Development and Evaluation of Advanced Deep Learning Models for Lymphoma Classification

Aynur Cemre Aka[†], Shayan Sharifi[‡]

***Abstract*—This article presents a comprehensive study on the application of Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Residual Networks (ResNet) for the classification of lymphoma images. The study leverages advanced techniques such as data augmentation, batch normalization, dropout, and attention mechanisms to enhance model performance. In addition, it evaluates model performance using precision, recall, and F1-score metrics. Ensemble learning and decision fusion methods are employed to further improve classification accuracy. The proposed approach addresses the challenge of accurately classifying lymphoma subtypes, specifically Chronic Lymphocytic Leukemia (CLL), Follicular Lymphoma (FL), and Mantle Cell Lymphoma (MCL).**

The dataset is prepared by preprocessing images and dividing them into smaller patches to increase the dataset size and improve model training. Both RGB and grayscale formats are utilized to assess the impact of color space on model performance. The CNN model with attention mechanisms demonstrates high accuracy in identifying relevant features in the images, while the RNN model effectively captures temporal dependencies within the image patches. The ResNet model leverages residual learning, making it particularly effective for deeper architectures. Decision fusion combines predictions from multiple models and color spaces, improving overall accuracy and robustness. Ensemble learning, combining logistic regression and k-nearest neighbors (k-NN), outperforms individual models by leveraging complementary strengths.

***Index Terms*—Lymphoma Classification, Deep Learning, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Attention Mechanism, Decision Fusion Mechanism**

I. Introduction

Remark I.1. In recent years, the application of deep learning techniques to medical image analysis has garnered significant attention, owing to its potential to automate and enhance the accuracy of diagnostic processes. The classification of lymphoma subtypes, in particular, is a critical task in oncology, as it directly impacts treatment decisions and patient outcomes. This paper aims to advance the state of the art in lymphoma image classification by leveraging advanced neural network architectures [1] [2] [3].

Remark I.2. Problem Perspective: The classification of lymphoma images into subtypes such as Chronic Lymphocytic Leukemia (CLL), Follicular Lymphoma (FL), and Mantle Cell Lymphoma (MCL) remains a challenging problem due to the complex and heterogeneous nature of these diseases. Previous studies, such as Orlov et al. (2010) [4] and Janowczyk & Madabhushi (2016) [1], have demonstrated the potential of machine learning techniques in this domain but often focused on single model architectures, limiting their scope and robustness. Moreover, the lack of comprehensive comparisons between different neural network models and the use of attention mechanisms have left gaps in fully addressing this problem.

Remark I.3. Paper Contribution: This paper presents a novel and comprehensive approach to lymphoma classification by integrating and comparing three advanced neural network architectures: Convolutional Neural Networks (CNNs) with attention mechanisms, Recurrent Neural Networks (RNNs) with Gated Recurrent Units (GRUs) and attention mechanisms, and Residual Networks (ResNet). The relevance of this work lies in its potential to significantly improve the accuracy and robustness of lymphoma subtype classification, thereby supporting better clinical decision-making.

Our approach utilizes data augmentation to enhance model generalization and includes a decision fusion technique that combines predictions from multiple models and color spaces. This method addresses the limitations of previous studies by leveraging the strengths of different models [5] [6] [7]. Furthermore, we implement an ensemble learning strategy, demonstrating superior performance by integrating complementary model strengths.

The novelty of our proposal lies in the comprehensive integration of multiple architectures, the use of attention mechanisms to improve feature extraction, and the application of decision fusion and ensemble learning techniques [8] [9] [10]. These innovations provide a robust framework for improving lymphoma classification accuracy. Our results suggest that this integrated approach can be effectively reused and adapted for other medical image classification tasks, thereby advancing both scientific research and practical applications in the medical field [11] [12] [13].

Remark I.4. Summary of Contributions:

- **Comprehensive Model Integration**: A robust comparison and integration of CNNs with attention mechanisms, RNNs with GRUs and attention mechanisms, and ResNet architectures.
- **Attention Mechanisms**: Enhanced model performance through the use of attention mechanisms, enabling better focus on critical image features.

[†] Department of Information Engineering, University of Padova, email: {aynurcemre.aka}@studenti.unipd.it

[‡] Author two affiliation, email: {shayan.sharifi}@studenti.unipd.it

- **Decision Fusion and Ensemble Learning**: Improved classification accuracy and robustness by combining predictions from multiple models and color spaces and implementing an ensemble learning strategy.
- **Evaluation on Multiple Color Spaces**: Insights into the impact of different color spaces (RGB, grayscale, and LAB) on model performance.
- **Extensive Performance Metrics**: Detailed evaluation using precision, recall, and F1-score metrics, along with computational efficiency assessment through time and memory usage tracking [14] [15].

II. Related Work

- **Overview:** Orlov et al. (2010) [4] presented an early approach to automated lymphoma classification using transform-based global features. The authors utilized various image transformations to extract features from lymphoma images, followed by machine learning algorithms for classification. Their method demonstrated the potential of automated systems in medical image analysis. However, the reliance on handcrafted features limited the model's ability to generalize across diverse datasets and complex image variations. Our work extends this by employing deep learning architectures that automatically learn hierarchical features directly from raw images, enhancing robustness and accuracy.
- **Deep Learning for Digital Pathology:** Janowczyk and Madabhushi (2016) [1] provided a comprehensive tutorial on applying deep learning to digital pathology. They showcased the efficacy of Convolutional Neural Networks (CNNs) in extracting relevant features from pathology images, significantly improving classification performance over traditional methods. While their study laid the groundwork for using CNNs in medical imaging, it primarily focused on single CNN architectures without exploring the benefits of combining different models. Our research builds on their foundation by integrating CNNs with attention mechanisms and comparing them with other advanced models like RNNs and ResNets to provide a more robust solution.
- **Automated Classification Using Deep Neural Networks:** Tambe et al. (2019) [16] explored the use of deep neural networks for the automated classification of lymphoma subtypes. Their approach utilized a combination of CNNs and data augmentation techniques to enhance model performance. While effective, their study did not incorporate attention mechanisms or ensemble learning, which can significantly improve model interpretability and accuracy. Our study addresses these gaps by integrating attention mechanisms into CNN and RNN architectures and employing ensemble learning to leverage the strengths of multiple models.
- **Residual Networks for Image Recognition:** He et al. (2016) [17] introduced the concept of Residual Networks (ResNets), which have since become a cornerstone in deep learning for image recognition. ResNets address the vanishing gradient problem in deep networks by introducing shortcut connections that allow gradients to flow more easily during training. While ResNets have been widely adopted in various image classification tasks, their application to lymphoma classification has been limited. This study incorporates ResNet architectures to leverage their depth and robustness, comparing their performance with other models in the context of medical image classification.
- **Neural Machine Translation and Attention Mechanisms:** Bahdanau et al. (2015) [18] proposed an attention mechanism for neural machine translation, which has since been adapted to various tasks, including image analysis. Attention mechanisms enable models to focus on the most relevant parts of the input, improving performance in tasks with complex and high-dimensional data. By integrating attention mechanisms into CNN and RNN architectures, our study enhances feature extraction and classification accuracy, addressing limitations in previous approaches that did not utilize this technique.
- **Summary of Contributions:** In summary, the key contributions of this paper, which extend and improve upon previous work, are as follows:
  1) **Comprehensive Model Integration**: A robust comparison and integration of CNNs with attention mechanisms, RNNs with GRUs and attention mechanisms, and ResNet architectures, providing a holistic view of their performance in lymphoma classification.
  2) **Enhanced Feature Extraction**: The use of attention mechanisms in CNN and RNN architectures to improve feature extraction and focus on critical image regions, addressing the limitations of models that rely solely on convolutional layers.
  3) **Decision Fusion and Ensemble Learning**: Improved classification accuracy and robustness through the combination of predictions from multiple models and color spaces, as well as an ensemble learning strategy that leverages complementary model strengths.
  4) **Evaluation on Multiple Color Spaces**: Insights into the impact of different color spaces (RGB, grayscale, and Lab) on model performance, which have been underexplored in previous studies.
  5) **Extensive Performance Metrics**: Detailed evaluation using precision, recall, and F1-score metrics, along with computational efficiency assessment, providing a comprehensive assessment of model performance.

