HIGH SCHOOL EDITION
Journal of Student Research
Volume 13 Issue 1 (2024)

Advancing the Diagnosis of Acute Lymphoblastic Leukemia with Hybrid Neural Networks

Meghana Somu[1], Anika Ramanathan[2] and Karma Luitel[3]

[1]Monta Vista High School, USA
[2]Presentation High School, USA
[3]Cupertino High School, USA
[#]Advisor

ABSTRACT

Acute Lymphoblastic Leukemia (ALL) is a malignancy of B or T lymphoblasts characterized by the uncontrolled replication of abnormal cells in the blood. The difficulty in diagnosing ALL arises from its visual similarity with other cells (HEM) in the blood, and misdiagnosis rates are high with other diseases. Thus, we seek an alternate solution to help medical professionals with the diagnosis of ALL. Recent solutions have utilized hybrid models that have delivered better performance than prior models, so we utilize a hybrid architecture unused for ALL detection. Our proposed architecture consists of EfficientNetB0 as our feature extractor and XGBoost classifier for its gradient boosting framework to perform a binary classification between ALL and HEM cells. The model maintains an accuracy of 85% when diagnosing ALL, proving the effectiveness of a data-driven approach for diagnosis. In the future, our model could be trained with other ALL datasets in order to become more versatile. Our work can assist doctors in providing an accurate and efficient diagnosis of leukemia, allowing for early intervention of the disease.

Introduction

Acute Lymphoblastic Leukemia (ALL) is a cancer of the blood cells in which immature cells are produced in the bone marrow and develop into lymphoblasts (Mayo Clinic). ALL is diagnosed in about 4000 people in the United States each year with the majority of patients being under the age of 18, making it the most common malignancy of childhood (Puckett and Chan). The most common presenting symptoms of Acute Lymphocytic Leukemia are non-specific and may be difficult to distinguish from common diseases of childhood. Furthermore, there is a great visual similarity between lymphoblasts and typical (HEM) cells. This visual similarity is demonstrated in the comparison between an ALL and HEM cell in FIG 1. In many cases, leukemia can be misdiagnosed or not diagnosed at all because patients do not show any symptoms until the disease has progressed further (The Personal Injury Center). Despite improvements in supportive care, death resulting from treatment toxicity remains a challenge.

The use of artificial intelligence in the accurate detection of ALL has incredible potential. The field of AI in medical imagery has grown exponentially with the introduction of neural network architectures such as CNNs (Schmidhuber) and RNNs (He et al.), along with other algorithms such as XGBoost (Chen and Guestrin). Due to the difficulty of human visual classification, we employ the use of deep learning to approach this problem. With these advancements, it is now possible to build a model utilizing computer vision to provide an accurate diagnosis of ALL.

Many studies have already looked into the use of deep learning in classifying ALL versus HEM cells. Initial solutions in the diagnosis of ALL employed the use of standalone neural networks. One such study used the VGG16 (Simonyan and Zisserman) framework, due to its simplicity and shallowness to maintain features.

