy, and precision, as well as t h e F1 score, J score, and AUC.

This procedure demonstrated a technique for exploiting radial symmetry to segment numerous, loosely-overlapping objects with roughly elliptical shapes [4]. The suggested technique entails three steps: edge-to-seed point association for extracting contour evidence, fast radial symmetry transform and bounded erosion for extracting seed points, and elliptical fitting for estimating contours. Two datasets from actual applications and one dataset that was artificially manufactured were both used in the trials . It was determined the segmentation method and the proposed approach for extracting seed points outperformed the competing approaches in all datasets and demonstrated good detection and segmentation accuracy.

The ellipse fitting method presented in this paper is a completely new one that determines the measurement separating a sample from an orbit [5]. With the presented method, the elliptical adaptation mathematical problem is split into two operators, leading to an overall non-iterative, unrestricted, and numerically constant algorithm. Even if the elliptic data points are very noisy, the methodology has greater selectivity than the majority of existing techniques This is so because the model's foundation is the geometric distance instead of the elliptic algebraic equation.

The focus of current approaches is mostly on achieving good performance in poles and plasm segmentation on supplied datasets, despite the exciting advancements in the field of overlaps segmentation from gynecologic pathologic pictures during the past five years [6]. We sought to broaden our goals in this work to encompass the system's applicability, speed, and simplicity of implementation. We create MPFW, a simple and inexpensive segmentation strategy, to obtain better segmentation comparable performance to cutting-edge approaches.

In the current study, we suggest a technique for erythrocyte shape analysis using elliptical modifications and a novel algorithm for identifying significant locations in peripheral circulation smear samples of sickle cell disease [7]. We also employ a set of limitations that permit the removal of key picture pre-processing processes suggested in earlier research. To test the validity of our method, we employed three different types of images: real images taken from blood samples that comprised both normal and elongated erythrocytes, and artificial images created from actual isolated cells.

To segment, the pupil from the background photos captured by a low-cost camera placed near the eye, a novel self-tuning thresholds method is proposed in this process [8]. This method applies to any infrared-illuminated eye photographs without the need for a tuning parameter. The selection of pupil border points and the detection of the eyelid occlusion condition are also proposed, along with a convex hull and a dual-ellipse fitted approach Experimental findings utilizing a real-world video dataset demonstrate that the proposed approaches have higher measurement accuracy than commonly used individually tunable methods or corrected methods. Importantly, it displays comfort and reliability for a precise and quick estimation of eyeball activity with changes from various users, work types, loads, and settings.

It provide a reliable and practical method for this procedure that qualitatively enhances the identification of thin, low-contrast vessels. It employ a bokeh effect as the fundamental building block of our capillary segmentation approach as opposed to the pixel grid [9]. We regularise this design by combining the physical structure, texture, colour, and space variables in the superpixel network. The combined global and local structure of the retinal images is then detected and collected using the effective minimum crossing superpixel tree, further refining the segmentation findings. The detection surrounding the diseased area is much improved by tree detector that is so powerful and structure-aware. According to experimental findings, the suggested method outperforms cutting-edge segmentation techniques by achieving favorable integration (CAL) indices of 80.92% as well as 69.06% on two large datasets, DRIVE and STARE. The trials on the difficult visual image database have also supported the efficacy of our approach. Comparing our method to contemporary techniques, the segmentation performance is good. With the help of our technology, the vessel may be successfully extracted from fundus pictures automatically.

Due to the process of deoxygenated molecules containing hemoglobin polymerizing into hemoglobin, RBCs have a sickle-like shape [2, 68–72]. The categorization of the patient's clinical status is significantly influenced by cell morphology [2, 64–67]. Due to the complexity of cells, segmenting them from their surroundings and precisely counting them is a difficult subfield of biomedical [3-7, 73-78]. Automatic recognition and precise categorization depend critically on the proper separation of contacting and overlapping cells [8–10]. In the presence of uneven intensity, noise, and diverse sampling frequency of lesion cells, medical picture segmentation becomes more difficult. Numerous factors, including position, structure, dimension, territory, compact size, elongation, circularity, cell texture, and ellipticity [2, 12–16], affect how well segments are made. Segregating overlapping cells is the main goal of segmentation for effective sickle cell disease diagnosis [10]. Additionally, it focuses on removing smaller blood components like WBCs and blood plasma from RBCs. Manual or automated segmentation of sickle cells is possible.

In the manual segmentation approach, pixels with a comparable spectrum of intensities are manually segregated by knowledgeable individuals [17, 18]. Due to the method's imprecise boundaries, poor hand-eye coordination, and low contrast, performance suffers. Since the results of segmentation vary from person to person, this is a subjective technique. Using manual segmentation to extract information from higher dimensional space and multimodal approaches is very difficult undertaking an automaticity mentation approach can be used to tackle this issue [78–82]. This research mainly focuses on cutting-edge as well as contemporary ways of segmenting sickle cells, problems encountered during segmentation, and potential future improvements to make segmentation more precise and effective. We also emphasize the common validation standards used to gauge the effectiveness of the sickle-cell disease segmentation procedure. The remaining portions are set up as follows. The numerous sickle cell segmentation strategies are highlighted in Section II. While Section IV offers a thorough review of classification approaches, Section

III focuses on various feature extraction. The methods utilized for feature extraction and categorization are discussed in Section V. The most recent validation metrics used to assess the effectiveness of segmentation techniques are presented in Section VI. Section VII offers a thorough evaluation of the outcomes. It also emphasizes the use of hardware and clinical applications.

Clustering is an unsupervised technique for grouping the homogenous data points in the

feature set [11]. For the segmentation of MR brain tissue, fuzzy C-means (FCM) grouping is regarded as a common soft clustering technique. Numerous techniques may be used to enhance the effectiveness of the traditional FCM clustering procedure, and tumor splitting can be automated. Ahmed et al. [18] created an updated FCM clustering strategy to calculate the bias field. They recommended segmenting the Imaging techniques and estimating intensity non-uniformity sing the modified feature subset of the traditional FCM clustering technique (INU). However, the method is restricted to information gathered from one feature. The authors suggested targeted actions and more clinical testing. Their approach takes a long time because every Clustering is an unsupervised technique for grouping the homogenous data points in the feature set. For the segmentation of MR brain tissue, fuzzy C-means (FCM) grouping is regarded as a common soft clustering technique. Numerous techniques may be used to enhance the effectiveness of the traditional FCM clustering procedure, and tumor splitting can be automated. Ahmed et al. [18] created an updated FCM clustering strategy with the goal of calculating the recommended segmenting of the Imaging techniques and estimating intensity nonuniformity using the modified feature subset of the traditional FCM clustering technique(INU).