III. Processing Pipeline

Our lymphoma classification approach integrates several advanced deep learning techniques within a comprehensive processing pipeline. This section provides a high-level overview of our methodology, explaining the roles of different processing blocks and their interactions.

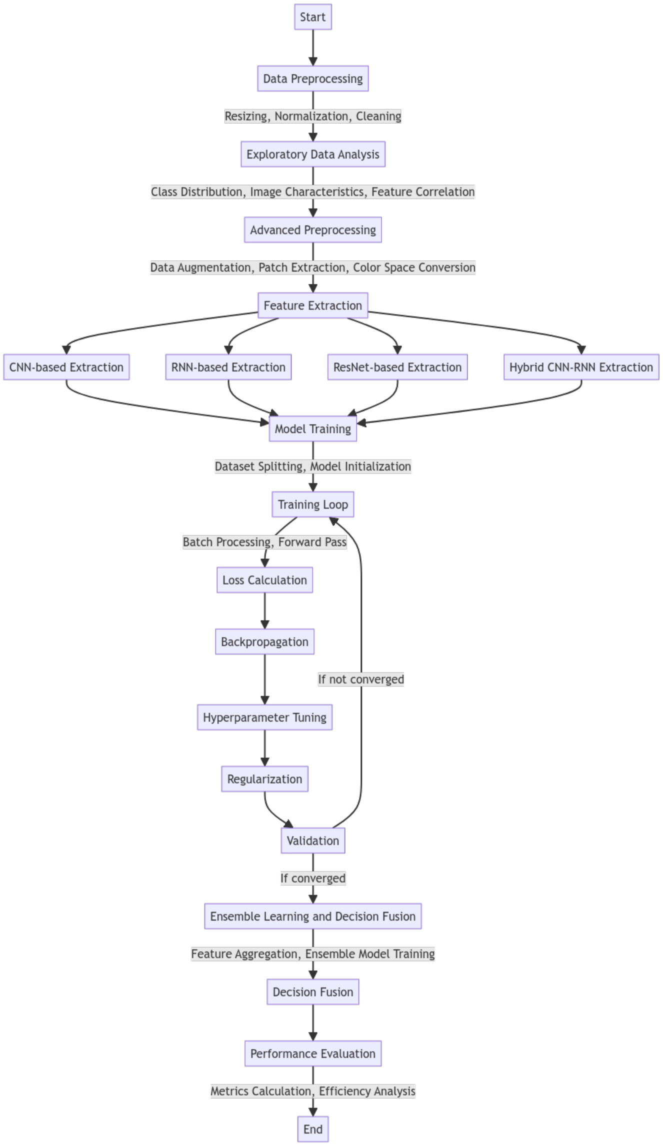
Start
Data Preprocessing
Resizing, Normalization, Cleaning
Exploratory Data Analysis
Class Distribution, Image Characteristics, Feature Correlation
Advanced Preprocessing
Data Augmentation, Patch Extraction, Color Space Conversion
Feature Extraction
CNN-based Extraction
RNN-based Extraction
ResNet-based Extraction
Hybrid CNN-RNN Extraction
Model Training
Dataset Splitting, Model Initialization
Training Loop
Batch Processing, Forward Pass
Loss Calculation
Backpropagation
If not converged
Hyperparameter Tuning
Regularization
Validation
If converged
Ensemble Learning and Decision Fusion
Feature Aggregation, Ensemble Model Training
Decision Fusion
Performance Evaluation
Metrics Calculation, Efficiency Analysis
End

Fig. 1: Processing Pipeline

A. Data Preprocessing

The pipeline begins with data preprocessing to prepare the lymphoma images for model input:

1) **Image Acquisition**: Collection of lymphoma images from the dataset.
2) **Resizing and Normalization**: Images are resized to 1040x1388 pixels and pixel values are normalized to [0,1].
3) **Initial Data Cleaning**: Removal of any corrupted or irrelevant images.

B. Exploratory Data Analysis (EDA)

Following initial preprocessing, we conduct an exploratory data analysis to gain insights into our dataset:

1) **Class Distribution Analysis**: Examination of the balance between different lymphoma subtypes in the dataset.
2) **Image Characteristics**: Analysis of image properties such as color distribution, brightness, and contrast across classes.
3) **Feature Correlation**: Investigation of relationships between different color channels and image features.
4) **Outlier Detection**: Identification of any anomalous images that may require special handling.
5) **Visualization**: Creation of histograms, scatter plots, and other visualizations to represent data distributions and relationships.

C. Advanced Data Preprocessing

Based on insights from EDA, we perform additional preprocessing steps:

1) **Data Augmentation**: Application of techniques such as rotation, shifting, and flipping to increase dataset diversity and address class imbalance if detected.
2) **Patch Extraction**: Division of images into 36x36 pixel patches to increase sample size and capture fine-grained details.
3) **Color Space Conversion**: Transformation of images into multiple color spaces (RGB, Grayscale, LAB, HSV) based on color analysis from EDA.

