Received: 16 March 2023 | Revised: 19 April 2023 | Accepted: 26 April 2023 | Published online: 27 April 2023

RESEARCH ARTICLE

Artificial Intelligence and Applications
2023, Vol. 1(4) 252–259
DOI: 10.47852/bonviewAIA3202853

BON VIEW PUBLISHING

Comparison of Deep Learning Techniques in Detection of Sickle Cell Disease

Kawuma Simon[1,*], Mabirizi Vicent[2,*], Kyarisiima Addah[3], David Bamutura[4], Barnabas Atwiine[5], Deborah Nanjebe[6] and Adolf Oyesigye Mukama[7]

[1]*Department of Software Engineering, Mbarara University of Science and Technology, Uganda*

[2]*Department of Information Technology, Kabale University, Uganda*

[3]*Department of Electrical Engineering, Mbarara University of Science and Technology, Uganda*

[4]*Department of Computer Science, Mbarara University of Science and Technology, Uganda*

[5]*Department of Paediatrics and Child Health, Mbarara University of Science and Technology, Uganda*

[6]*Department of Medical Laboratory Science, Epicentre Mbarara Research Centre, Uganda*

[7]*Department of Medical Laboratory Science, Mbarara University of Science and Technology, Uganda*

Abstract: Recently, transfer learning technique has proved to be powerful in enhancing development of deep learning methods for sickle cell disease (SCD) detection as a complement to the clinical method where a hemoglobin electrophoresis machine is used. This is evidenced by a number of models and algorithms with ≥90% prediction accuracy. From literature, most of the proposed methods are trained and tested on pre-trained deep learning models like VGG16, VGG19, ResNet, Inception_V3 and ReNet. However, training and testing of these methods are limited on one model and separate dataset which may lead to biased results due to implementation in variation of these models which affects results produced. To this end, there exists a need to evaluate the SCD models using the same dataset. Thus, in this research study, we carried out a comparative investigation and evaluated predominate pre-trained models used to detect SCD using the same dataset to ascertain which one has the best accuracy. We used secondary dataset obtained from an online dataset. In our study, we have discovered that Inception V3 yielded the highest accuracy of 97.3% followed by VGG19 at 97.0%, VGG16 at 91%, ResNet50 at 82% and ReNet at 67%, and the CNN-scratch model achieved 81% accuracy. Results from our study will aid researchers and industry practitioners to make decision on the best deep learning model to use while detecting SCD.

Keywords: deep learning, techniques, models, sickle cell disease, detection

1. Introduction

Blood is an important fluid in human body with the red blood cell (RBC) being the prime components which contains the hemoglobin responsible for gaseous exchange. The normal shape of a RBC is biconcave, its size is between 6.8 and 7.8 μm in diameter and between 2 and 2.5 μm in thickness (Walker et al., 1990); however, due to certain disorders such as sickle cell, this shape can be deformed into an ovalocytes (C shape), thus resulting into sickle cell anemia (SCA) (Breakey et al., 2017). This abnormal shape makes it difficult for sickle cells to move through the bloodstream, hence decreasing the oxygen flow (Alzubaidi et al., 2020). Sickle cell anemia is a blood disorder characterized by structurally abnormal hemoglobin which deforms the RBCs into an ovalocyte (Breakey et al., 2017; Rakshit & Bhowmik, 2013). Today, millions of people around the globe have inherited the sickle cell mutation and those carriers of the trait have more than 300,000 children born each year with SCA (Hernandez et al., 2021). In Africa, 75% of the annual global SCA births occur in sub-Saharan Africa where there are currently no institutional newborn screening programs, resulting in children going undetected and dying at a very young age (Hernandez et al., 2021). This current statistic reveals that burden of sickle cell diseases (SCDs) has increased recently as compared to previous years where over 220,000 children were born with sickle cell anemia worldwide (Green et al., 2016), and these statistics are expected to increase by 50% by 2050 if no intervention is put in place (Chy & Rahaman, 2018). 50% of the children living with sickle cells in low-income countries are anticipated to die due to lack of diagnosis and treatment (Yang et al., 2013).

These rapidly increasing statistics have attracted a global concern to advocate for early diagnosis of SCA in high-burden countries. In recent years, several techniques for diagnosing SCA have been used (Alzubaidi et al., 2020). The manual

*Corresponding author: Kawuma Simon, Department of Software Engineering, Mbarara University of Science and Technology, Uganda. Email: simon.kawuma@must.ac.ug; Mabirizi Vicent, Department of Information Technology, Kabale University, Uganda. Email: vmabirizi@kab.ac.ug

technique for examining the blood smear has heavily relied on a pathologist's skills. It is a comprehensive and tiresome task. On the other hand, application of Artificial Intelligence specifically deep learning techniques is among the recent alternative technologies to provide faster, cheap and more accurate diagnosis of SCA (Kumar et al., 2022) as compared to the clinical hemoglobin electrophoresis machine which is too expensive to be enrolled in every hospital (Vicent et al., 2022). As a result, a number of models and algorithms for SCD detection have been developed. For example, many researchers have utilized pre-trained and fine-tuned the model to detect SCA, such as VGG16, VGG19, ResNet, Inception_V3 and ReNet. However, from literature, all these models are trained and tested on separate and individual datasets whereby each study claims having the best accuracy in detecting SCA; thus, there exist a need to evaluate these models on same dataset. In this research study, we carried out a comparative investigation on predominate deep learning techniques used to detect SCD to evaluate their accuracy on the same dataset.

2. Common Deep Network Architectures

In the previous section, we listed some of the deep networks that have been widely applied to diagnose sickle cells; in this section, we present a detail review of these networks.