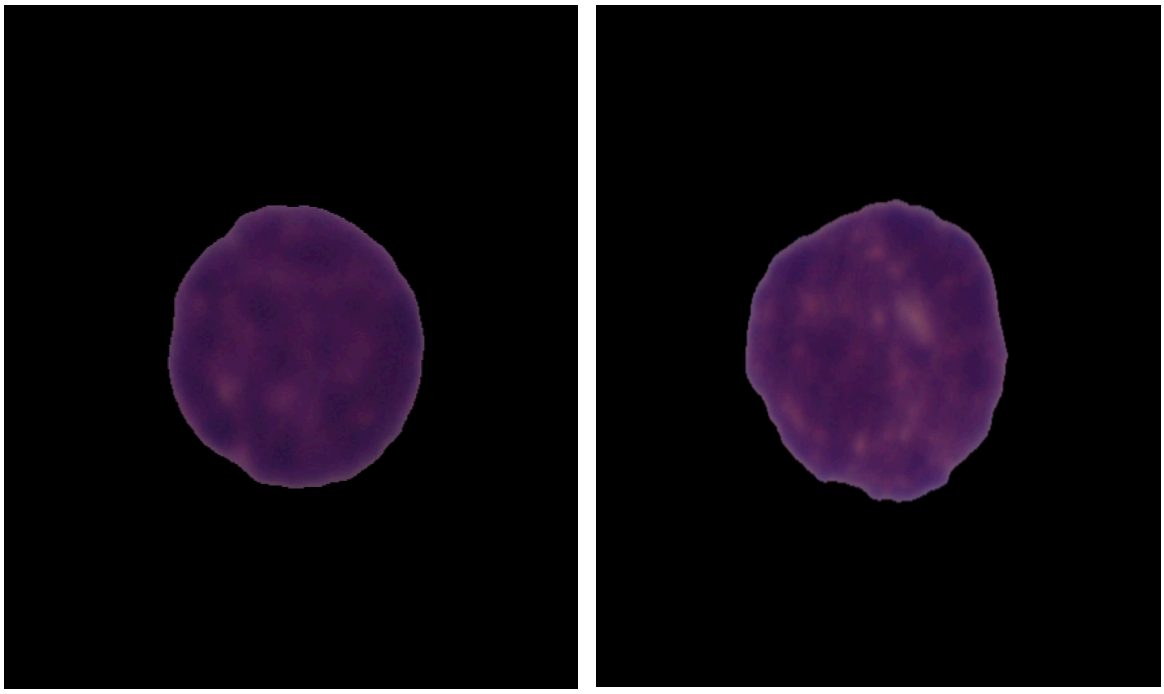
On top of VGG16, the study added an ECA module to amplify semantic features. They achieved the best accuracy of 0.911 with the VGG16+ECA model. (Ullah et al.). Another study tested VGG-16, ResNet50 (He et al.), and a proposed CNN against the dataset. Their proposed CNN had layers of 2d convolution, max pooling, and dropout, ending with one FC layer and an output layer with the two classes. Their testing accuracy included: 0.8163 (ResNet50), 0.8210 (Proposed CNN), and 0.8462 (VGG 16) (Rezayi et al.).

Other solutions have utilized concatenated networks. One paper sought to fix the imbalance in data between ALL and HEM cells, along with addressing the inter-class similarity through a modified bagging ensemble training strategy. With this strategy, both models learn minority features while learning specific majority features, accounting for the imbalance and helping distinguish between ALL and HEM cells. Their architecture consisted of two InceptionResNets concatenated with 2 ending FC layers to deliver a prediction, achieving a final weighted f1 score of 0.876 (Liu and Long). Another concatenated network addressed the issue of having a lack of training data by inputting the cell images into three networks AlexNet, CNN-RNN, and a DCT-LSTM. The output of the three models was averaged to get the final prediction, and their best accuracy was 0.866 (Shah et al.).

Finally, recent solutions have utilized hybrid models, with a neural network for feature extraction, a feature selector, and a final classifier model. One group of researchers wanted to optimize training time and avoid overfitting when training networks for ALL classification, and created a hybrid model called ResRandSVM trained on using ResNet feature extractor and a Random Forest feature selector to a support vector machine (SVM) classifier, achieving an accuracy of 0.900 (Sulaiman et al.). Another study believed that concatenated neural networks were over complicated, and utilized hybrid learning to simplify. They used a model that used an InceptionNetV3 (Szegedy et al.) feature extractor to an XGBoost Classifier to diagnose ALL and achieved a weighted f1 score of 0.986 (Ramaneswaran et al.). This paper did not report their accuracy.

Even with the current state of the art's high-performing models, one flaw in today's ALL classification models is their relative architectural complexity. The best-achieving hybrid models utilize a middle feature selector or extra modules which has to be trained and also further lessens model interpretability. For example, ResRandSVM utilized a middle random forest selector, while the VGG16 model utilized an ECA module. By removing the random forest selector or any extra modules, one could vastly decrease the number of parameters the model has to learn during training, and this would reduce training time as well. Furthermore, we can quantify complexity as a measure of the number of learned parameters during training. Models such as InceptionNetV3 also have a high amount of parameters. Resnet50 in ResRandSVM has 25,557,032 parameters, while InceptionNetV3 has 27,161,264 parameters. One could also utilize EfficientNetB0 (Tan and Le) to reduce the number of parameters, as it has just 5,288,548 parameters. Furthermore, EfficientNetB0 performs similarly to InceptionNetV3 and better than ResNet50 on the ImageNet database (Keras). ResNet50 achieved a top-1 accuracy of 74.9%, InceptionNetV3 achieved a top-1 accuracy of 77.9%, and EfficientNetB0 achieved a top-1 accuracy of 77.1%. Finally, EfficientNetB0 is much smaller in size than current state of the art models, only taking 28 MB, in comparison to InceptionV3 (92 MB), ResNet50 (98 MB), and VGG16 (528 MB) (Keras). Even with all of these improvements, EfficientNetB0 has still not been tested in a hybrid configuration with XGBoost in ALL classification.