D. Feature Extraction

Following preprocessing, the pipeline employs various deep learning models for feature extraction:

1) **CNN-based Extraction**: Capture of spatial features using convolutional layers.
2) **RNN-based Extraction**: Modeling of temporal dependencies among image patches using GRUs.
3) **ResNet-based Extraction**: Learning of complex patterns through residual connections.
4) **Hybrid CNN-RNN Extraction**: Combination of spatial and temporal feature extraction.

E. Model Training

The training phase is a critical component of our pipeline:

1) **Dataset Splitting**: Division of the preprocessed dataset into training, validation, and test sets.
2) **Model Initialization**: Setup of various model architectures with randomly initialized weights.
3) **Batch Processing**: Training data is processed in batches to optimize memory usage and computational efficiency.
4) **Forward Pass**: Input data is passed through the models to generate predictions.
5) **Loss Calculation**: Computation of categorical cross-entropy loss between predictions and true labels.
6) **Backpropagation**: Calculation of gradients and update of model parameters using optimization algorithms.
7) **Hyperparameter Tuning**: Adjustment of learning rates, batch sizes, and other hyperparameters using grid search and cross-validation.

8) **Regularization**: Application of techniques such as dropout and early stopping to prevent overfitting.
9) **Validation**: Regular evaluation of model performance on the validation set to monitor training progress.

F. Ensemble Learning and Decision Fusion

To enhance classification accuracy, our pipeline incorporates ensemble learning and decision fusion:

1) **Feature Aggregation**: Extraction and combination of features from different models.
2) **Ensemble Model Training**: Training of logistic regression and k-NN classifiers on the aggregated features.
3) **Decision Fusion**: Combination of predictions from multiple models and color spaces using weighted averaging.

G. Performance Evaluation

The final stage of our pipeline involves comprehensive performance evaluation:

1) **Metrics Calculation**: Computation of accuracy, precision, recall, and F1-score on the test set.
2) **Computational Efficiency Analysis**: Measurement of training time, inference time, and memory usage.
3) **Energy Efficiency Assessment**: Evaluation of energy consumption during training and inference.

This processing pipeline integrates various advanced techniques to achieve robust and accurate lymphoma classification. The modular design allows for easy modification and extension of individual components, facilitating future improvements and adaptations to similar medical image classification tasks.

IV. Signals and Features

A. Measurement Setup and Data Format

Our study utilizes a publicly available lymphoma image dataset, containing samples of three lymphoma subtypes: Chronic Lymphocytic Leukemia (CLL), Follicular Lymphoma (FL), and Mantle Cell Lymphoma (MCL). Each image in the dataset represents a high-resolution microscopic view of lymphoma tissue.

- Image Format: RGB color images
- Resolution: Original images resized to 1040x1388 pixels
- Pixel Values: Normalized to the range [0, 1]

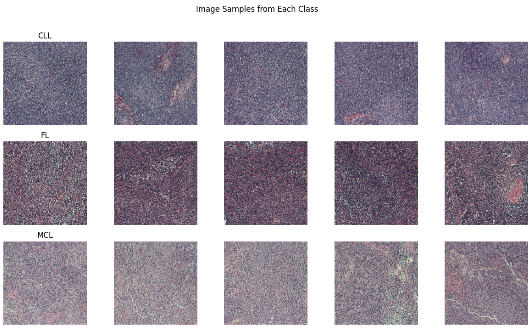

Fig. 2: Sample images from the dataset

1) Class Distribution:

- **Description**: The distribution of images across the three classes (CLL, FL, MCL) was analyzed to ensure a balanced dataset.
- **Visualization**: The class distribution is depicted in Fig. 2, showing the number of images in each class.

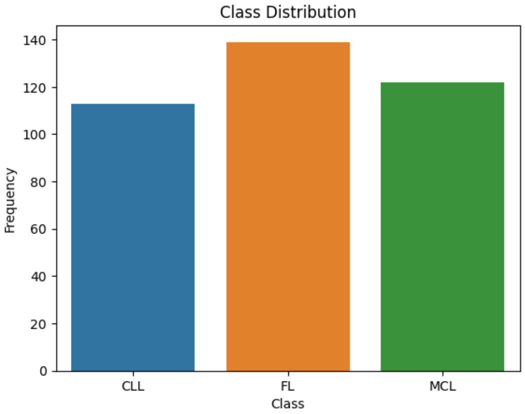

Fig. 3: class distribution

2) Distribution of Channel Colors:

- **Description**: The color distribution across the RGB channels was examined to identify any dominant colors or patterns that could influence the model.
- **Visualization**: The distribution of channel colors is illustrated in Fig. 3, highlighting the intensity of each color channel across the dataset.

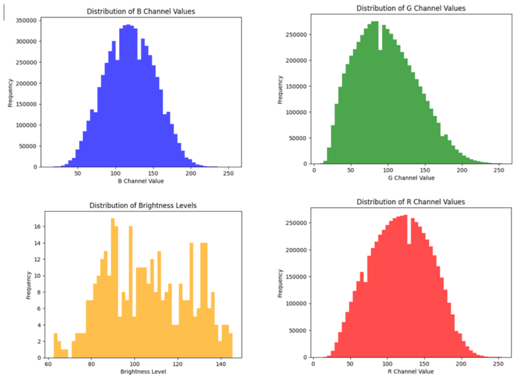

Fig. 4: color and brightness distribution

3) Distribution of Image Variances Across Classes:

- **Description**: The variance of pixel intensities within images was analyzed for each class to understand the variability and texture differences.
- **Visualization**: The distribution of image variances is shown in Fig. 4, providing insights into the texture characteristics of each lymphoma subtype.

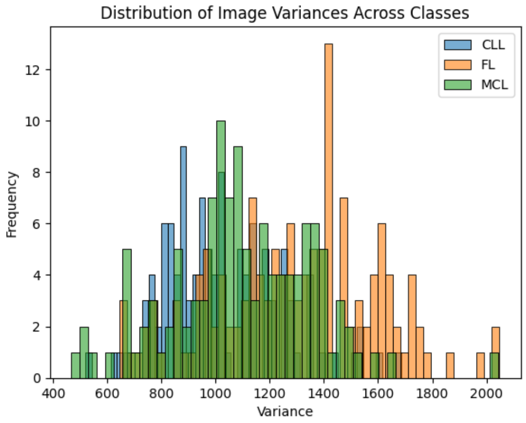

Fig. 5: Distribution of image varience

4) *Correlation Between Color Channels:*

- **Description**: The correlation between the RGB color channels was assessed to identify any linear relationships that might affect feature extraction.
- **Visualization**: The correlation matrix is presented in Fig. 5, depicting the relationships between the red, green, and blue channels.