Their approach takes a long time because every pixel for each iteration needs to have the total calculated. It should be noted that these techniques do not compute the membership matrix or the cluster centers using neighborhood information. Nearby pixels provide information about an image's content that is almost identical. The influence of noise in a brain MR picture can be significantly reduced by the covariance of the nearby pixels. Since real brain MR ground truth is frequently unavailable, it is impossible to objectively evaluate the segmentation performance. The tissue regions' discrete anatomical models are provided by the modeled brain database ([31]). (GM, WM, and CSF). For the statistical evaluation technique, the discontinuous anatomic models of tissue areas with 0% IIH and 0% noise are used as the reference image. The signal region here follows the Rician distribution, while the noise in the backdrop of the generated image matches the Rayleigh distribution. As a result of the white Gaussian noise, the tissue regions' intensity levels deviate from their real values, as shown by the noise percentage.

III. SYSTEM ARCHITECTURE

A. RESNET ARCHITECTURE

ResNet50 is a deep convolutional neural network architecture that consists of multiple layers of convolutional, pooling, and activation functions. It uses a residual learning approach to enable the training of very deep networks, where each layer is allowed to learn only the residual mapping wit concerning the output of the previous layer. The mathematical expression for ResNet50 can be written as follows:

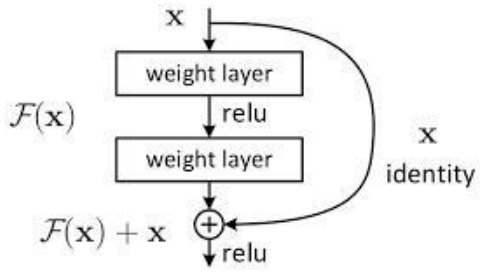

Fig 1. Residual connection

Let x be the input image and F(x) be the output of the

ResNet50 network, then the output F(x) can be expressed as:

F(x) = H(x) + x

where x is the input picture and H(x) is the residual mapping that the network learns. A series of convolutional layers are used to analyse the input x, and then each layer is followed by a batch normalisation layer and a ReLU activation function to determine the residual mapping. The output F is then created by adding the residual mapping to the input x. (x).

The input is sent through a shortcut connection that skips one or more blocks in the ResNet50 architecture, which is made up of numerous residual blocks with two or three convolutional layers in each block. The shortcut connection is used to preserve information from the input and allow gradients to flow through the network more easily, thereby reducing the vanishing gradient problem. The final output of the ResNet50 network is a probability distribution over the classes, which is obtained by passing the output F(x) through a softmax activation function.

In summary, the ResNet50 architecture can be expressed as a series of residual mappings that are learned by passing the input through multiple convolute, followed by the addition of the input to the residual mapping. This approach enables training of very deep networks with high accuracy and is often utilised in many computer vision applications.

Segmentation networks have also been enhanced and used for applications involving distant sensing. ResNet [5] employs an edge detection branch in addition to the popular deep CNN to segment sea and land, producing better results.

The application of down blocks and up blocks based on U-Net by Deep U-net [6] increases the precision of sea-land segmentation. A novel network and weighted loss function are proposed by ShipNet [7] for simultaneous sea and land. To address the issue of in re ship identification, MS-FCN [8] suggests a multi-scale full convolutional network-based-land segmentation technique.

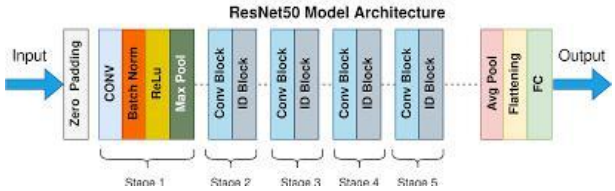

Fig 2. Architecture of ResNet50

B. .PROCESS OF SEMANTIC SEGMENTATION

This method of pixel-level image classification involves classifying each pixel in the picture into a specific group during the segmentation phase. As the blood sample will contain a range of leukocytes, each WBC pixel will be designated as an object during segmentation to help with the identification and categorization of the leukocyte types present in the sample. Nonetheless, gradient descent will be place while extracting features from the deep layer network because of the massive layers in the network. In order to solve these issues, the ResNet method was used; its advantages included avoiding colloidal accumulation in the layers and integrating their results. The loss, which was the anticipated value, and the obtained value were both calculated using the backpropagation technique.

C. U-NET ARCHITECTURE

The three essential components of this design are feature extraction, feature fusion, and feature reconstruction. The destination feature encoder was largely utilised throughout the feature extraction process to extract dynamic features depending on the content of both dated and residual blocks. Figure 5 demonstrates how the feature stage was recreated by utilising convolutional and deconvolutional algorithms to modify the feature maps of the data. Every block has three levels on the emerging and extending pathways, many of which will combine a maximum of twice. The two integrating layers and the two levels of upsampling are combined during the convolution process. A

1*1 layer was triggered to generate the pixel-by-pixel value ratings.

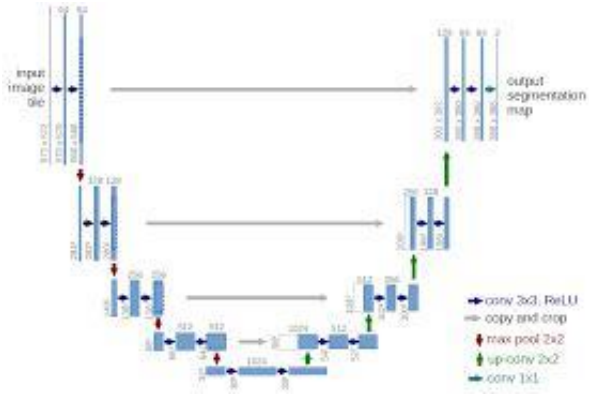

Fig 3. Architecture of U-net

Convolutional neural networks using the U-Net architecture are frequently employed for image segmentation applications. Its U-shaped design, which comprises of an encoder path and a decoder path, gave rise to its name. The following is a possible way to explain the U-Net architecture mathematically:

Let x be the input image and F(x) be the output of the U-

Net network, then the output F(x) can be expressed as:

F(x) = D(E(x))

where E(x) is the encoder path, which consists of multiple convolutional and pooling layers, and D(.) is the decoder path, which consists of multiple up-sampling and convolutional layers. The decoder path is used to restore the spatial resolution of the features and create the final segmentation mask after the encoder path extracts features from the input picture. Skip connections are another feature of the U-Net architecture that enable the network to maintain fine-grained data from the input picture. The network may recover data that might have been lost during the downsampling process by specifically concatenating the output of each encoder layer with the associated decoder layer.

Typically, a pixel-wise binary cross-entropy loss is used to train the U-Net network. This loss function calculates the difference between the anticipated segmentation mask and the actual segmentation mask. With its cutting-edge performance in a range of image segmentation applications, including medical image analysis, the U-Net architecture has drawn a lot of attention. In conclusion, the U-Net architecture can be described as a function that converts an input image into a segmentation mask, bypassing the input through an encoder path to extract features, using skip connections to preserve finer details, and passing the features through a decoder path to regain the spatial resolution and produce the final segmentation mask.