Our solution simplifies the issue of architectural complexity by utilizing a hybrid model but removing the random forest regressor from past works. We first perform image preprocessing through color normalization and 100x100 cropping. We utilize EfficientNetB0 with transfer learning, a CNN architecture that was not tested by the current state of the art in a hybrid format with XGBoost. EfficientNetB0 is pre-trained on ImageNet weights, to lower the amount of data required for accurate classification through transfer learning. The data is from the International Symposium on Biomedical Imaging (ISBI) ALL challenge dataset. The data was pre-split into training and test sets subject-wise, and we split the given test set with a 0.5 split for our test and validation sets. Our architecture can be visualized in FIG 2.



HEM cell ALL cell

Figure 1. The close visual similarity between a HEM and an ALL cell.

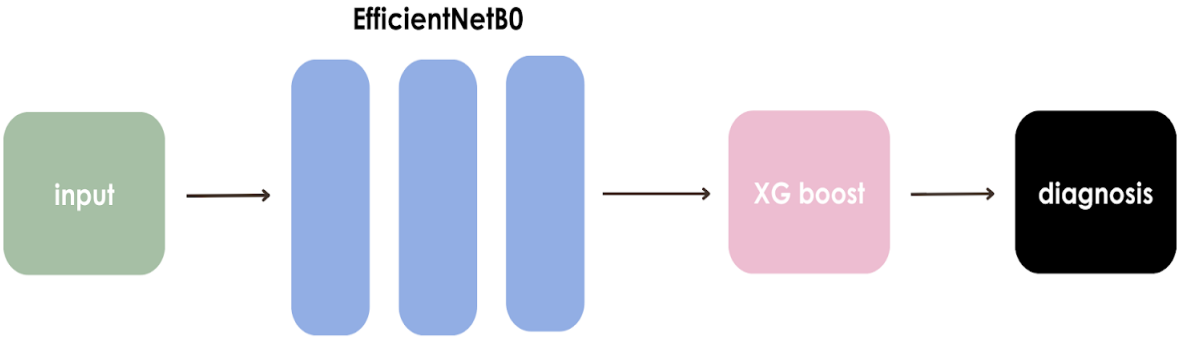


Figure 2. A high-level flowchart of our proposed hybrid neural network architecture.

The 85% accuracy demonstrates our success in creating a relatively simple model for leukemia classification while still maintaining similar performance to more complex state-of-the-art hybrid models. These model results are presented in TABLE 2 and TABLE 3. In TABLE 3, we present an accuracy comparison between the proposed model and the state of the art, which proves that our model performed better than many of the current state-of-the-art models. In this paper,

- We utilize EfficientNetB0 as a feature extractor, a previously untested architecture in a hybrid architecture with XGBoost, for the classification of ALL.
- We build a new hybrid architecture without a middle feature selector or additional modules, increasing the architectural simplicity due to a lower amount of parameters.

Methods

Model Architecture

Our model architecture consists of EfficientNetB0 as a feature extractor, with its output features sent to an XGBoost classifier to return a final classification for ALL. EfficientNets (Tan and Le) were developed to achieve high accuracy with low parameter counts. This model has an standard input size of 224x224x3 and an output size of 1280 features. They utilize a scalable architecture made of standardized building blocks, with a coefficient to change their size. Their architecture consists of repeated building blocks, called MBConv blocks.

MBConv stands for Mobile Inverted BottleNecked Convolution. This block is made up of smaller sections detailed in FIG 3.