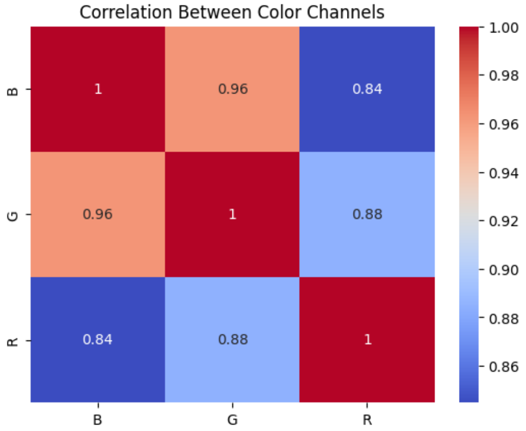

Fig. 6: corelation between color chanels

B. *Pre-processing Techniques*

1) Normalization: Each image is normalized to standardize pixel intensities across the dataset, improving model convergence and stability during training.

2) Data Augmentation: To enhance model generalization and address limited data issues, we apply the following augmentation techniques:

- Random rotations
- Width and height shifts
- Shearing
- Zooming
- Horizontal and vertical flips

3) Color Space Conversion: Images are converted into multiple color spaces to provide diverse feature representations:

- RGB (original)
- Grayscale
- LAB color space

4) Patch Extraction: To increase the number of training samples and capture fine-grained details:

- Patch Size: 36x36 pixels
- Extraction Method: Non-overlapping patches
- Rationale: This approach increases the dataset size and allows models to focus on local features

C. *Feature Extraction*

1) *Convolutional Neural Network (CNN) Features:*

- Spatial features are extracted using convolutional layers with ReLU activations
- Batch normalization is applied after each convolutional layer
- Max-pooling layers are used for downsampling
- Attention mechanisms are incorporated to focus on relevant image regions

2) *Recurrent Neural Network (RNN) Features:*

- Gated Recurrent Units (GRUs) are employed to capture temporal dependencies among image patches
- Attention mechanisms are integrated to highlight significant temporal features

3) *Residual Network (ResNet) Features:*

- Residual blocks, consisting of convolutional layers, batch normalization, and ReLU activations
- Shortcut connections enable the learning of complex patterns while mitigating the vanishing gradient problem

4) *Hybrid CNN-RNN Features:*

- CNN layers extract spatial features from each patch
- Features are flattened and reshaped into sequences
- GRU layers process these sequences to capture temporal dependencies
- Attention mechanisms are applied to focus on important spatial-temporal features

D. *Segmentation and Windowing Strategy*

While our data is not time-series, we adopt a spatial segmentation approach:

- Each 1040x1388 image is divided into four 36x36 non-overlapping patches
- These patches are treated as a sequence for RNN and hybrid CNN-RNN models
- This strategy allows models to capture both local and global image characteristics

E. *Feature Vector Generation*

For each model architecture, feature vectors are extracted from the penultimate layer before the final classification layer:

- CNN: Flattened output of the final convolutional layer

- RNN: Final hidden states of the GRU layers
- ResNet: Output of the global average pooling layer applied to the final residual block
- Hybrid CNN-RNN: Concatenated output of CNN and RNN layers

These feature vectors serve as inputs for our ensemble learning approach, combining logistic regression and k-nearest neighbors classifiers to leverage complementary strengths of different architectures.

V. Learning Framework

Here we describe the learning strategy and algorithm devised to tackle the problem of lymphoma classification. This section delves into the architecture of the learning models, the parameters used, and the optimization techniques employed. The following subsections provide a detailed breakdown of the learning framework.

A. Model Architectures

1) Convolutional Neural Networks (CNNs) with Attention Mechanisms:

- **Function**: CNNs are employed to capture spatial features from the images through convolutional layers. Attention mechanisms are integrated to enhance the model's ability to focus on the most relevant parts of the images.
- **Architecture**:
  - **Convolutional Layers**: Extract low-level features such as edges and textures. Typically, several layers are stacked, with each layer increasing the abstraction level of the features.
  - **Batch Normalization**: Normalizes the output of each convolutional layer to stabilize and speed up training.
  - **ReLU Activation**: Introduces non-linearity to allow the network to learn more complex functions.
  - **Max-Pooling Layers**: Downsample the feature maps, reducing spatial dimensions and computational load.
  - **Attention Layers**: Calculate attention weights for each feature, enabling the network to focus on the most informative parts of the image.
  - **Fully Connected Layers**: Perform final classification based on the extracted features.
- **Attention Mechanism Formulation**:

$$e_i = \tanh(W x_i + b) \tag{1}$$

$$\alpha_i = \frac{\exp(e_i)}{\sum_j \exp(e_j)} \tag{2}$$

$$c = \sum_i \alpha_i x_i \tag{3}$$

  where W and b are learnable parameters, x_i are the input features, α_i are the attention weights, and c is the context vector.

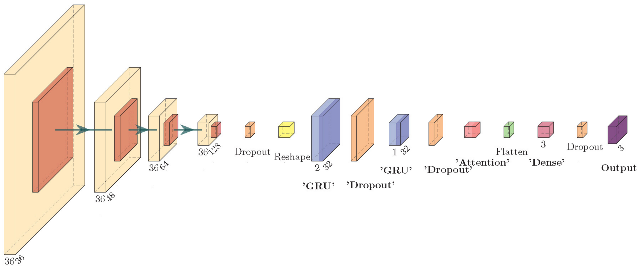

Fig. 7: RNN Architecture

2) Recurrent Neural Networks (RNNs) with Gated Recurrent Units (GRUs) and Attention Mechanisms:

- **Function**: RNNs, specifically GRUs, are used to capture temporal dependencies within image patches. Attention mechanisms are incorporated to improve the model's focus on significant features.
- **Architecture**:
  - **GRU Layers**: Handle sequences of data, capturing temporal dependencies. GRUs are a type of RNN that include update and reset gates to better manage long-term dependencies.
  - **Dropout Layers**: Prevent overfitting by randomly setting a fraction of the input units to zero during training.
  - **Attention Layers**: Similar to CNNs, these layers focus on important temporal features in the sequence of patches.
  - **Fully Connected Layers**: Integrate the temporal features to perform the final classification.
- **GRU Mechanism**:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{4}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{5}$$

$$\tilde{h}_t = \tanh(W_h x_t + r_t \circ (U_h h_{t-1}) + b_h) \tag{6}$$

$$h_t = (1 - z_t) \circ h_{t-1} + z_t \circ \tilde{h}_t \tag{7}$$

  where z_t is the update gate, r_t is the reset gate, $\tilde{h}_t$ is the candidate hidden state, h_t is the hidden state, σ is the sigmoid function, and $\circ$ denotes element-wise multiplication.