2.1. VGG

The Visual Geometry Group (VGG) is a convolutional neural network (CNN) developed by VGG from University of Oxford (Simonyan & Zisserman, 2014). They developed several models including the VGG16 which is their first model released in 2013 (Garcia-Garcia et al., 2017). The model was named VGG16 due to the fact that it is composed of 16 weight layers. The model becomes popular after its submission to the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) in 2013. The model achieved 92.7% accuracy. The success for VGG16 paved way for VGG19 which was released in 2014 (Shaha & Pawar, 2018). The model comprises of 19 layers including 16 convolution layers and 3 fully connected layers.

2.2. ResNet

ResNet also known as Residual is well known for its depth (152 layers), and the model won the ILSVRC-2016 with 96.4% test accuracy (He et al., 2016). The model was introduced by Microsoft in 2015. In order to solve complex problem, there are additional layers stacked in deep neural network which improves accuracy and performance. The reason behind additional layers in ResNet is that these layers progressively learn more complex features for example during image processing, the first layer may detect the edge, the second layer detects the texture, and the next layer detects the object. This improves the performance accuracy of the model. During model training, the model depth may be reduced; hence, models such as ResNet-50 have been introduced.

2.3. Inception

Inception also known as GoogleNet was invented by Szegedy et al. (2015). Compared to VGG, Inception achieved 93.3% accuracy and it won the 2014 ILSVRC challenge of Top-5 test accuracy (Garcia-Garcia et al., 2017). It is composed of 22 layers and a building block known as "inception module." The modules in inception consist of Network in Network layer, a large-sized convolution layer, small-sized convolution layer and pooling layer, thus creating a provision for stacking CNN in multiple ways. The computation of these layers is parallel in nature with a 1×1 convolution operation which reduces dimensionality.

2.4. ReNet

Graves et al. (2007) proposed a Multi-dimensional Recurrent Neural Network (MDRNN) architecture to extend the Recurrent Neural Network (RNN). In MDRNN, each dimension replaces a single recurrent connection in RNN with d connection where d is the number of spatio-temporal data dimensions. It is upon this setup that (Visin et al., 2015) built ReNet architecture where sequence RNN replaces the multidimensional RNN. In ReNet, the number of RNNs increases linearly at each layer based on the number of input dimensions.

2.5. CNN-Scratch

Besides pre-trained models, in this study, we developed a CNN from scratch using the following CNN layers discussed below;

Convolutional layer: this is the core building block used in CNN. This layer consists of a number of filters commonly known as kernels used for extracting bio-markers (features) from the input image. To improve feature extraction accuracy, Rectified Linear Unit (ReLu) was used to activate neuron (nodes) through which data and computation flow.

Pooling layer: To avoid pixelated and blurry-looking output images, we applied downsampling to decrease the size of the feature matrix obtained after passing the input image through the convolution layer. This was achieved by passing a filter over the results of the convolution layer selecting one number out of each group of values.

Fully connected layer: also known as linear layer applies a linear transformation to the input vector via a weight matrix. Thus, all possible layer-to-layer connections are present; hence, every input vector influences every output vector.

The developed model contains a total of three blocks. In the first block, we applied three convolution layers each having 32 filters, 3×3 in size and a ReLu function. This is followed by max pooling layer with 64 filters, 3×3 in size and a pool size of (2, 2) for downsampling. The third block is similar to the second block, only that 128 filters are used in the third block. To cater for overfitting, we utilized dropout technique with a threshold of 0.2 at each block. To generalize results, the threshold value was adjusted to 0.5. Finally, we applied a dense layer to minimize the vector height from 64 to 2 elements. The output of the developed model is a binary classification; that is, an image is either classified as sickle or non-sickle cell (normal).

3. Literature Review

In this section, we discuss various deep learning techniques, which have been efficiently used in detection of SCD and other diseases. Deep learning methods like deep CNNs (Xu et al., 2017) and RNNs (Breakey et al., 2017) have been used to detect SCD. Das et al. (2019) provide a detailed methodological review about deep learning technique and tools used to detect SCD. Vicent et al. (2022) developed an algorithm to detect presence of sickle cells in overlapping RBCs. In their method, canny edge and double threshold machine learning techniques were used to separate overlapping cells of digital blood smears. From the results, the algorithm achieved 98.18% accuracy automation detection of

overlapping RBCs for sickle cell diagnosis. In Kiruthika et al. (2022), authors automated detection of sickle cells by image splitting, to achieve their objective, watershed segmentation techniques was applied on RBCs to extract parameters used for sickle cells analysis. Begum et al. (2018) proposed a method for classifying sickle cells using image processing techniques; in this study, authors used Otsu thresholding to segment microscopic images of RBCs, whereas Naïve Bayes, random forest support vector machine and logistic regression were used for model training and testing. Alzubaidi et al. (2020) applied deep learning techniques to detect SCA from microscopic RBC images. The model achieved 99.98% classification accuracy. Despite the accuracy achieved by these methods in detecting sickle cells, training and testing of these methods are limited on one model and separate dataset which may lead to biased results due to implementation in variation of these models which affects results produced.

Furthermore, deep learning techniques have been widely applied in medical imaging; for example, Nahid et al. (2020) applied multichannel CNN to detect radiographs for pneumonia diagnosis. In their work, VGG16 and VGG19 pre-trained models were used to train their model. In a study by Daoud et al. (2020) deep learning techniques were applied to classify ultrasound breast tumor images to detect breast cancer. The model achieved 96.1% detection accurate. This clearly implies the model could possibly detect breast cancer by image processing using ultrasound images. Yang et al. (2021) proposed a method that utilizes peripheral blood images to diagnose acute leukemia. In their method, color clustering, mathematical morphology and image segmentation were performed, whereas CNNs were used to classify the cells. The model achieved 85.8% classification accuracy and 94% diagnosis accuracy (Yang et al., 2021). Kasani et al. (2020) applied VGG19 and NASNetLarge in their proposed method for classifying leukemic B-lymphoblast cells and B-lymphoid precursor cells; the model diagnosis accuracy was 96.58%.