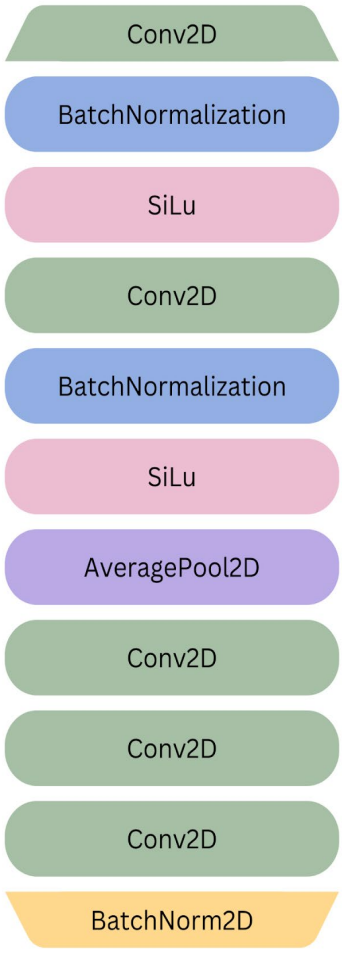


Figure 3. A model of our hybrid neural network architecture.

Because it is a residual block, it utilizes an identity skip connection, allowing deeper EfficientNets to behave like shallow networks. In a regular residual block, the number of channels goes from larger, to smaller, and back to larger. When the MBConv block has an inverted residual connection (IRC), the number of channels in the bypassed conv layers instead goes from small, to large, and back to small (16->64->16 in EfficientNet blocks). The purposes of these convolutional blocks are to extract features from the image using kernels, in a

process called convolution. The weights of the kernel are learned by the model and are linearly applied to the image to extract features such as edges or other patterns. These blocks use the SiLU (Sigmoid Linear Units) activation function described in Equation 2. SiLU acts similar to ReLU. Its equation is x multiplied by the sigmoid function of x, shown in Equation 3. This stops the model from monotonically increasing with the use of the sigmoid at smaller values. As values become large, however, the value of SiLU approaches ReLU. SiLU activation is used by MBConv blocks. A variation of this function, called Swish activation, is used in EfficienNet's first layer. This adds a trainable coefficient to the x variable in the sigmoid function. This allows for better fine-tuning for gradient smoothing, similar to batch normalization. The equation for Swish is described in Equation 1.

$$Swish = x\Sigma(\beta x)$$

Equation 1. Swish function.

$$\Sigma(x) = \frac{1}{1+e^{-x}}$$

Equation 2. Sigmoid function.

$$SiLU = x\Sigma(x)$$

Equation 3. SiLU function.

We utilize a pre-trained EfficientNetB0 on ImageNet weights. By having pre-initialized weights, the model utilizes transfer learning, providing better accuracy. Early convolutional layers, when preloaded with weights from ImageNet, extract similar features needed in all classification problems. Through training, we fine-tune the weights to our dataset, resulting in a better prediction with less data used.

Our secondary model is an XGBoost classifier to differentiate the cells between ALL or HEM with input from the trained EfficientNetB0 to increase the accuracy of our overall model, rather than using the EfficientNet as a standalone extractor and classifier. The XGBoost (Chen and Guestrin) classifier provides a gradient-boosting framework, utilizing a decision tree-based algorithm. Gradient boosting involves new models that are created for predicting previous models 'errors and then they are combined for the final prediction, minimizing errors. XGBoost can be used for classification or regression, and in this case, we utilize XGBoost as a final classification head, giving an final classification output with inputted features from EfficientNetB0. Once we implemented the XGBoost model on top of our EfficientNetB0 model, our accuracy increased from 81% to 85%.