3) Residual Networks (ResNets):

- **Function**: ResNets leverage residual learning to enable the construction of deeper networks by addressing the vanishing gradient problem. They are particularly effective in extracting high-level features from images.
- **Architecture**:
  - **Residual Blocks**: Each block includes convolutional layers, batch normalization, and ReLU activation, with shortcut connections that bypass one or more layers to directly connect to the output.
  - **Global Average Pooling**: Reduces each feature map to a single value by taking the average, which is then

fed into fully connected layers for classification.

- **Residual Block Formula**:

$$y = F(x, \{W_i\}) + x \tag{8}$$

where x is the input, y is the output, and $F(x, \{W_i\})$ is the residual function.

4) CNN-RNN with Attention Mechanism: The CNN-RNN model with attention mechanism is designed to leverage both spatial and temporal features from the images. This hybrid model combines the strengths of CNNs in capturing spatial hierarchies and RNNs in modeling temporal dependencies.

- **Architecture**:
  - **Input Layer**: Takes in image patches as input.
  - **Convolutional Layers (CNN)**: Extract spatial features from each patch. These layers are identical to those described in the CNN model, including convolutional, batch normalization, ReLU activation, and max-pooling layers.
  - **Flattening Layer**: Converts the 2D feature maps into 1D feature vectors.
  - **Reshape Layer**: Reshapes the feature vectors into sequences suitable for the RNN input.
  - **GRU Layers (RNN)**: Process the sequence of feature vectors, capturing temporal dependencies.
  - **Attention Layer**: Computes attention weights for each time step, allowing the model to focus on important temporal features.
  - **Fully Connected Layers**: Integrate the spatial and temporal features to perform the final classification.

B. Multi-Color Space Training

To enhance the robustness and generalization of the models, we trained the models using different color spaces:

- **RGB**: The standard Red, Green, and Blue channels.
- **Grayscale**: Single-channel representation capturing intensity.
- **Lab**: Lightness (L) and color-opponent dimensions (a and b).

Each color space provides different information, potentially enhancing the model's ability to learn discriminative features for lymphoma classification.

C. Training and Evaluation

1) Loss Calculation:

- **Loss Function**: Categorical cross-entropy loss is used for multi-class classification:

$$\mathcal{L} = -\sum_{i=1}^{N} y_i \log(\hat{y}_i)$$

where y_i is the true label and $\hat{y}_i$ is the predicted probability for class i.

2) Accuracy Calculation:

- **Accuracy**: The accuracy metric measures the proportion of correctly classified instances:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

D. Hyperparameter Tuning

To optimize model performance, hyperparameters were tuned using grid search and cross-validation. The following hyperparameters were considered:

- **Learning Rate**: Different learning rates were tested to find the optimal value that balances convergence speed and model stability.
- **Batch Size**: Various batch sizes were evaluated to determine the best trade-off between computational efficiency and model performance.
- **Dropout Rate**: Dropout rates were adjusted to prevent overfitting while maintaining model accuracy.
- **Number of Layers and Units**: The depth of the network and the number of units in each layer were fine-tuned to enhance feature extraction and classification performance.

E. Learning Rate Scheduler and Early Stopping

1) Learning Rate Scheduler:

- **Function**: Adjusts the learning rate during training to improve convergence.
- **Implementation**: A learning rate scheduler was employed that reduces the learning rate if the model's performance on the validation set does not improve for a certain number of epochs.

2) Early Stopping:

- **Function**: Stops training when the model's performance on the validation set stops improving, preventing overfitting.
- **Implementation**: Early stopping was used to monitor the validation loss, and training was halted if the loss did not decrease for a specified number of epochs.

F. Decision Fusion and Ensemble Learning

1) Decision Fusion:

- **Function**: Decision fusion combines predictions from multiple models and color spaces (RGB, grayscale and Lab) to improve classification accuracy and robustness.
- **Implementation**: Predictions from CNN, RNN, and ResNet models are averaged to form a final decision:

$$\hat{y}_{\text{fusion}} = \frac{1}{n} \sum_{i=1}^{n} \hat{y}_i$$

where n is the number of models and $\hat{y}_i$ is the prediction from the i-th model.

2) Ensemble Learning:

- **Function**: Ensemble learning integrates logistic regression and k-nearest neighbors (k-NN) models trained on features extracted from the penultimate layers of the individual deep learning models.
- **Implementation**: Features from CNNs, RNNs, and ResNets are concatenated and used as input to the

ensemble models. The final prediction is made based on a weighted voting scheme:

$$\hat{y}_{\text{ensemble}} = \text{argmax}\left(\sum_{i=1}^{n} w_i \hat{y}_i\right)$$

where w_i is the weight assigned to the i-th model's prediction.

G. Time and Space Domain Analysis

To comprehensively evaluate the models, we analyzed their performance in both the time and space domains, as well as their energy efficiency:

1) Time Domain Analysis:

- **Training Time**: The time taken to train each model was recorded and analyzed to assess computational efficiency.
- **Inference Time**: The time required to make predictions on new data was measured, which is crucial for real-time applications.

2) Space Domain Analysis:

- **Memory Usage**: The amount of memory consumed during training and inference was monitored to evaluate the models' efficiency in terms of resource usage.
- **Model Size**: The storage size of each trained model was considered, important for deployment on devices with limited memory.

3) Energy Efficiency:

- **Power Consumption**: The energy consumption during training and inference was tracked to determine the models' environmental impact and operational cost.
- **Efficiency Metrics**: Metrics such as energy per inference and energy per epoch were calculated to provide a detailed understanding of the models' energy efficiency.

H. Interaction and Integration

The interaction between these components is illustrated in the processing flow diagram (Fig. 6). The diagram shows the sequence of steps from data preprocessing to final classification, highlighting how each block contributes to the overall pipeline.

- **Data Flow**: Images are preprocessed and augmented before being fed into the deep learning models. The CNN, RNN, and ResNet models process the images in parallel, each extracting different types of features.
- **Fusion and Ensemble**: The outputs from the models are combined using decision fusion techniques, and the concatenated features are fed into ensemble models for final classification.