Besides, transfer learning methods have registered success in medical image classification; for example recently, Alshazly et al. (2021) utilized transfer learning to train a model for COVID-19 diagnosis using chest CT images, and the model achieved 92.9% accuracy on COVID-19-CT dataset. A model for classifying clostridioides difficile bacteria cytotoxicity was trained using transfer learning and achieved 93.5% accuracy on 369 images (El-Khatib et al., 2020). Models used in all these studies are based on deep learning techniques where majority apply CNNs. However, all the models were trained and tested on separate dataset and each model claims to have the best accuracy. In this research study, we evaluate commonly used deep learning models used to detect sickle cell on the same dataset.

4. Research Methodology

4.1. Dataset acquisition

We used secondary dataset obtained from an online dataset for digitized thin blood films of SCD detection; these images were collected by Manescu et al. (2023). These images were collected using a custom-built brightfield microscope fitted with a 100X/ 1.4NA objective lens, a color camera and a motorized x–y sample positioning stage. To improve image quality, z-stacks were projected onto a single (xy) plane using a Wavelet-based Extended Depth of Field algorithm. The prepared images were then exposed to a hemoglobin electrophoresis to obtain hemoglobin phenotype and test patient for SCD. This dataset was collected with support from UK Research Council and College of Medicine of the University of Ibadan, Nigeria. This dataset is publicly available: https://doi.org/10.5522/04/12407567

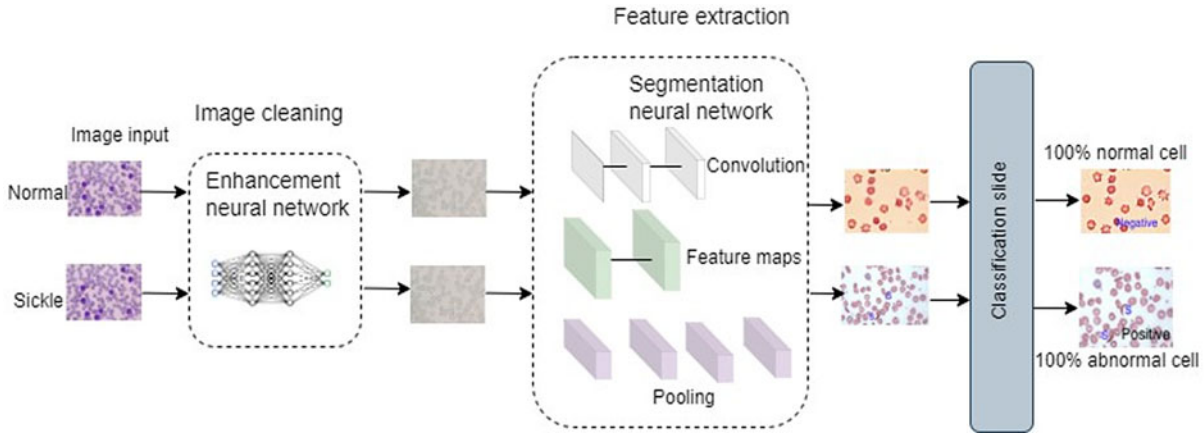
4.2. Ethical consideration

We did not consider ethical approval since the collection process of the secondary dataset used in this study met international ethics obtained from ethics committee at the Institute for Advanced Medical Research and Training (IAMRAT) of College of Medicine, University of Ibadan.

4.3. Data pre-processing

The blood films had variation (noise), and these variations are as a result of unwanted cells like white blood cells, uncontrolled light intensity, and mechanical shifts of the microscope used in taking the photos. Therefore, enhancement neural network as shown in Figure 1 was applied to standardize images and improve their quality in terms of spatial and spectral features, hence transforming them to look closely similar to those taken using benchtop microscope. As discussed earlier, dropout technique with a threshold of 0.2

Figure 1
Model development activities



at each block was used to eliminate overfitting. During data cleaning, a total of 678 images were dropped by the filter due to weak pixels; hence, they could not be standardized. The remaining 4322 images were divided into two subsets, that is, 3000 images for model training and 1322 for testing.

4.4. Image segmentation

The cleaned blood films were exposed to a segmentation neural network with two layers, convolutional layer, and pooling layer, as shown in Figure 1 to extract sickle cell screening features. In each layer, we used a total of 32 filters, each with 3×3 dimensions. The convolutional layer receives the cleaned image from enhancement neural network as input to perform semantic segmentation of the blood cells. The segmentation network determines the number of normal and sickle cells within each image. The newly formed image was subjected to the pooling layer with a pool size of (2×2) and stride of 2 pixels which reduced the number of trainable parameters while maintaining the image quality (composition). To achieve this, we reduced the image size to 64×64 pixels by focusing on larger areas of the input parameter.