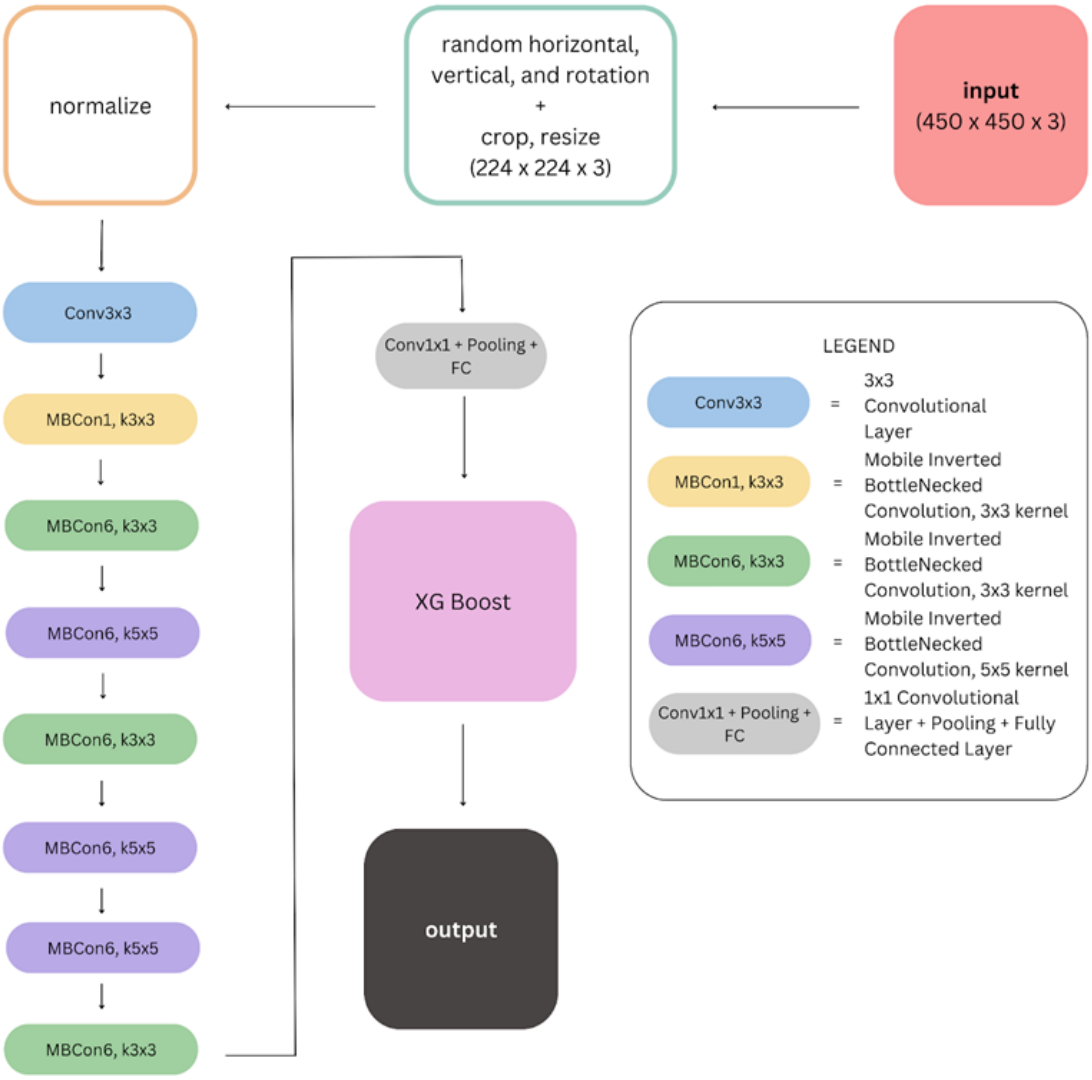


Figure 4. A detailed flowchart of our neural network architecture.

Dataset and Data Preprocessing

Our dataset is the ISBI 2019 ALL Challenge Dataset (C-NMC 2019). The training dataset is 10,661 samples, classified as ALL or HEM cells. These images were taken from 47 cancerous and 26 healthy patients. There is a class imbalance however, with 7272 ALL cells, and just 3389 HEM cells in the training data. The training and validation dataset 1867 images, with a CSV file that is read to add labels to each image. The data is then split into test and validation with a split of 0.5.

For preprocessing, our data is first center cropped to be 448x448, and then resized to 224x224. This is to remove some of the black boundary on the edges, which serves no role to classification. After the resize, the images are randomly flipped horizontally and vertically with a probability of 0.2, and rotated randomly with a

margin of 10 degrees. At the end, the images are converted to tensors, and individual pixel values are normalized from 0-255 to be between 0-1.

Model Training

Our architecture is trained in a two-step process. First, the EfficientNetB0 feature extractor model is trained for 5 epochs. We utilize a CrossEntropy loss function during training of EfficientNetB0. Our optimizer is Adamax, with a learning rate of 0.001. While training EfficientNetB0, our images are batched sets of 40. The training accuracy for EfficientNetB0 can be visualized in FIG 6. We then process raw training, testing, and validation image data through the trained CNN to generate train, test, and validation feature sets used for the XGBoost classifier. When training the XGBoost classifier, we change the "eta" hyperparameter to 0.35, which is an alias for the learning rate, and also set 'tree_method 'to 'gpu_hist'. The training error for the XGBoost model can be visualized in FIG 8.