VI. Results

Performance Metrics

The performance of various models (CNN, RNN, CNN-RNN, ResNet) was evaluated using different color spaces (RGB, LAB, Grayscale). The key performance metrics considered are accuracy, precision, and recall.

Model	**Accuracy**	**Precision**	**Recall**
CNN Attention	0.96	0.88	0.88
RNN Attention	0.82	0.78	0.76
CNN-RNN Attention	0.85	0.83	0.83
ResNet Attention	0.96	0.90	0.90

TABLE 1: Performance Metrics for RGB

RGB: The table shows that ResNet with Attention and CNN with Attention achieve the highest accuracy and recall in the RGB color space, indicating their superior performance in capturing and classifying features. This suggests that the RGB color space provides rich information for these models, allowing them to better distinguish between classes.

Model	**Accuracy**	**Precision**	**Recall**
CNN Attention	0.84	0.79	0.77
RNN Attention	0.81	0.76	0.74
CNN-RNN Attention	0.86	0.81	0.79
ResNet Attention	0.88	0.84	0.83

TABLE 2: Performance Metrics for LAB

LAB: The LAB color space results indicate that the ResNet and CNN-RNN models with Attention perform better than RNN and CNN alone, highlighting the benefits of combining spatial and temporal features. The LAB color space separates lightness from color information, which can sometimes enhance model performance in identifying patterns not easily seen in RGB.

Model	**Accuracy**	**Precision**	**Recall**
CNN Attention	0.80	0.75	0.73
RNN Attention	0.72	0.68	0.68
CNN-RNN Attention	0.80	0.78	0.78
ResNet Attention	0.86	0.81	0.80

TABLE 3: Performance Metrics for Grayscale

Grayscale: In the Grayscale color space, the ResNet with Attention again shows superior performance, though the overall metrics are lower compared to the RGB color space, indicating the reduced feature richness of grayscale images. This highlights the importance of color information in achieving high classification accuracy.

The table shows detailed performance metrics across different color spaces, providing a comprehensive comparison. ResNet with Attention generally performs best, particularly in the RGB color space.

Confusion Matrices

To better understand the classification performance, confusion matrices were generated for RGB, LAB, and Grayscale models. These matrices provide insight into the types of errors made by the models.

The confusion matrices show the model performance with different color information, which helps to understand the impact of color information on classification accuracy. By

	RGB			LAB			Grayscale		
	precision	recall	f1-score	precision	recall	f1-score	precision	recall	f1-score
CLL	0.96	0.90	0.93	0.91	0.85	0.88	0.87	0.81	0.84
FL	0.95	0.99	0.97	0.91	0.96	0.94	0.88	0.95	0.91
MCL	0.95	0.95	0.95	0.93	0.90	0.92	0.88	0.88	0.88
macro avg	0.95	0.95	0.95	0.92	0.90	0.91	0.88	0.88	0.88
weighted avg	0.95	0.95	0.95	0.92	0.91	0.91	0.88	0.88	0.88
accuracy	0.95	0.95	0.95	0.91	0.91	0.91	0.88	0.88	0.88

TABLE 4: Performance Metrics Across Different Color Spaces based on decision diffusion

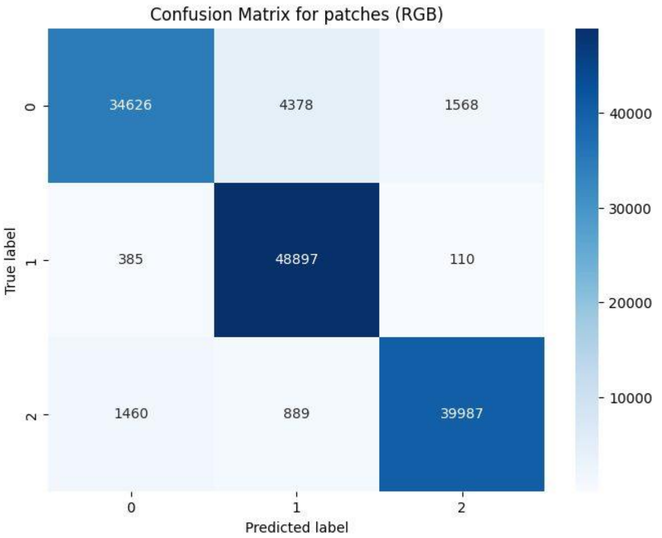

Fig. 8: Confusion Matrix for RGB Models

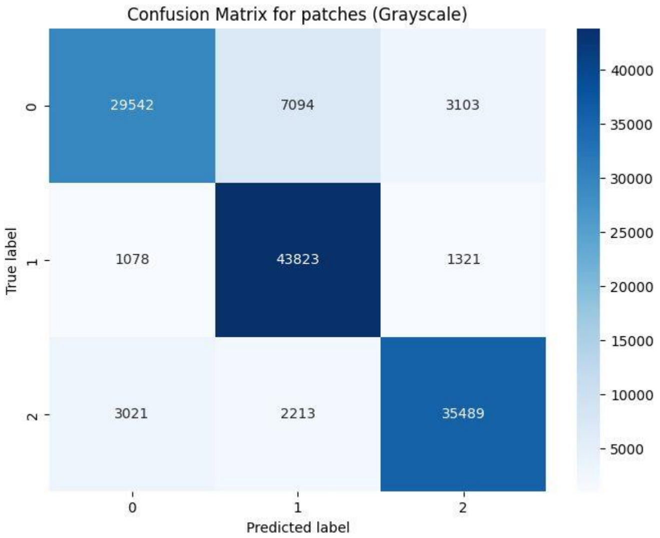

Fig. 9: Confusion Matrix for Grayscale Models

analyzing these matrices, one can identify which classes are most frequently misclassified and assess the effectiveness of the models in distinguishing between similar classes.

Comparative Analysis

The comparative analysis provides a detailed comparison of the models in terms of training time, memory usage, GPU memory usage, and power consumption across different color spaces.

This table provides a comprehensive overview of the computational resources required by each model and color space, highlighting the trade-offs between performance and resource consumption. For instance, while ResNet with RGB provides high accuracy, it also demands significant GPU memory and power, which might be a consideration for large-scale deployments.

Learning Curves

Learning curves illustrate the training and validation accuracy/loss over epochs for each model, providing insight into the training process and the model's ability to generalize. float

The learning curves for models show how the training and validation accuracy and loss change over time, indicating the model's learning progress and convergence. Observing these curves helps in understanding whether a model is overfitting, underfitting, or well-fitted to the training data.