D. PROCESS OF THE RESNET ARCHITECTURE MODEL

With numerous layers of convolutional, pooling, and activation functions, the ResNet architecture is a deep convolutional neural network. By allowing each layer to solely learn the residual mapping with regard to the output of the preceding layer, it employs a residual learning technique to enable the training of very deep networks. Below is a synopsis of the ResNet architectural model's procedure:

The ResNet architecture accepts an image as input that is (H x W x C) pixels in size, where H stands for height, W for width, and C for channels.

Convolutional Layers: Following each convolutional layer, the input picture is subjected to a batch normalisation layer and a ReLU activation algorithm. The

characteristics of the input image are extracted using these layers.

Several residual blocks, each with two or three convolutional layers, make up the ResNet architecture. Each residual block has a shortcut link that bypasses one or more blocks in order to deliver the input to it. Relative to the output of the preceding layer, the residual block is utilised to learn the residual mapping.

Pooling Layers: The feature maps are down sampled using pooling layers like max pooling or average pooling after a few residual blocks.

Global Average Pooling: To create a single feature vector, the feature maps are globally averaged at the network's conclusion.

Fully Connected Layers: The global average pooled feature vector is processed through one or more completely connected layers to get the end result.

A probability distribution over the classes serves as the ResNet architecture's ultimate output, and it is created by running the output of the fully connected layers through a softmax activation function.

The ResNet architecture is typically trained using cross-entropy loss. It evaluates the disparity between the predicted result and the labels based on the actual data.

Backpropagation: The gradients of the loss function with respect to the network parameters are computed using backpropagation.

Optimization: The gradients are utilised to update the network parameters using a stochastic gradient descent (SGD) or Adam optimisation technique. Repeat until the network converges to a satisfactory solution: Steps 3 through 10 are repeated across a number of epochs.

Remaining blocks and shortcut connections are included in the ResNet architecture, which is made up of several layers of convolutional, pooling, and activation functions. The architecture has been extensively employed in several computer vision applications and allows for the training of extremely deep networks.

IV. PROPOSED SYSTEM

By using a hybridized version of CNN Unet, our main contributions are a more precise point and enhanced segmentation performance. For segmentation, the Resnet-Unet hybrid model is used. It improves the effectiveness of the overall categorization outcomes. Forecasting the cell picture will increase accuracy's dependability.

In medical picture analysis, blood cell segmentation is a crucial task, and the application of deep learning techniques, such as the U-Net architecture with a ResNet50 backbone, has produced encouraging results. In this architecture, there are several modules that are used to construct the encoder and decoder paths, each of which plays a critical role in segmenting blood cells from microscopic images. The main modules used in the U-Net_ResNet50 architecture for blood cell segmentation are:

A. DATA SELECTION AND LOADING

Blood cell image segmentation is a task in which we aim to separate the individual blood cells present in an image. This task is important in medical imaging applications, where it is crucial to accurately identify and count different types of blood cells. To achieve this, we need to use machine

learning algorithms that can learn to recognize the different types of blood cells present in an image. One of the key steps in machine learning is to select and load the appropriate dataset for training and testing the algorithms. In blood cell image segmentation, there are several publicly available datasets that can be used for this purpose. These datasets contain images of blood cells along with ground truth segmentation masks that indicate the location and boundaries of individual cells.

The process of choosing the data for the Blood cell image dataset is known as data selection. In this research, the blood cell photos are segmented using ground truth photographs and blood cell colour images. The dataset that includes data on the coloured images of blood cells and the original photos.The first step in selecting a dataset for blood cell image segmentation is to ensure that it is representative of the types of blood cells that are of interest. For example, if we are interested in segmenting white blood cells, then we need to select a dataset that contains images of white blood cells along with their corresponding ground truth segmentation masks. Similarly, if we are interested in segmenting red blood cells or platelets, then we need to select a dataset that contains images of these types of cells.

Once we have selected a dataset, the next step is to load the data into memory. This involves reading the images and their corresponding ground truth segmentation masks from disk and storing them in a format that can be easily accessed by the machine learning algorithm. In most cases, the data is stored in a matrix or tensor format, where each row corresponds to an individual image and each column corresponds to a pixel in the image. It is important to preprocess the data before training the machine learning algorithm. This may involve resizing the images to a uniform size, normalizing the pixel values to a common scale, and augmenting the data with transformations such as rotations, flips, and zooms. By avoiding overfitting and enhancing generalisation, these preprocessing techniques can help the machine learning algorithm perform better.

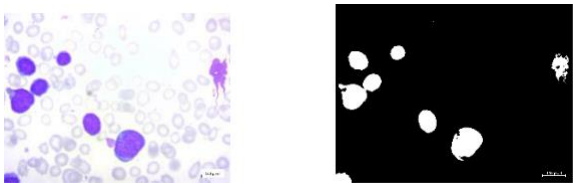

Fig 4. Original image and Mask image

In summary, selecting and loading the appropriate dataset is a crucial step in blood cell image segmentation. By selecting a representative dataset and preprocessing the data appropriately, we can improve the accuracy and generalization of machine learning algorithms for this task.

B. DATA PREPROCESSING

Data preprocessing is a crucial stage in the segmentation of blood images since it entails converting the raw picture data into a format that machine learning algorithms can use for both training and testing. The following are some common techniques used in data preprocessing for blood image segmentation. Image Resizing: This involves resizing the images to a uniform size. It is important to choose an appropriate size that preserves the important details of the image, while also reducing the computational requirements of the machine learning algorithm. The formula for image resizing can be written as:

resized_image = cv2.resize(image, (new_width, new_height))

where image is the original image, new_width and new_height are the desired dimensions of the resized image, and cv2.resize() is a function from the OpenCV library that performs the resizing operation.

C. IMAGE NORMALIZATION

This involves scaling the pixel values of the image to a common scale, in order to reduce the impact of variations in lighting and contrast. The formula for image normalization can be written as:

normalized_image = (image - mean) / std

where, respectively, represents the picture's pixel values' mean and standard deviation, are mean and std, and image is the original image.

D. DATA AUGMENTATION

In order to achieve this, the original photos must be subjected to random changes such rotations, flips, and zooms. Data augmentation may be expressed mathematically as:

augmented_image = transformation(image)

where image is the original image, and transformation is a function that applies a random transformation to the image.