4.5. Classification

To classify RBC slide films into sickle and normal cells, we used CNN-based transfer learning method with different pre-trained models including; VGG16, VGG19, Inception V3, ResNet50 and ReNet. Furthermore, we also applied classification by CNN-scratch method, this was done in five steps as follows:

(1) Data loading and normalization: we first defined the transformation to apply to the training and testing dataset; in this case, we used 3D transformation. The images were then converted from its original format into a tensor that was used with the torch library to normalize the data. We fine-tuned the images by splitting it into batches to enhance constant learning and enhance the model.
(2) Defining filters to use: A total of 32, 64 and 128 filters were used in the 1st, 2rd and 3rd block, respectively to smoothen, sharpen and enhance the dataset.
(3) Loss function and optimizer definition: Cross-entropy Loss from PyTorch was used to calculate the loss during training. This was done through combining the log softmax and negative log likelihood. The softmax was used to scale the classification numbers into probability for each outcome, and the negative log likelihood was used in tandem with softmax to calculate the loss based on the range of its function and PyTorch's optimizer was used for optimizing.
(4) Model training: The model was then trained on a total of 3,000 images using varying epochs; that is, the epochs scale was shifted to improve model accuracy on training dataset.
(5) Model testing: Step 4 was reported on testing dataset to assess model accuracy. Model performance for the different proposed pre-trained models and training from scratch method was evaluated and compared using four performance metrics, including precision, recall, accuracy and F1 score, as shown in equations (1–4).

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + \mathrm{F}_p + F_n} \quad (1)$$

$$Precision = \frac{T_p}{T_p + \mathrm{F}_p} \quad (2)$$

$$Recall = \frac{T_p}{T_p + F_n} \quad (3)$$

$$F1 - score = 2\left(\frac{Precision * Recall}{Precision + Recall}\right) \quad (4)$$

Where:

T_p is true-positive value, T_n is true-negative value, F_p is false-positive value, F_n is false-negative value.

The proposed models were trained on a total of 3000 images including 1730 images of sickle cells and 1270 images of non-sickle cells. We used cross-entropy function and PyTorch's optimizer to reduce the dimensions of the extracted features. Training was done in the form of batch processing using a batch of 100 and 36 epochs. To test the proposed model, we used a total of 1322 images. Both model training and testing were done using Python library for training, running on a standard laptop with 32 GB RAM, 3.1GHz processor speed with a GPU.

5. Results and Discussion

The curves in Figures (2–13) and Tables 1 and 2 represent model performance on both training and testing dataset for VGG16, VGG19, ResNet50, Inception V3, CNN-scratch as shown below.

Table 1 presents the performance metrics on testing dataset for each of the proposed models for detecting sickle cells. As observed in Table 1, Inception version achieved the highest performance with 100%. The model loss on training and testing dataset was 0.0016 and 0.4721, respectively. VGG19 achieved 99% accuracy with 0.97 recall, F1-score of 0.98, and the model loss was 0.082. Among other pre-trained models, VGG19 achieved better results with 97% accuracy, followed by ResNet50 with 82% accuracy, and ReNet had the worst performance on training dataset with 69% accuracy and 0.66 training loss. 0.98, and the model loss was 0.082. Among other pre-trained models, VGG19 achieved better results with 97% accuracy, followed by ResNet50 with 82% accuracy and ReNet had the worst performance on training dataset with 67% accuracy and 0.66 training loss. CNN-scratch