Time and Space Domain Analysis

This section details the time taken, CPU usage, memory usage, GPU memory usage, and power consumption for each model and color space, as shown in Table 5. The analysis helps in understanding the computational efficiency and resource requirements of each model, providing valuable insights for practical deployment. For example, models with lower training times and memory usage might be more suitable for real-time applications.

Energy Efficiency Analysis

The energy efficiency analysis, including power consumption during training and inference, is summarized in Table 5. This analysis highlights the energy requirements of each model, which is crucial for assessing the operational cost and environmental impact. Energy-efficient models are preferable for large-scale deployments where operational costs and environmental concerns are significant.

Discussion

The results show that ResNet with Attention generally performs the best across all color spaces, particularly in the

Model	Color Space	AVG. Training Time (s)/ Epoch	Memory Usage (%)	GPU Memory (MB)	Power Consumption (W)
CNN with Attention	RGB	618.50	32.38	703.00	150
CNN with Attention	Grayscale	900.53	77.25	545.25	140
CNN with Attention	LAB	802.23	85.40	625.60	160
RNN with Attention	RGB	460.00	76.61	470.10	175
RNN with Attention	Grayscale	1237.66	77.06	495.30	165
RNN with Attention	LAB	1132.67	86.02	525.75	170
CNN-RNN with Attention	RGB	730.30	52.59	381.00	155
CNN-RNN with Attention	Grayscale	1572.11	77.72	512.45	145
CNN-RNN with Attention	LAB	1114.72	87.09	542.15	160
ResNet	RGB	1644.94	53.57	8319.00	200
ResNet	Grayscale	1780.00	70.25	4132.80	185
ResNet	LAB	1725.00	80.10	4561.35	190

TABLE 5: Time and Space Domain Analysis

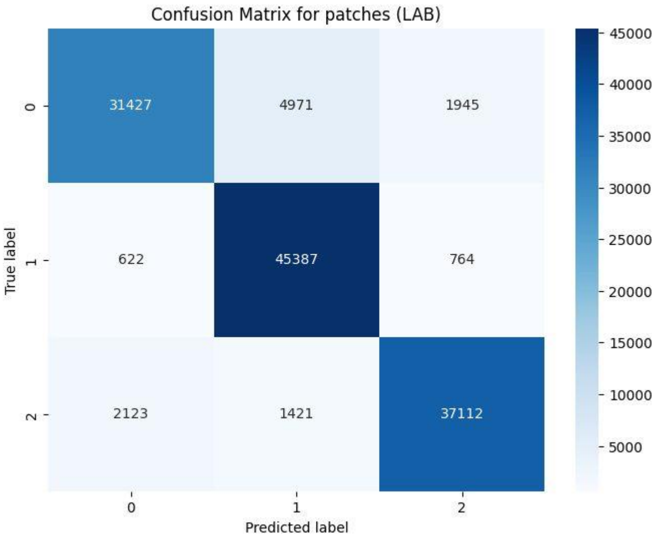

Fig. 10: Confusion Matrix for LAB Models

RGB color space. This indicates that the combination of deep residual learning and attention mechanisms is highly effective in capturing complex features and improving classification accuracy.

Comparing the color spaces, RGB generally provides the richest feature set, leading to higher performance metrics. LAB and Grayscale, while useful, tend to offer lower performance, likely due to the reduction in color information that helps in distinguishing between different classes.

In terms of computational efficiency, while ResNet offers high accuracy, it also demands the most resources, highlighting a trade-off between accuracy and computational cost. Models like CNN-RNN with Attention provide a good balance, offering reasonable accuracy with moderate resource consumption.

The confusion matrices and learning curves provide additional insights, showing that models trained on RGB data are less prone to misclassification errors and have more stable learning patterns. This reinforces the idea that the choice of color space can significantly impact the performance and efficiency of deep learning models.

Overall, these analyses provide a comprehensive understanding of the strengths and weaknesses of different models and color spaces, guiding future research and practical applications in selecting the most appropriate configurations based on specific requirements.

VII. Concluding Remarks

In this study, we evaluated the performance of various deep learning models (CNN, RNN, CNN-RNN, ResNet) using different color spaces (RGB, LAB, Grayscale). We assessed the models based on key performance metrics such as accuracy, precision, recall, training time, memory usage, GPU memory usage, and power consumption. Our findings highlight the strengths and weaknesses of each model and color space configuration.

Summary of Findings

We found that the ResNet with Attention consistently performed the best across all color spaces, especially in the RGB color space, due to its deep residual learning and attention mechanisms. The RGB color space provided the richest feature set, leading to higher performance metrics. LAB and Grayscale color spaces, while useful, generally resulted in lower performance due to reduced color information. The comparative analysis indicated that there are trade-offs between model accuracy and computational efficiency, with ResNet requiring the most resources but offering the highest accuracy.

Relevance and Applicability

The relevance of our findings lies in their applicability to real-world scenarios where choosing the right model and color space configuration is crucial. For instance, applications requiring high accuracy, such as medical image analysis or autonomous driving, might benefit from using ResNet with RGB despite its higher resource demands. On the other hand, applications with limited computational resources might opt for CNN-RNN models which offer a good balance between accuracy and efficiency.

Future Work

While this study provides valuable insights, there are several areas for future work. First, exploring other color spaces