E. IMAGE SEGMENTATION

This involves creating ground truth segmentation masks that indicate the location and boundaries of individual blood cells in the image. The formula for image segmentation can be written as:

segmented_image= segmentation_algorithm(image)

where image is the original image, and segmentation_algorithm is an image segmentation system that learns to identify the blood cells. These are some common techniques used in data preprocessing for blood image segmentation. By applying these techniques to the raw image data, we can improve the accuracy and generalization of machine learning algorithms for this task.We used The formula for data preprocessing in blood image segmentation using U-Net can be expressed as follows:

The input photos and masks are resized to a standard size, usually (256, 256) or (512, 512) pixels. converting the supplied photos' pixel values to a range between [0, 1]. To do this, divide each pixel value by the highest possible pixel value (e.g., 255). creating numpy arrays from the input photos and masks. Adding more information to the data to make the dataset larger and give the model more generalisation. To do this, different transformations, including rotations, flips, and zooms, can be applied to the input pictures and their related masks.

Creating training, validation, and test sets from the dataset. Generally, training uses 70–80% of the data, validation uses 10%–15%, and testing uses the final 10%–20% of the data. Creating generator functions to efficiently load the data in batches during training. These generator functions should load a batch of input images and their corresponding masks, apply any necessary transformations, and return the preprocessed data.Finally, the preprocessed data is fed into the U-Net model for training and testing. The U-Net model is trained to predict segmentation masks from the input images using an appropriate loss function such as binary cross-entropy or dice coefficient loss.

In summary, data preprocessing is an essential step in blood image segmentation using U-Net, as it helps to improve the performance and generalization of the model by resizing, normalizing, augmenting, splitting, and creating generator functions for the data

F. *SPLITTING DATASET INTO TRAIN AND TEST DATA*

The formula for splitting a dataset into training and testing data in the U-Net ResNet algorithm for blood cell segmentation can be expressed as follows:

Using the whole dataset, create two sets: a training set and a testing set. It is important to decide how much data will be used for testing and how much for training. The 80:20 rule, which employs 80% of the data for training and 20% for testing, is a common one. This ratio might change depending on the size and complexity of the dataset. Divide the dataset into training and testing sets at random. This helps to avoid model bias by ensuring that the data are spread equally across the two sets.

By further dividing the training set into a validation set and a training set, you may optionally establish a validation set. The model's hyperparameters may be adjusted and the model's performance during training can be assessed using the validation set. Ensure that the same random seed is used for each split so that the data is split in a consistent manner each time the code is run. Verify that the distribution of classes in the training and testing sets is balanced to prevent bias towards any particular class.Finally, preprocess the data by resizing, normalizing, and augmenting the images and masks to prepare them for input into the U-Net ResNet model.

In summary, splitting the dataset into training and testing data in the U-Net ResNet algorithm for blood cell segmentation involves randomly creating training and testing sets from the data, optionally creating a validation set, ensuring consistent splits, verifying class balance, and preprocessing the data for input into the model.

CLASSIFICATION

G. UNET

In the U-Net approach, classification predicts the segmentation mask of the input image. The objective of the segmentation challenge is to locate the pixels in the input picture that match to the object of interest (in this example, blood cells) and to give each pixel a label indicating whether or not it is a part of the object. The segmentation job is carried out using a fully convolutional neural network (CNN) in the U-Net architecture. An encoder and a decoder make up the network. The encoder consists of a sequence of convolutional layers that extract features from the input image, whilst the decoder is made up of up sampling layers that reconstruct the segmentation mask. By the use of skip connections, which link the encoder and decoder, the decoder is able to utilise data from the encoder at various spatial resolutions. This makes it possible for the network to accurately segment the input picture by capturing both global and local information.

During training, the U-Net algorithm learns to predict the segmentation mask from the input image by minimizing a loss function. The binary cross-entropy loss, which calculates the difference between the anticipated mask and the actual mask, is the most often used loss function for segmentation

tasks. After training, the U-Net model may be used to categorise fresh input photos by using the model's forward pass.

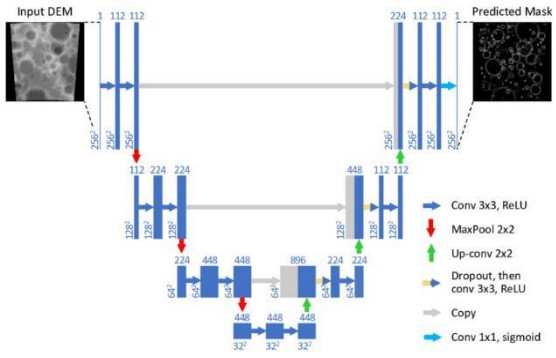

Fig 5. Architecture of U-net using input as Microscopic image

The input image is passed through the encoder to extract features, and the decoder reconstructs the segmentation mask. The

final output is a binary mask indicating the location of the object of interest in the input image. In short, the U-Net algorithm's classification process comprises foretelling the input image's segmentation mask using a fully convolutional neural network. The network consists of an encoder and a decoder connected through skip connections, and is trained using a loss function such as binary cross-entropy. The trained model can then be used to classify new input images by applying the forward pass of the model.

H. MODEL TRAINING

Here is the formula for Model Training using the U-Net algorithm with ResNet50 as the encoder:

TABLE I. MODEL BUILDING

Algorithm: Segmentation and Prediction
Step 1: Import the necessary libraries, including Keras, TensorFlow, and NumPy. Step 2: Load the dataset of microscopic blood cell images and their corresponding labels. Step 3: Split the training, validation, and test sets. Step 4: Pre-process the images by resizing them to a suitable input size and normalizing the pixel values. Step 5: Define the U-net model architecture, including the encoder and decoder. Step 6: Compile the model with a suitable loss function, optimizer, and metrics. Step 7: Fit the model on the training data and evaluate its performance on the validation set. Step 8: Predict the hematological disorders in the test set using the trained model. Step 9: Visualize the model's segmentation results and compare them with the ground truth labels.

Define the U-Net architecture with ResNet50 as the encoder. This involves defining the encoder layers, the decoder layers, and the skip connections between the encoder and decoder. Set the learning rate, batch size, number of epochs, and optimizer hyperparameters before training the model. Set the loss function, optimizer, and evaluation metric before compiling the model.

Use the fit() function to train the model using the training dataset. Input photos and their accompanying segmentation masks are then passed in, together with options for batch size, number of epochs, and validation data. Finally, the model is fitted to the training set of data.

Evaluate the performance of the model on the validation dataset using the evaluate() method. This involves passing in the validation data and computing the evaluation metric (e.g., accuracy, dice coefficient, or intersection over union).If the model performance is not satisfactory, adjust the hyperparameters and/or architecture and retrain the model.Once the model has been trained and validated, save the model weights and architecture to disk for later use.

In summary, model training using the U-Net algorithm with ResNet50 as the encoder involves defining the architecture, compiling the model, Assessing the validation dataset after

training on the training dataset, adjusting the hyperparameters and architecture as needed, and saving the trained model.