Results

We experimented with a multitude of different model architectures by switching out which feature extractor was used. The feature extractors that we used included VGG16, Resnet18, InceptionV3, and EfficientNetB0. The evaluation accuracy of each feature extractor is listed in TABLE 1. When integrated into our model, EfficientNetB0 achieved the highest performance with an accuracy of 0.85. Other evaluation metrics of EfficientNetB0 are detailed in TABLE 2.

Table 1. Accuracy scores of our model with each feature extractor.

Feature Extractor	Accuracy
VGG16 (Simonyan and Zisserman)	0.71
Resnet18 (He et al.)	0.78
InceptionV3 (Szegedy et al.)	0.81
EfficientNetB0 (Tan and Le)	**0.85**

Table 2. Precision, recall, and f1-scores of the EfficientNetB0 hybrid model.

	Precision	Recall	F1-score
ALL	0.86	0.91	0.89
HEM	0.82	0.72	0.77

Attention heatmaps represent which elements of an image are given importance by the feature extractor in classifying it. These plots are generated from the 7th feature layer of EfficientNetB0. They allow us to better

interpret the reasons behind the predictions made by our feature extractor. FIG 5 contains the GRAD-CAM (Selvaraju) heatmaps of two ALL cells and two HEM cells with the red areas of the cell being important and the blue areas being less important to the model. Our final model prediction confusion matrix can be seen in FIG 7. With this matrix, we see the details of where our model makes false positive, true negative, false negative, or true positive predictions, and these results will be further analyzed in the conclusion section.

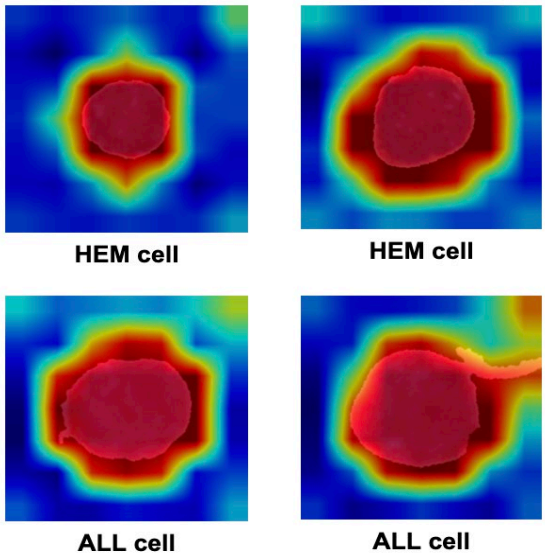


Figure 5. Attention heatmaps for the feature extractor.

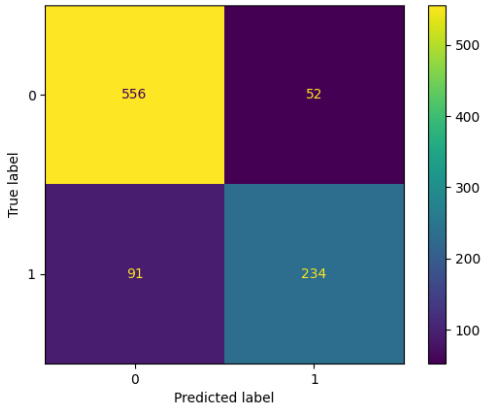


Figure 7. Confusion matrix. Class 0 is ALL, and Class 1 is HEM.

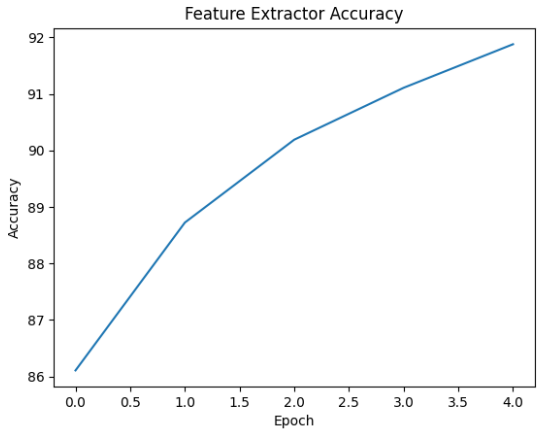


Figure 6. Accuracy of the EfficientNetB0 feature extractor.