Figure 2
VGG16 model accuracy

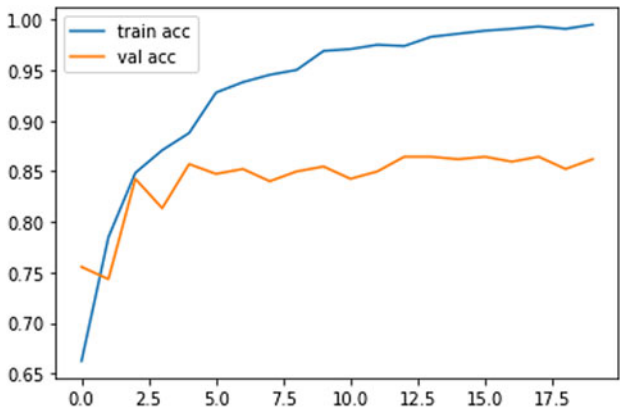


Figure 3
VGG16 model loss

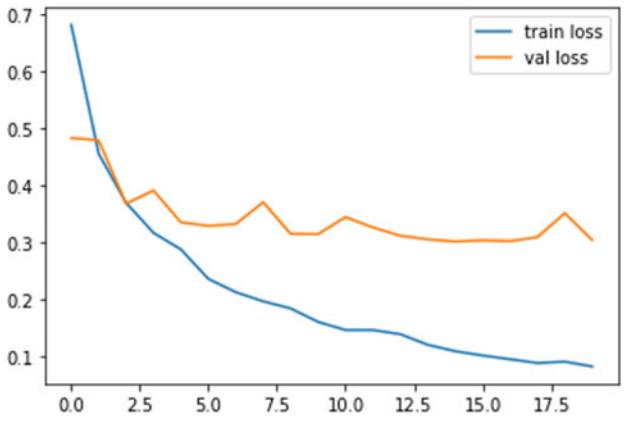


Figure 4
VGG19 model accuracy

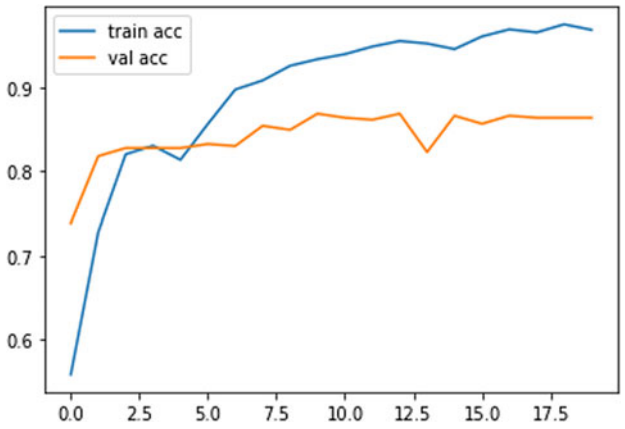


Figure 5
VGG19 model loss

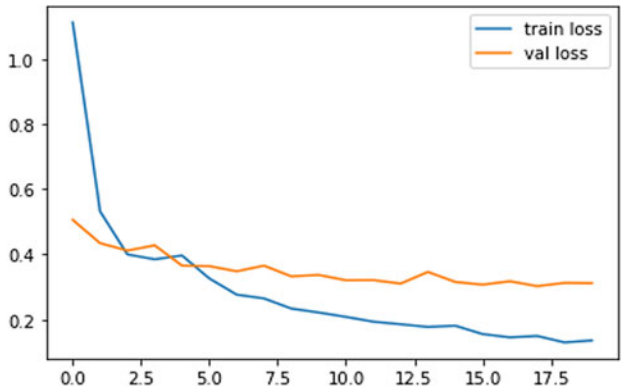


Figure 6
ResNet50 accuracy

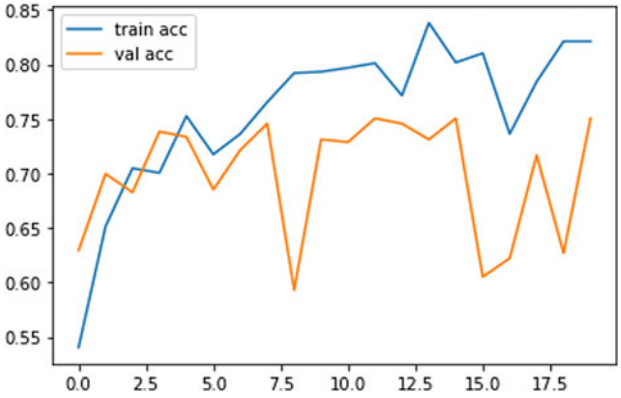


Figure 7
ResNet50 model loss

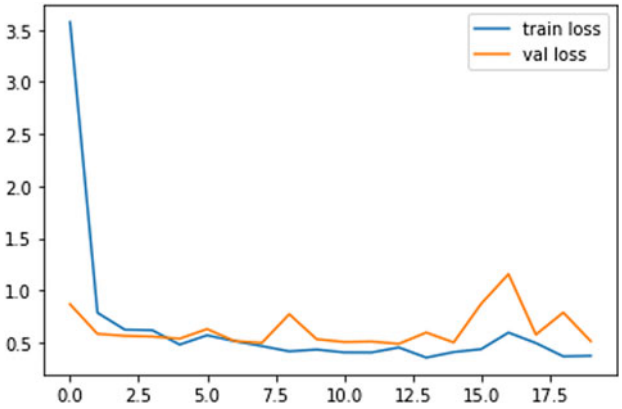


Figure 8
Inception V3 model accuracy

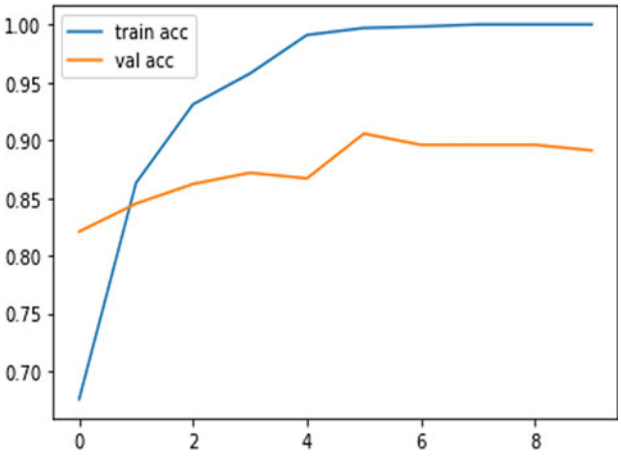


Figure 9
Inception V3 model loss

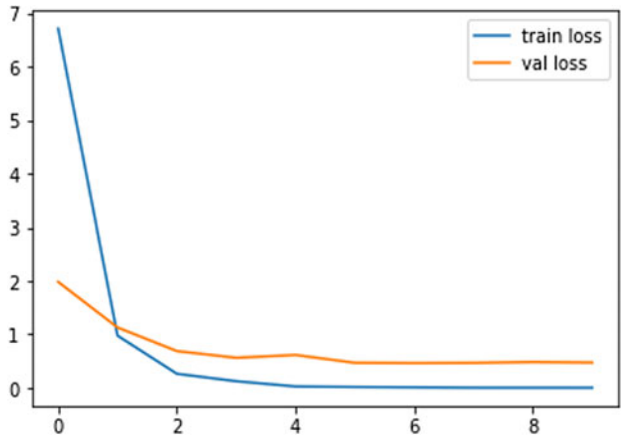


Figure 10
Scratch model accuracy

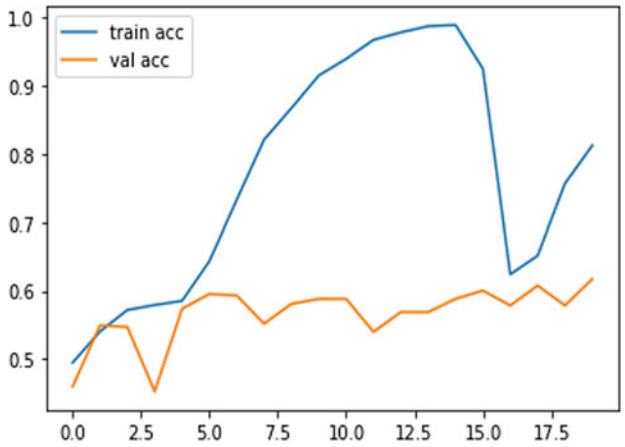


Figure 11
Scratch model loss

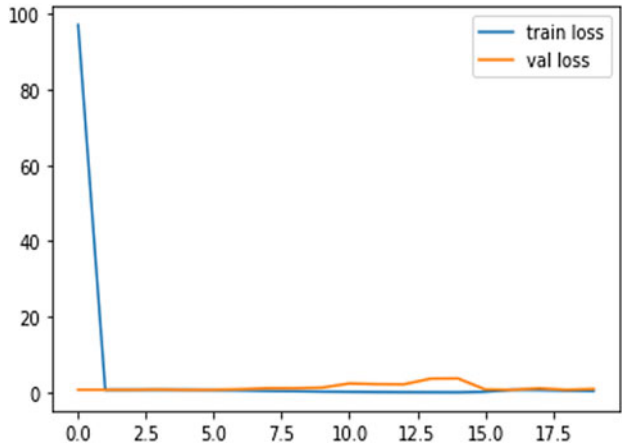


Figure 12
ReNet model accuracy

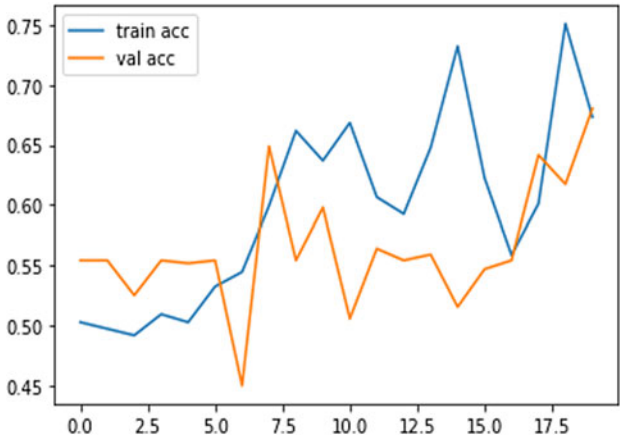


Figure 13
ReNet model loss

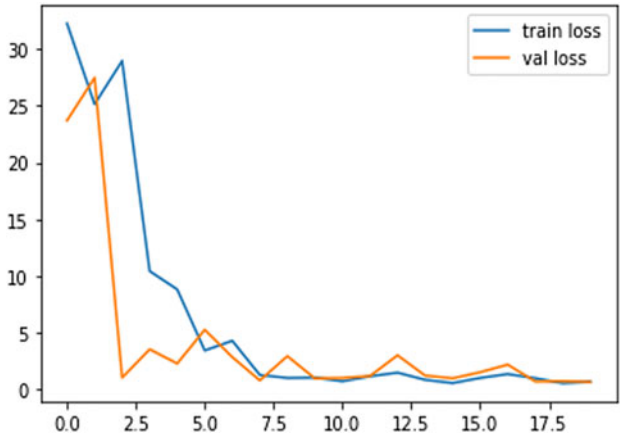


Table 1
Model performance on training dataset

Models	Precision	Recall	F1-score	Accuracy	Training loss
VGG16	0.99	0.97	0.98	0.99	0.082
VGG19	0.97	0.97	0.97	0.97	0.14
ResNet50	0.82	0.82	0.82	0.82	0.37
Inception v3	1.00	1.00	1.00	1.00	0.0016
Scratch	0.81	0.81	0.81	0.81	0.39
ReNet	0.67	0.67	0.67	0.67	0.66

model, that is, training from scratch, achieves 81% precision, 81% recall, 81% F1-score, 81% accuracy and 0.39 training loss which is better than ReNet and slightly less than the pre-trained models. This results can be attributed to the less dataset used to train the model due to the fact that training from scratch needs a large dataset compared to the pre-trained models.