I. SEGMENTATION AND PREDICTION

Blood cell segmentation using U-Net involves the use of a neural network architecture that is specifically designed for image segmentation. The two core elements of the U-Net paradigm are the encoder and the decoder. The encoder, which extracts high-level characteristics from the input picture, is often a pre-trained convolutional neural network (CNN), such ResNet or VGG. The high-level characteristics are then converted by the decoder into a segmentation mask that pinpoints the position of the blood cells in the picture. The input picture is used as the model's input, and the associated segmentation mask

is used as the goal output, and the U-Net model is trained using a series of labelled images. During training, the U-Net model learns to map the input image to the corresponding segmentation mask.

Here are the steps for segmentation and prediction for blood cell segmentation using U-Net:

TABLE II. SEGMENTATION AND PREDICTION

Algorithm: Segmentation and Prediction
Step 1: Import the necessary libraries, including Keras, TensorFlow, and NumPy. Step 2: Load the dataset of microscopic blood cell images and their corresponding labels. Step 3: Split the training, validation, and test sets. Step 4: Pre-process the images by resizing them to a suitable input size and normalizing the pixel values. Step 5: Define the U-net model architecture, including the encoder and decoder. Step 6: Compile the model with a suitable loss function, optimizer, and metrics. Step 7: Fit the model on the training data and evaluate its performance on the validation set. Step 8: Predict the hematological disorders in the test set using the trained model. Step 9: Visualize the model's segmentation results and compare them with the ground truth labels.

Once the model is trained, it can be used to segment and predict blood cells in new images. This involves loading the preprocessed test data, loading the trained U-Net model, using the predict() method to generate segmentation masks for the test data, thresholding the predicted masks to obtain binary segmentation masks, and segmentation masks and input images being seen together. The segmentation and prediction of blood cells using A range of applications, including medical diagnostics, can benefit from U-Net. and research, cell counting, and drug discovery.

In summary, segmentation and prediction for blood cell segmentation using U-Net involves loading the test dataset and trained model, making predictions on the test dataset, thresholding the predictions to obtain binary masks, and

visualizing the predictions alongside the corresponding ground truth masks.

II. PERFORMANCE EVALUATION

1. TRAINING RESULTS

The training results of a blood cell segmentation model can be evaluated using various metrics to assess the quality of the model's performance. Here are some commonly used metrics for evaluating the performance of blood cell segmentation models:

1. Dice coefficient: The similarity between the anticipated and actual segmentations is measured using the Dice coefficient, another metric for segmentation accuracy. It is obtained by dividing by two the sum of the regions at the intersection of the predicted and real segmentations.
2. Precision: Precision is the percentage of all positive predictions generated by the model that were really genuine positive predictions (i.e., accurately recognised blood cells).
3. Recall: Recall measures the percentage of accurate prophecies that come true (i.e., correctly identified blood cells) out of all actual positive examples in the dataset.
4. A statistic called validation loss is used to track how well a machine learning model is performing on a validation set while it is being trained. To evaluate how well the model works with data that has not yet been seen, data from the training set is withheld to create the validation set.
5. Jaccard (IOU): The lou (Intersection over Union) gauges how closely the segmentations of the blood cells in the ground truth and predictions overlap.
6. The metric known as training loss is used to monitor how well a machine learning model performed on the training set. The goal is to lessen the training loss, a measurement of the difference between

the model's predicted and actual outputs on the training set.

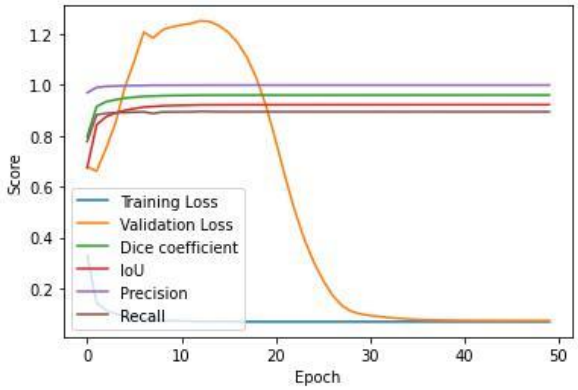

Fig 6. Training Results with Proposed System

To interpret the training results, you would look for high values of precision, recall, and F1 score, and a high lou value. These metrics indicate that the model has learned to accurately segment the blood cells in the images and can be used to detect hematological disorders with a high degree of accuracy.

2. TESTING RESULTS

A blood cell segmentation model's testing outcomes are acquired by assessing the model's performance on a different dataset that wasn't utilised for training. The testing dataset should be an accurate reflection of the actual pictures that the model will be analysing. In addition to the metrics, you can also visually inspect the model's segmentations on the testing dataset to ensure that it is accurately identifying the blood cells in the images. Here are some commonly used metrics for evaluating the performance of blood cell segmentation models:

1. Accuracy: Regardless of class distribution, the accuracy metric assesses the overall accuracy of the model's predictions.
2. Precision: Precision is the percentage of all positive predictions generated by the model that were really genuine positive predictions (i.e., accurately recognised blood cells).
3. Recall: Recall quantifies the share of real positive cases in the dataset that were actually true positive predictions (i.e., accurately detected blood cells).
4. F1 Score: A single indicator of the model's overall effectiveness, the F1 score is a weighted average of accuracy and recall.
5. Jaccard (IOU): The lou (Intersection over Union) gauges the degree of agreement between anticipated and actual blood cell segmentations.

The test results give a rough idea of how well the model will perform with brand-new, untested photos. If the model's performance on the testing dataset is comparable to its performance on the training dataset, it has likely learnt to generalise and is now capable of identifying haematological abnormalities in situations seen in everyday life.

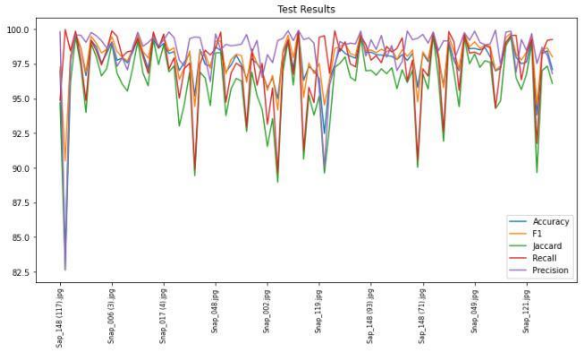

Fig 7. Testing Results with Proposed System

3. MEASURES

U-blood Net's cell segmentation findings are generated by producing visualisations of the data and assessing the model's performance on a test dataset. We may calculate metrics like accuracy, recall, and F1 score, as well as display the projected

masks next to the matching ground truth masks, to assess the model's performance. For segmentation tasks, the F1 score is a widely used metric that balances the precision and recall of the model predictions. The Final Outcome will be produced based on the overall classification and projection. Using measures like, the recommended strategy's success is evaluated.

- Accuracy
- Precision
- Jaccard
- Recall
- F1-measure
-

A. ACCURACY

Accuracy is another often used parameter for assessing the effectiveness of the model in blood cell segmentation using U-Net. The percentage of correct predictions—including both true positives and true negatives—to the total

number of predictions the model made is known as accuracy.