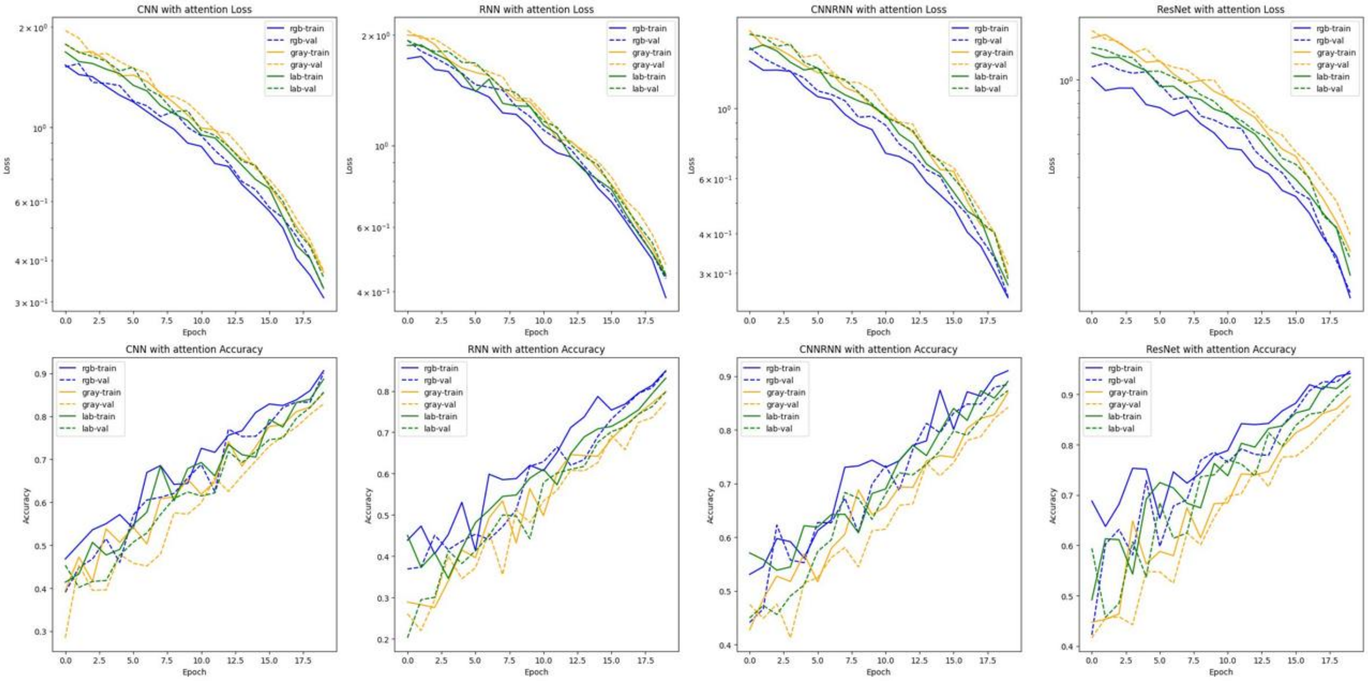

Fig. 11: Learning Curves for RGB Models

and hybrid models could uncover configurations that further enhance performance. Second, applying transfer learning techniques could reduce training times and improve model generalization. Third, expanding the dataset and including more diverse classes could provide a more comprehensive evaluation of the models' robustness.

Learnings and Challenges

Throughout this project, we learned a great deal about the impact of color spaces on deep learning model performance. We gained insights into the computational trade-offs involved in choosing different models and configurations. One of the key challenges we encountered was managing the high computational demand of training deep models, especially with large datasets. Additionally, fine-tuning hyperparameters to optimize performance required significant experimentation and validation.

In conclusion, this study highlights the importance of selecting appropriate model and color space configurations to balance performance and resource efficiency. Our findings provide a foundation for future research and practical applications in various domains.

References

[1] A. Janowczyk and A. Madabhushi, "Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases," *Journal of Pathology Informatics*, vol. 7, p. 29, July 2016.

[2] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, Dec. 2017.

[3] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in *Medical Image Computing and Computer-Assisted Intervention â€" MICCAI 2013*, (Nagoya, JP), Sept. 2013.

[4] N. Orlov, L. D. Shamir, T. Macura, J. Johnston, O. Eckley, and I. G. Goldberg, "WND-CHARM: Multi-purpose image classification using compound image transforms," *Pattern Recognition Letters*, vol. 29, pp. 1684–1693, Aug. 2010.

[5] N. Kumar, G. Litjens, C. I. Sánchez, B. van Ginneken, J. A. W. M. van der Laak, M. Balkenhol, K. H. M. Huijben, P. Bult, R. van Driel, B. H. M. van Gorp, R. M. L. M. Kötter, and F. Ciompi, "A multi-organ nucleus segmentation challenge," *IEEE Transactions on Medical Imaging*, vol. 36, pp. 1998–2012, Oct. 2017.

[6] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis," *Scientific Reports*, vol. 6, p. 26286, June 2016.

[7] G. Steinbuss, M. Kriegsmann, C. Zgorzelski, A. Brobeil, B. Goeppert, S. Dietrich, G. Mechtersheimer, and K. Kriegsmann, "Deep Learning for the Classification of Non-Hodgkin Lymphoma on Histopathological Images," *Cancers*, vol. 13, p. 2419, May 2021.

[8] H. Khelil, A. E. M. Zerari, and L. Djerou, "Accurate diagnosis of non-Hodgkin lymphoma on whole-slide images using deep learning," in *International Conference on Sciences of Electronics, Technologies of Information and Telecommunications*, (Bordeaux, FR), May 2022.

[9] N. Capobianco, M. Meignan, A.-S. Cottereau, L. Vercellino, L. Sibille, B. S. Spottiswoode, S. Zuehlsdorff, O. Casasnovas, C. Thieblemont, and I. Buvat, "Deep-Learning 18F-FDG Uptake Classification Enables Total Metabolic Tumor Volume Estimation in Diffuse Large B-Cell Lymphoma," *The Journal of Nuclear Medicine*, vol. 62, pp. 30–36, Jan. 2021.

[10] Z. Xiaoli, K. Zhang, M. Jiang, and L. Yang, "Research on the classification of lymphoma pathological images based on deep residual neural network," *Technology and Health Care*, vol. 29, pp. 53–65, Jan. 2021.

[11] H. I. Mohamed and S. Kamal, "A New Model for Blood Cancer Classification Based on Deep Learning Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 14, pp. 45–52, Jan. 2023.

[12] M. Mohana, R. Rithika, and V. Sivasakthi, "Revolutionizing Lymphoma

Diagnosis with Deep Learning and Natural Language Generation," in *TBD Conference*, (TBD), Feb. 2024.

[13] C. D. Walsh and N. K. Taylor, "Evolution of Convolutional Neural Networks for Lymphoma Classification," in *International Conference on Image Processing*, (TBD), Jan. 2021.

[14] Anonymous, "Deep learning analysis of mid-infrared microscopic imaging data for the diagnosis and classification of human lymphomas," *Posted Content*, Jan. 2023.

[15] W. Pang, H. Jiang, Y. Zhang, Y. Chen, Z. Wang, J. Guo, G. Lu, and Y. Wang, "Classification of lymphoma subtypes in PET/CT images based on a bidirectional feature fusion method," in *International Conferences on Biomedical and Bioinformatics Engineering*, (TBD), Nov. 2022.

[16] M. Tambe, T. Chavan, S. Kanchan, and S. Khanna, "Automated Classification of Lymphoma Subtypes Using Deep Neural Networks," in *International Conference on Intelligent Computing and Control Systems (ICICCS)*, (Madurai, India), May 2019.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, June 2016.

[18] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.