Table 2 presents model performance for the above-mentioned pre-trained models and the proposed model (CNN-scratch). For the training dataset, Inception V3 and VGG19 yield a similar performance with 97% accuracy, 97% precision, 97% recall, 97% F1-score, and the testing loss was 0.14, followed by VGG16 with 91% accuracy, 91% precision and 91% recall. The F1-score for

Table 2
Model performance on testing dataset

Models	Precision	Recall	F1-score	Accuracy	Training loss
VGG16	0.91	0.91	0.91	0.91	0.29
VGG19	0.97	0.97	0.97	0.97	0.14
ResNet50	0.75	0.75	0.75	0.75	0.51
Inception v3	0.97	0.97	0.97	0.97	0.14
Scratch	0.81	0.81	0.81	0.81	0.39
ReNet	0.68	0.68	0.68	0.68	0.65

VGG16 was 91%, whereas the model has a 0.29 testing loss as observed in Table 2 and Figure 3. ResNet50 yielded better results with 75% accuracy and 75% recall, 75% precision and 75% F1-score as compared to ReNet which had poor performance with 68% accuracy and 0.65 testing loss. CNN-scratch achieved similar results 81% for all performance metrics used as observed in Tables 1 and 2.

Comparing VGG models, VGG19 yielded fairly better results compared to VGG16. The VGG19 architecture has three more weight layers as compared to VGG16 which facilitates faster training speed, fewer training samples per time and improved accuracy, unlike VGG16 which requires more training time and a bigger training dataset to increase accuracy. The two models were trained using the same size of the dataset and for the same period of time. Thus, increasing the dataset and training time for VGG16 can improve its performance. In both models, a total of 32 filters were used, reducing filters would also increase performance in both cases since the higher the number of filters the higher the number of abstraction to the network which reduces its performance in extracting features from the image.