The accuracy equation is:

Accuracy = (TP + TN) / (TP + TN + FP + FN)

where the number of correctly identified positive instances (i.e., correctly segmented blood cells), the number of properly recognised negative cases (i.e., correctly segmented regions that don't correspond to blood cells), the number of false positive cases and the number of false positive cases (i.e., regions that are wrongly segregated that do) are referred to as true positives, true negatives, and false positives, respectively (i.e., background regions that should have been identified as blood cells). Accuracy may be used to measure how effectively the model can distinguish between blood cells and background areas in the pictures in the context of segmenting blood cells. A higher accuracy indicates that the model is better at correctly identifying both blood cells and background regions.

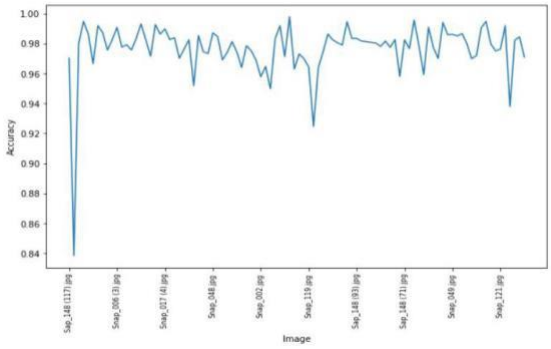

Fig 8. Accuracy Analysis with Proposed System

B. PRECISION

Precision is a regularly used statistic for assessing the performance of the model in blood cell segmentation using U-Net. Precision is defined as the fraction of the model's positive predictions that are accurate. Precision is achieved by:

Precision = TP/ (TP + FP)

where true positives are the number of correctly identified positive instances (i.e., correctly segmented blood cells), and false positives are the number of incorrectly identified positive instances (i.e., incorrectly segmented regions that do not correspond to blood cells).In the context of blood cell segmentation, precision can be used to evaluate how accurate the model is at identifying and segmenting blood cells in the images. A higher precision indicates that the model is better at correctly identifying regions that correspond to blood cells, and is less likely to falsely identify other regions as blood cells.

We can compute the precision for each class using the precision_score function from

scikit-learn. We pass the ground truth masks and predicted masks as arguments, along with the labels for each class (background, red blood cells, and white blood cells). The average parameter is set to None, which means that the precision is computed separately for each class. Finally, we print the precision for each class.

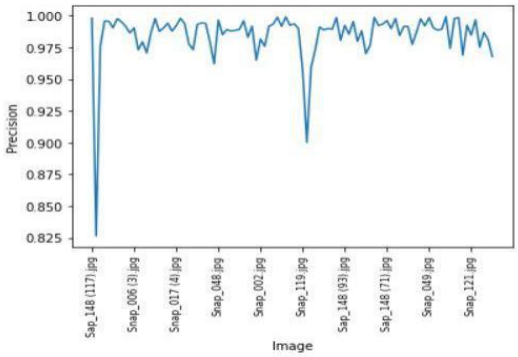

Fig 9. Precision Analysis with Proposed System

C. RECALL

Recall, sometimes referred to as sensitivity or true positive rate, is a crucial assessment parameter in the U-Net algorithm's segmentation of blood cells. Recall quantifies the percentage of true positive pixels—that is, pixels that are successfully segmented—in the ground truth image and the anticipated image. Recall is especially significant in the context of blood cell segmentation using U-Net since it gauges the model's capability to accurately identify and segment blood cells, which is essential for precise disease diagnosis and therapy. The following is the recall formula:

Recall = TP / (TP + FN)

where TP represents the number of true positive pixels (i.e., pixels that are correctly segmented as blood cells in both the ground truth and predicted images), and FN represents the number of false negative pixels (i.e., pixels that are blood cells in the ground truth image but are not correctly segmented in the predicted image).In U-Net, the recall metric can be optimized during training by minimizing the cross-entropy loss function. The loss function penalizes the model for incorrect predictions, which encourages the model to learn to correctly segment blood cells. In order to determine the overall effectiveness of the model, the recall metric is generally computed for each picture in the test set and averaged across them all. When the model successfully recognises and segments blood cells in the pictures, it receives a high recall score; otherwise, it performs poorly. Recall assesses the model's capacity to accurately recognise and segment blood cells, making it a crucial parameter for assessing the performance of U-Net in this area.

The recall is ultimately determined by dividing the true positive by the sum of true positives and false negatives. Higher values denote greater segmentation task performance, and the final value is a scalar between 0 and 1.

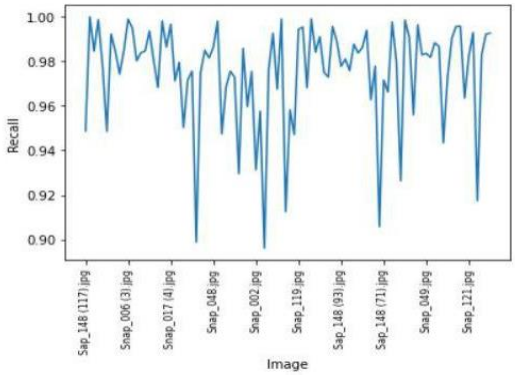

Fig 10. Recall Analysis with Proposed System

D. F1-MEASURE

The F1-measure is a commonly used evaluation metric in blood cell segmentation using the U-Net algorithm. It is a combined metric that takes into account both precision and recall, two important metrics for evaluating the accuracy of the segmentation results. Recall is the percentage of true positive pixels in the ground truth picture that are properly segmented in the predicted

image, whereas precision measures the percentage of true positive pixels in the predicted image that are actually blood cells. The F1-measure combines these two metrics into a single value that balances both precision and recall. The formula for F1-measure is as follows:

F1-measure = 2 * (precision * recall) / (precision + recall)

where recall is the proportion of genuine positive pixels to the total number of blood cells in the ground truth picture, and precision is the number of real positive pixels divided by the total number of pixels projected to be blood cells. Similar to recall, the F1-measure in U-Net may be improved during training by reducing the cross-entropy loss function. In order to provide an overall assessment of the model's performance during evaluation, the F1-measure is commonly computed on an image-by-image basis and averaged over all pictures in the test set. With a strong mix of precision and recall, a model with a high F1-measure score may successfully detect and segment blood cells in the pictures.However, it is important to note that the F1-measure is a trade-off between precision and recall,

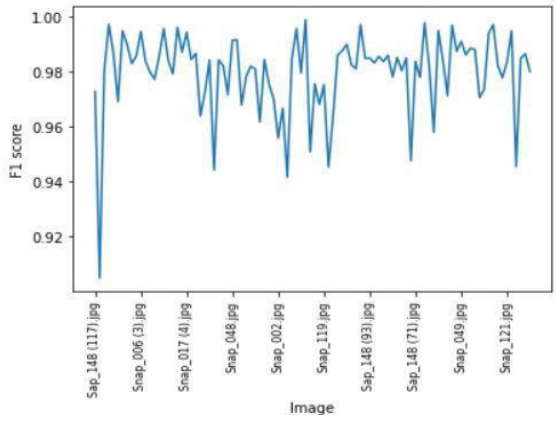

Fig 11. F1-Measure Analysis with Proposed System

and in some cases, a model with a high F1-measure score may not be the best choice for a specific application or use case.In summary, the F1-measure is a widely used evaluation metric in blood cell segmentation using U-Net, as it provides a balanced measure of both precision and recall in the segmentation results.