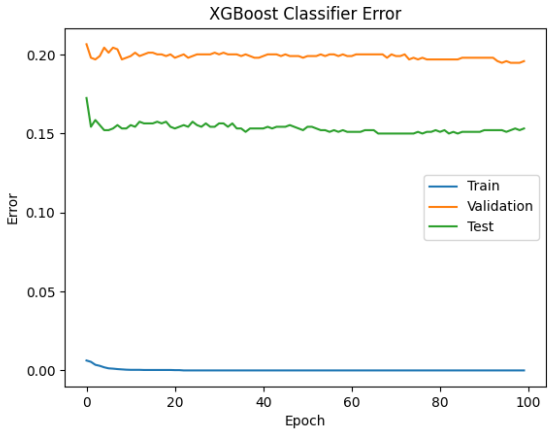


Figure 8. XGBoost classifier error.

Table 3. A comparison of the proposed model to other models from previous works that have used the same dataset.

Model	Accuracy
ResNet50 (Rezayi et al.)	0.816
CNN (Rezayi et al.)	0.821
VGG-16 (Rezayi et al.)	0.846
DCT-LSTM and CNN (Shah et al.)	0.866
ResRandSVM (Sulaiman et al.)	0.900
ECA-Net Based on VGG16 (Ullah et al.)	0.911
Proposed Model	**0.850**

Conclusion

We achieved an accuracy of 85% with this hybrid model, as described in TABLE 1, which is on-par with the performance of existing models. In comparison to models that performed better, shown in TABLE 3, such as VGG16 with an ECA module, DCT-LSTM and an CNN, and ResRandSVM, we argue that EfficientNetB0 model utilizes a much simpler architecture with far fewer parameters. We can consider parameter size to be an indication of the network size. For example, VGG16 from VGG16 + ECA (Ullah et al.) alone has over 128,000,000 parameters, AlexNet from DCT-LSTM (Shah et al.) itself has over 62,000,000, and Resnet50 from ResRandSVM (Sulaiman et al.) has over 25,000,000, while our model's feature extractor has just 5,288,548 (Tan and Le). Thus, our model would be useful in situations where large models cannot be used, such as in hospitals with lower powered devices or in developing countries. With these results, we have achieved utilizing EfficientNetB0 in an architecturally simple hybrid format for ALL detection. When looking at precision, recall, and F1-Score in TABLE 2, we also note how each score is higher for ALL than HEM. We attribute this to the dataset imbalance, as there were 7272 ALL samples, yet only 3389 HEM samples in the data. This is also reflected in the confusion matrix in FIG 7, as there were a greater number of predictions for ALL when the ground truth was HEM than HEM when the truth was ALL. This indicates that our model was weighted to predict ALL somewhat more often, which makes sense with the dataset imbalance towards ALL. Feature extraction was visualized through GRAD-CAM heatmaps, shown in FIG 5. Our model is clearly learning by focusing on the cell itself, and we notice that the model does not use the outer regions of the cell image.

The limitations of this study were partly due to the dataset imbalance. While we did try class weighing using weighted cross entropy loss, our model still delivered a low accuracy, and thus we removed it due to it adding unneeded complexity to our method pipeline. We believe that with more samples from the HEM dataset, possibly through further data augmentation techniques, our model would perform better. Another limitation was the time we had to train our final model, as we found out about an issue with our original model having subject overlap very late in our research. With more time for fine-tuning, we believe our model can be improved to achieve better results.

In the future, we would like to improve the model accuracy through further data augmentation and alternate ALL datasets. This would make our model more generalized to accurately classify a wide range of ALL cases, allowing it to better serve doctors to provide a diagnosis for ALL. This would give patients the opportunity to move forward to getting the treatment they need based on an efficient and accurate diagnosis without needing access to a powerful computer. In the end, this would save patient lives and prevent unnecessary treatment from taking place for ALL.

Through this study, we present a new method to classify lymphoblasts from healthy cells to aid the diagnosis of ALL. Our model utilizes EfficientNetB0 as a feature extractor, a low parameter model, trained on the raw images and XGBoost as our ultimate classifier using the output from EfficientNetB0, which is a novel architecture for the state of the art. We hope to see applications for our model in the healthcare industry for early diagnosis of ALL, as it is suited towards low powered machines such as the ones found in hospitals. With efficient tools that can diagnose ALL early on, we hope to prevent cases in which the disease cannot be diagnosed until it has spread and improve the lives of children who have ALL.