ResNet50 and ReNet networks require much more dataset than any other traditional deep learning models; thus, the poor performance of the two networks is attributed to the size of the dataset used in this study and can be improved by training the model for a longer time using a bigger dataset. Inception v3 achieved a better performance compared to other networks on both training and testing dataset. This is due to it's inception blocks capable of convolving same input tensor with multiple filters hence facilitating easier and faster extraction of features in a shorter training time.

The concept of transfer learning has been widely applied in the detection of sickle cells; for example, a study by de Haan et al. (2020) utilized deep learning techniques in their framework for detecting sickle cell using blood smears taken with smartphone microscope. The framework achieved 98% accuracy; also, Vicent et al. (2022) reported 98.18% accuracy automation detection of overlapping RBCs for sickle cell diagnosis. Arishi et al. (2021) conducted a review study on current and emerging techniques for SCD detection and highlighted potential methods for early diagnosis of SCD. In their findings, technologies such as coupling solubility tests with portable devices, smartphone-based microscopic classifications, image processing techniques, rapid immunoassays and sensor-based platforms were sported out. Each of these platforms utilizes deep learning transfer learning techniques.

However, these studies are either limited to one or few pre-trained models or utilized training from scratch method; for example, de Haan et al. (2020) utilized scratch method. In this study, we have considered four different CNN pre-trained models, ReNet and training from scratch method and evaluated these models using the same dataset. Thus, accommodating for a wider range of potential current deep learning techniques for SCD detection.

6. Conclusion

Authors of this paper believe that this is the first of its kind to conduct comparative study on performance of deep networks used in detection of SCD on the same dataset. A number of models and algorithm for SCD detection have been proposed; however, the scope has been limited to one or two deep network(s). In this study, we performed an experiment where a total of five deep networks including VGG16, VGG19, Inception V3, RestNet and ReNet were compared in term of training and testing performance.

We discussed some of the commonly used deep networks for image processing; later, we reviewed some of the studies where these networks have been applied in the field of medical imaging. We then performed an experiment assessing the performance of these networks in detecting sickle cells disease using a same dataset. From results obtained, Inception V3 achieved the best performance in detecting SCD followed by VGG16, VGG19 and ResNet50. Whereas ReNet had the least performance amongst all pre-trained models. We also applied scratch method, where 26 filters were used to train and test the model. Model testing using scratch method achieved 81% accuracy. The dataset used in this study was obtained from an open online dataset for digital thin blood films of SCD detection. In conclusion, deep networks have proved to be powerful in SCD detection. This is evidenced by the emerging proposed methods with higher training and testing accuracy but still remains a gap in newborn sickle cell screening since currently there are no clinical programs for newborn screening. Similarly, no study has been carried out for application of deep learning to detect SCD in babies.

Although several deep learning techniques have been proposed as potential solution to ease detection of sickle cells by image processing, these methods rely on clinically prepared dataset; that is, the datasets used for model training and testing are obtained using compound light microscopes and clinically prepared. This limits the application of these methods especially in developing countries, where a number of rural-based health facilities do not have well-equipped science laboratories. Therefore, future studies can assess the possibility of applying deep learning techniques in detecting sickle cells using images taken using mobile smartphone camera that do not require clinical preparation stages.

Acknowledgements

The authors acknowledge the Uganda Government for funding this research through Mbarara University of Science and Technology. The authors in same way recognize Petru Manescu for availing the dataset used in this research study.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

References

Alshazly, H., Linse, C., Barth, E., & Martinetz, T. (2021). Explainable COVID-19 detection using chest CT scans and deep learning. *Sensors*, *21*(2), 455. https://doi.org/10.3390/s21020455

Alzubaidi, L., Fadhel, M. A., Al-Shamma, O., Zhang, J. L., & Duan, Y. (2020). Deep learning models for classification of red blood cells in microscopy images to aid in sickle cell anemia diagnosis. *Electronics*, *9*(3), 427. https://doi.org/10.3390/electronics9030427

Arishi, W. A., Alhadrami, H. A., & Zourob, M. (2021). Techniques for the detection of sickle cell disease: A review. *Micromachines*, *12*(5), 519. https://doi.org/10.3390/mi12050519

Begum, K., Mannan, M., Sanyal, M., Hosen, I., Chakraborty, S., & Shekhar, H. (2018). Molecular diagnostic approach prevails superior over conventional gel-electrophoresis method in detecting sickle cell anemia. *Journal of Molecular Biomarkers & Diagnosis*, *9*(382), 2. https://doi.org10.4172/2155-9929.1000382

Breakey, V. R., Harris, L., Davis, O., Agarwal, A., Ouellette, C., Akinnawo, E., & Stinson, J. (2017). The quality of information about sickle cell disease on the internet for youth. *Pediatr Blood Cancer*, *64*(4), e26309. https://doi.org/10.1002/pbc.26309

Chy, T. S., & Rahaman, M. A. (2018). Automatic sickle cell anemia detection using image processing technique. In *2018 International Conference on Advancement in Electrical and Electronic Engineering*, 1–4. https://doi.org/10.1109/ICAEEE.2018.8642984.

Daoud, M. I., Abdel-Rahman, S., Bdair, T. M., Al-Najar, M. S., Al-Hawari, F. H., & Alazrai, R. (2020). Breast tumor classification in ultrasound images using combined deep and handcrafted features. *Sensors*, *20*(23), 6838.