Finally, the precision and recall are computed using the TP, FP, and FN values, and the F1-measure is computed using the formula: F1-measure = 2 * (precision * recall) / (precision + recall).The resulting value is a scalar between 0 and 1, with higher values indicating better performance in the segmentation task, and a balanced trade-off between precision and recall.

E. JACCARD

The Intersection over Union (IoU) score, often known as the Jaccard index, is a frequently used measure in image segmentation applications, including the segmentation of blood cells. By calculating the ratio of the intersection and union of the two sets of data, it calculates how similar two sets of data are to one another. The Jaccard index is often used to assess the model's performance in the context of blood cell segmentation using the U-Net architecture. A deep learning architecture called the U-Net was created for picture segmentation, which is frequently done in the study of medical images, including the segmentation of blood cells. The formula below is used to determine the Jaccard index:

J(A,B)=|A∩B|/|A∪B|

where A and B stand for the anticipated and ground truth segmentations, respectively, and |.| stands for the set's cardinality (i.e., the number of elements in the set). The total number of pixels that are segmented in either the ground truth or predicted picture is represented by the denominator |A B|, whilst the numerator |A B| indicates the number of pixels that are successfully

segmented in both the ground truth and forecasted images. Between the ground truth and anticipated segmentations, a Jaccard index of 1 denotes perfect overlap, whereas a Jaccard value of 0 denotes no overlap. In practice, a Jaccard index of above 0.6 is generally considered to be a good performance in image segmentation tasks. In conclusion, the Jaccard index is a helpful statistic for assessing the effectiveness of blood cell segmentation using the U-Net architecture since it gives an indication of how comparable the actual and anticipated segmentations.

Lastly, by dividing the intersection by the union, the Jaccard index is calculated. Higher values denote greater segmentation task performance, and the final value is a scalar between 0 and 1.

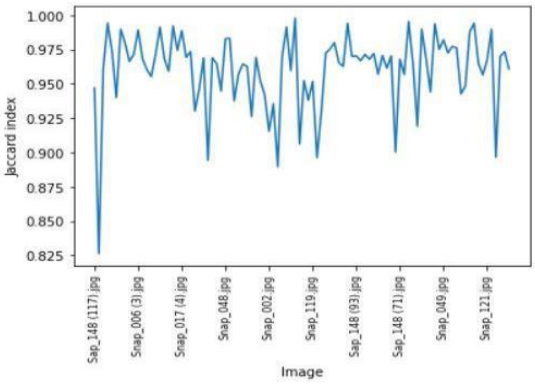

Fig 12. Jaccard Analysis with Proposed System

V. COMPARISON WITH EXISTING SYSTEM

A. TABLE III – COMPARISON ANALYSIS

S. NO.	TITLE	PROS	CONS
1	A Novel Type Fuzzy C-Means Clustering for Brain MR Image Segmentation.	An improved intuitionistic FCM (IIFCM) clustering approach uses the advantages of intuitionistic fuzzy set theory.	It the noisy pixels because it ignores the nearby pixels when calculating membership values.
2	Detection and Classification of Acute Lymphocytic Leukemia	Using principle components analysis, features are reduced for improved categorization..	It is difficult to remove the lymphocyte from the smear of peripheral blood.
3	A Review of Automated Methods for the Detection of Sickle Cell Disease	It focuses on handling issues with noise reduction, IIH correction, and segmentation of overlapping cells that are inherently problematic.	Independent of how photos contain spatial information
4	Segmentation of Overlapping Elliptical Objects in Silhouette Images	Using silhouette photos, extract a seed point from a collection of closely overlapping items.	If the estimated seed points do not exactly reflect the object centroids, the Euclidean distance may be unclear.
5	ElliFit: An unconstrained, non-iterative, least squares based geometric Ellipse Fitting method	Iterative non-linear optimisation is done using Ahn's method.	Local minima and high temporal complexity are issues it faces.

B. EXISTING ALGORITHMS ACCURACY

Algorithm	**Result**
AWSFCM	87%
GLRLM	92.28%
Two-RBCs cluster	98.08%
CECS- Feature Extraction	53.04%
Inception V4	89.16%

VI. CONCLUSION

In this process, the U_Net segmentation is applied and analyze the blood cell images. The blood cell coloured and ground truth image data's are taken as input data and applied into pre-processing method. Finally the classification method is used to segment the blood cell by comparing coloured and ground truth images. Deep learning

segmentation algorithm of ResNet50_Unet is implemented and predict the result based on accuracy, precision, jaccard, recall and f1-measure.

In conclusion, the use of the Unet model for the detection of hematological disorders has shown promising results. The Unet architecture, which utilizes a combination of convolutional and deconvolutional layers, has proven effective in accurately segmenting and identifying abnormal blood cells in medical images. The ability of the Unet model to effectively distinguish between healthy and abnormal blood cells can might assist in the early identification and detection of certain haematological illnesses. This could lead to earlier intervention and treatment, ultimately improving patient outcomes.While there is still room for further research and development, the use of the Unet model holds great potential in the field of medical imaging and the detection of hematological disorders. Its accuracy and efficiency make it a valuable tool for medical professionals seeking to improve the diagnosis and treatment of these diseases.

VII. FUTURE WORK

While Unet has shown promising results for the detection of hematological disorders, there are still some areas that require further research and development. Some possible future works include:

1. Improving the network design: Notwithstanding Unet's success in detecting haematological illnesses, there is always opportunity for network architecture advancement. Future studies might concentrate on creating more sophisticated network designs to boost the precision and effectiveness of the detection process..
2. Dataset expansion: One limitation of using Unet for the detection of hematological disorders is the availability of large and diverse datasets. Future work can focus on expanding the datasets used for training and validation to include more diverse blood samples and disorders, thereby improving the generalizability of the network.
3. Integration with clinical workflows: For the use of Unet to be more effective in clinical practice, it needs to be integrated with existing clinical workflows. Future work can focus on developing tools and interfaces that can seamlessly integrate Unet with existing clinical workflows and make it more accessible to healthcare professionals.
4. Exploration of other deep learning techniques: While Unet has shown promising results, there are other deep learning techniques that can be explored for the detection of hematological disorders. Future research can compare the effectiveness of various deep learning approaches to see which ones are best for this application.