Acknowledgments

We would like to thank S. Shailja, our research mentor, and our TA's Arthur Caetano and Satish Kumar for their guidance throughout the review, programming, and writing process. Furthermore, we would like to thank the SRA program as a whole, especially Lina Kim and Teresa Holden, for the opportunity to perform university-level research at UCSB.

Author Contribution Statement

K.L. created the model. K.L., A.R., and M.S. conducted the experiments and further developed the algorithm. A.R. and M.S. conducted a thorough literature review to discuss and compare our results to the state-of-the-art. M.S. created the figures and tables. All authors analyzed and discussed the results of the model and reviewed the manuscript.

References

"Acute lymphocytic leukemia - Symptoms and causes." *Mayo Clinic*, 21 September 2022, https://www.mayoclinic.org/diseases-conditions/acute-lymphocytic-leukemia/symptoms-causes/syc-20369077. Accessed 15 July 2023.

Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data miningProceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, 785--794. Arxiv, https://arxiv.org/pdf/1603.02754.pdf.

He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition, vol. 2016, no. 2016, 2016, 770--778. IEEE Xplore, Deep residual learning for image recognition.

"Keras Applications." *Keras*, https://keras.io/api/applications/. Accessed 24 July 2023.

"Leukemia Misdiagnosis - The Personal Injury Center." *Medical Malpractice Center*, https://malpracticecenter.com/misdiagnosis/leukemia/. Accessed 15 July 2023.

Liu, Ying, and Feixiao Long. "Acute lymphoblastic leukemia cells image analysis with deep bagging ensemble learning." 2019. *ResearchGate*,

https://www.researchgate.net/publication/331942838_Acute_lymphoblastic_leukemia_cells_image_analysis_with_deep_bagging_ensemble_learning.
Puckett, Yana, and Onyee Chan. "Acute Lymphocytic Leukemia." *Treasure Island (FL)*, 2022, https://www.ncbi.nlm.nih.gov/books/NBK459149/. Accessed 15 July 2023.
Ramaneswaran, S., et al. "Hybrid Inception v3 XGBoost Model for Acute Lymphoblastic Leukemia Classification." 2021. *Hindawi*, https://www.hindawi.com/journals/cmmm/2021/2577375/.
Rezayi, Sorayya, et al. "Timely diagnosis of acute lymphoblastic leukemia using artificial intelligence-oriented deep learning methods." *Computational Intelligence and Neuroscience*, no. 2021, 2021. *Hindawi*, https://www.hindawi.com/journals/cin/2021/5478157/.
Schmidhuber, Jurgen. "Deep learning in neural networks: An overview." *Neural networks*, vol. 61, 2015, 85--117. *Elsevier*, https://people.idsia.ch/~juergen/DeepLearning2July2014.pdf.
Selvaraju, Ramprasaath R. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization." *arXiv*, 7 October 2016, https://arxiv.org/abs/1610.02391. Accessed 26 July 2023.
Shah, Salman, et al. "Classification of normal and leukemic blast cells in B-ALL cancer using a combination of convolutional and recurrent neural networks." *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging: Select Proceedings*, 2019, 23--31. *Springer*, https://link.springer.com/chapter/10.1007/978-981-15-0798-4_3.
Simonyan, Karen, and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *arXiv*, 4 September 2014, https://arxiv.org/abs/1409.1556. Accessed 26 July 2023.
Sulaiman, Adel, et al. "ResRandSVM: Hybrid Approach for Acute Lymphocytic Leukemia Classification in Blood Smear Images." *Diagnostics*, vol. 13, no. 12, 2023, p. 2121. *MDPI*, https://www.mdpi.com/2075-4418/13/12/2121.
C. Szegedy *et al*., "Going deeper with convolutions," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 1-9, doi: 10.1109/CVPR.2015.7298594. Accessed 26 July 2023.
Tan, Mingxing, and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks." *International conference on machine learning*, 2019, 6105--6114. *PMLR*, http://proceedings.mlr.press/v97/tan19a.html?ref=jina-ai-gmbh.ghost.io.
Ullah, Muhammad Zakir, et al. "An Attention-Based Convolutional Neural Network for Acute Lymphoblastic Leukemia Classification." *Applied Sciences*, vol. 11, no. 22, 2021, p. 10662. *MDPI*, https://www.mdpi.com/2076-3417/11/22/10662.