Das, P. K., Meher, S., Panda, R., & Abraham, A. J. (2019). A review of automated methods for the detection of sickle cell disease. *IEEE Reviews in Biomedical Engineering*, *13*, 309–324. https://doi.org/10.1109/RBME.2019.2917780

de Haan, K., Ceylan Koydemir, H., Rivenson, Y., Tseng, D., Van Dyne, E., Bakic, L., . . . , & Ozcan, A. (2020). Automated screening of sickle cells using a smartphone-based microscope and deep learning. *NPJ Digital Medicine*, *3*(1), 76. https://doi.org/10.1038/s41746-020-0282-y

El-Khatib, H., Popescu, D., & Ichim, L. J. S. (2020). Deep learning–based methods for automatic diagnosis of skin lesions. *Sensors*, *20*(6), 1753. https://doi.org/10.3390/s20061753

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., & Garcia-Rodriguez, J. (2017). A review on deep learning techniques applied to semantic segmentation. *arXiv Preprint: 1704.06857*.

Graves, A., Fernández, S., & Schmidhuber, J. (2007). Multi-dimensional recurrent neural networks. *International Conference on Artificial Neural Networks*, 549–558. https://doi.org/10.1007/978-3-540-74690-4_56

Green, N. S., Mathur, S., Kiguli, S., Makani, J., Fashakin, V., LaRussa, P., . . . , & Mupere, E. (2016). Family, community, and health system considerations for reducing the burden of pediatric sickle cell disease in Uganda through newborn screening. *Global Pediatric Health*, *3*, 2333794X16637767. https://doi.org/10.1177/2333794X16637767

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778

Hernandez, A. G., Kiyaga, C., Howard, T. A., Ssewanyana, I., Ndeezi, G., Aceng, J. R., & Health, I. (2021). Trends in sickle cell trait and disease screening in the Republic of Uganda, 2014–2019. *Tropical Medicine & International Health*, *26*(1), 23–32. https://doi.org/10.1111/tmi.13506

Kasani, P. H., Park, S. W., & Jang, J. W. (2020). An aggregated-based deep learning method for leukemic B-lymphoblast classification. *Diagnostics*, *10*(12), 1064. https://doi.org/10.3390/diagnostics10121064

Kiruthika, V., Vallikannu, A., & Vimalarani, G. (2022). Automatic sickle cell anaemia detection using image processing technique. In *International Conference on Micro-Electronics and Telecommunication Engineering*, 281–288. https://doi.org/10.1007/978-981-16-8721-1_27

Kumar, Y., Koul, A., Singla, R., & Ijaz, M. F. (2022). Artificial intelligence in disease diagnosis: A systematic literature review, synthesizing framework and future research agenda. *Journal of Ambient Intelligence and Humanized Computing*, *14*, 8459–8486. https://doi.org/10.1007/s12652-021-03612-z

Manescu, P., Narayanan, P., Bendkowski, C., Elmi, M., Claveau, R., Pawar, V., . . . , & Fernandez-Reyes, D. (2023). Detection of acute promyelocytic leukemia in peripheral blood and bone marrow with annotation-free deep learning. *Scientific Reports*, *13*(1), 2562. https://doi.org/10.1038/s41598-023-29160-4

Nahid, A. A., Sikder, N., Bairagi, A. K., Razzaque, M. A., Masud, M., Z. Kouzani, A., & Mahmud, M. P. (2020). A novel method to identify pneumonia through analyzing chest radiographs employing a multichannel convolutional neural network. *Sensors*, *20*(12), 3482. https://doi.org/10.3390/s20123482

Rakshit, P., & Bhowmik, K. (2013). Detection of abnormal findings in human RBC in diagnosing sickle cell anaemia using image processing. *Procedia Technology*, *10*, 28–36. https://doi.org/10.1016/j.protcy.2013.12.333

Shaha, M., & Pawar, M. (2018). Transfer learning for image classification. In *2018 IEEE Second International Conference on Electronics, Communication and Aerospace Technology*, 656–660. https://doi.org/10.1109/ICECA.2018.8474802

Simonyan, K., & Zisserman, A. J. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv Preprint: 1409.1556.*

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., . . . , & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9.

Vicent, M., Simon, K., & Yonasi, S. (2022). An algorithm to detect overlapping red blood cells for sickle cell disease diagnosis. *IET Image Processing*, *16*(6), 1669–1677. https://doi.org/10.1049/ipr2.12439

Visin, F., Kastner, K., Cho, K., Matteucci, M., Courville, A., & Bengio, Y. (2015). Renet: A recurrent neural network based alternative to convolutional networks. *arXiv Preprint:1505.00393.*

Walker, H. K., Hall, W. D., & Hurst, J. W. (1990). *Clinical methods: The history, physical, and laboratory examinations.* UK: Butterworth Publishers.

Xu, M., Papageorgiou, D. P., Abidi, S. Z., Dao, M., Zhao, H., & Karniadakis, G. E. (2017). A deep convolutional neural network for classification of red blood cells in sickle cell anemia. *PLoS Computational Biology*, *13*(10), e1005746. https://doi.org/10.1371/journal.pcbi.1005746

Yang, R., Du, Y., Weng, X., Chen, Z., Wang, S., & Liu, X. (2021). Automatic recognition of bladder tumours using deep learning technology and its clinical application. *International Journal of Medical Robotics and Computer Assisted Surgery*, *17*(2), e2194. https://doi.org/10.1002/rcs.2194

Yang, X., Kanter, J., Piety, N. Z., Benton, M. S., Vignes, S. M., & Shevkoplyas, S. S. (2013). A simple, rapid, low-cost diagnostic test for sickle cell disease. *Lab on a Chip*, *13*(8), 1464–1467. https://doi.org/10.1039/C3LC41302K

How to Cite: Simon, K., Vicent, M., Addah, K., Bamutura, D., Atwiine, B., Nanjebe, D., & Oyesigye Mukama, A. (2023). Comparison of Deep Learning Techniques in Detection of Sickle Cell Disease. *Artificial Intelligence and Applications*, *1*(4), 252–259, https://doi.org/10.47852/bonviewAIA3202853