Overall, the use of Unet for the detection of hematological disorders has the potential to significantly improve the diagnosis and treatment of these disorders. Further research and development in this area can lead to more accurate and efficient detection methods and ultimately better outcomes for patients.

REFERENCES

[1] Mishro, Pranaba K., Sanjay Agrawal, Rutuparna Panda, and Ajith Abraham. "A novel type-2 fuzzy C-means clustering for brain MR image segmentation." IEEE Transactions on Cybernetics 51, no. 8 (2020): 3901-3912 .

[2] Das, Pradeep Kumar, Priyanka Jadoun, and Sukadev Meher. "Detection and classification of acute lymphocytic

leukemia." In 2020 IEEEHYDCON, pp. 1-5. IEEE, 2020.

[3] Khan, Siraj, Muhammad Sajjad, Tanveer Hussain, Amin Ullah, and Ali Shariq Imran. "A review on traditional machine learning and deep learning models for WBCs classification in blood smear images." IEEE Access 9 (2020):10657- 10673.

[4] Zafari, Sahar, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. "Segmentation of overlapping elliptical objects in silhouette images." IEEE Transactions on Image Processing 24, no. 12 (2015): 5942-5952

[5] Prasad, Dilip K., Maylor KH Leung, and Chai Quek. "ElliFit: An unconstrained, non-iterative, least squares based geometric Ellipse Fitting method." Pattern Recognition 46, no. 5 (2013): 1449-1465.

[6] Sahu, Menika, Amit Kumar Biswas, and K. Uma. "Detection of Sickle Cell Anemia in Red Blood Cell." International Journal of Engineering and Applied Sciences (IJEAS) 2, no. 3 (2015): 45-48

[7]Li, Gang, Chengjie Sun, Chuanyun Xu, Yu Zheng, and Keya Wang. "Cervical Cell Segmentation Method Based on Global Dependency and Local Attention." Applied Sciences 12, no. 15 (2022): 7742.

[8] Gonzalez-Hidalgo, Manuel, F. A. Guerrero- Pena, Silena HeroldGarcía, Antoni Jaume-i- Capó, and Pedro D. Marrero-Fernández. "Red blood cell clusterseparation from digital images for use in sickle cell disease." IEEE journal of biomedical and health informatics 19, no. 4 (2014): 1514-1525.

[9] Chen, Siyuan, and Julien Epps. "Efficient and robust pupil size and blink estimation from near- field videosequences for human–machine interaction." IEEE transactions on cybernetics 44, no. 12(2014): 2356-2367.

[10]Liem, Robert I., Maurice R. O'Gorman, and Deborah L. Brown. "Effect of red cell exchange transfusion on plasma levels of inflammatory mediators in sickle cell patients with acute chest syndrome." American journal of hematology 76, no. 1(2004): 19-25.

[11] Meena Prakash, R., and R. Shantha Selva Kumari. "Fuzzy C means integrated with spatial information and contrast enhancement for segmentation of MR brain images." International Journal of Imaging Systems and Technology 26, no.2 (2016): 116-123.

[12] Prasad, Dilip K., Maylor KH Leung, and Chai Quek. "ElliFit: An unconstrained, non-iterative, least squares based geometric Ellipse Fitting method." Pattern Recognition 46, no. 5 (2013): 1449-1465.

[13]Zafari, Sahar, Mariia Murashkina, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. "Resolving overlapping convex objects in silhouette images by concavity analysis and Gaussian process." Journal of Visual Communication and Image Representation 73 (2020): 102962.

[14] Agrawal, Sanjay, Rutuparna Panda, and Ajith Abraham. "A novel diagonal class entropy-based multilevel image

thresholding using coral reef optimization." IEEE Transactions on Systems, Man, and Cybernetics: Systems 50, no. 11 (2018): 4688- 4696.

[15] Das, Pradeep Kumar, Sukadev Meher, Rutuparna Panda, and Ajith Abraham. "A review of automated methods for the detection of sickle cell disease." IEEE reviews in biomedical engineering 13 (2019): 309- 324..

[16] Meng, Cai, Zhaoxi Li, Xiangzhi Bai, and Fugen Zhou. "Arc adjacency matrix-based fast ellipse detection." IEEE Transactions on Image Processing 29 (2020): 4406-4420..

[17] Li, Jing, Yi Wang, Baiying Lei, Jie-Zhi Cheng, Jing Qin, Tianfu Wang, Shengli Li, and Dong Ni. "Automatic fetal head circumference measurement inultrasound using random forest and fast ellipse fitting." IEEE journal of biomedical and health informatics 22, no. 1 (2017): 215-223.

[18] Gonzalez-Hidalgo, Manuel, F. A. Guerrero- Pena, Silena Herold- García, Antoni Jaume-i- Capó, andPedro D. Marrero-Fernández. "Red blood cell cluster separation from digital images for use in sickle cell disease." IEEE journal of biomedical and health informatics 19, no. 4 (2014): 1514-1525.

[19] Azhar, Faisal, and Tardi Tjahjadi. "Significant body point labeling and tracking." IEEE transactions on cybernetics 44, no. 9 (2014): 1673-1685.

[20] Winter, Mark, Walter Mankowski, Eric Wait, Edgar Cardenas De La Hoz, Angeline Aguinaldo, and Andrew R. Cohen. "Separating touching cells using pixel replicated elliptical shape models." IEEE transactions on medical imaging 38, no. 4 (2018): 883-893.

[21] Chen, Siyuan, and Julien Epps. "Efficient and robust pupil size and blink estimation from nearfield video sequences for human–machine interaction." IEEE transactions on cybernetics 44, no. 12 (2014): 2356-2367.

[22] Panagiotakis, Costas, and Antonis A. Argyros. "Cell segmentation via region-based ellipse fitting." In 2018 25th IEEE International Conference on Image Processing (ICIP), pp. 2426-2430. IEEE, 2018..

[23] Sheng, Bin, Ping Li, Shuangjia Mo, Huating Li, Xuhong Hou, Qiang Wu, Jing Qin, Ruogu Fang, and David Dagan Feng. "Retinal vessel segmentation using minimum spanning superpixel tree detector." IEEE transactions on cybernetics 49, no. 7 (2018): 2707-2719

[24] Tareef, Afaf, Yang Song, Heng Huang, Dagan Feng, Mei Chen, Yue Wang, and Weidong Cai. "Multi-pass fast watershed for accurate segmentation of overlapping cervical cells." IEEE transactions on medical imaging 37, no. 9 (2018): 2044-2059.

[25] Bai, Xiangzhi, Yuxuan Zhang, Haonan Liu, and Zhiguo Chen. "Similarity measure- based possibilistic FCM with label information for brain MRI segmentation." IEEE transactions on cybernetics 49, no. 7 (2018): 2618-2630.