

BCSE498J Project II / CBS1904 Capstone Project

**AN EXPLAINABLE DEEP NEURAL NETWORK
APPROACH FOR NON HODGKIN LYMPHOMA
CLASSIFICATION**

22BCE3411 VIREN DEEPAK JAWADWAR

22BCE3762 SAHIL N

22BCE3807 H PREYANSH NAHAR

Under the Supervision of

Dr. DIVIYA M

Assistant Professor Sr. Grade 1

School of Computer Science and Engineering (SCOPE)

B.Tech.

in

Computer Science and Engineering



VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

February 2026

Abstract

Non Hodgkin Lymphoma (NHL) comprises multiple subtypes such as Mantle Cell Lymphoma (MCL), Follicular Lymphoma (FL), and Chronic Lymphocytic Leukemia (CLL), whose accurate classification is critical for effective diagnosis and treatment planning. This project proposes an explainable deep neural network-based framework for the automated classification of NHL subtypes using medical data. The primary purpose of this study is to develop a reliable and interpretable model that can assist clinicians in identifying lymphoma subtypes with improved confidence.

In the first stage, a deep learning classification model is trained to distinguish between MCL, FL, and CLL cells by learning discriminative features from the dataset. Standard preprocessing and data augmentation techniques are applied to improve robustness and generalization. The performance of the classification model is evaluated using metrics such as accuracy, precision, recall, and F1 score.

In the second stage, explainability is incorporated using Grad CAM and SHAP to interpret the predictions of the trained model. Grad CAM is used to visualize the important regions contributing to classification decisions, while SHAP provides feature level explanations that quantify the contribution of input attributes to each prediction. This two layer approach classification followed by interpretation ensures both high predictive performance and transparency of the model's decisions.

The key findings demonstrate that the proposed framework achieves effective subtype classification while providing meaningful visual and feature based explanations of model decisions. The conclusions highlight that integrating explainable artificial intelligence with deep neural networks enhances trust, interpretability, and usability of automated diagnostic systems in clinical environments, ultimately supporting better decision making in lymphoma diagnosis.

TABLE OF CONTENTS

Chapter No.	Contents	Page No.
	Abstract	
1.	INTRODUCTION	
	1.1 BACKGROUND	
	1.2 MOTIVATIONS	
	1.3 SCOPE OF THE PROJECT	
2.	PROJECT DESCRIPTION AND GOALS	
	2.1 LITERATURE REVIEW	
	2.1.1 NHL BASED	
	2.1.2 DEEP LEARNING BASED	
	2.1.3 EXPLAINABLE AI BASED	
	2.2 GAPS IDENTIFIED	
	2.3 OBJECTIVES	
	2.4 PROBLEM STATEMENT	
	2.5 PROJECT PLAN	
	2.5.1 GANTT CHART	
	2.5.2 WORK BREAKDOWN STRUCTURE	
	2.5.3 ACTIVITY CHART	
3.	TECHNICAL SPECIFICATION	
	3.1 REQUIREMENTS	
	3.1.1 Functional	

	3.1.2 Non Functional	
	3.2 FEASIBILITY STUDY	
	3.2.1 Technical Feasibility	
	3.2.2 Economic Feasibility	
	3.2.2 Social Feasibility	
	3.3 SYSTEM SPECIFICATION	
	3.3.1 Hardware Specification	
	3.3.2 Software Specification	
4.	DESIGN APPROACH AND DETAILS	
	4.1 SYSTEM ARCHITECTURE	
	4.2 DESIGN	
	4.2.1 Data Flow Diagram	
	4.2.2 Class Diagram	
5.	METHODOLOGY AND TESTING	
	5.1 CODE	
	5.2 RESULT	
	REFERENCES	

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

Non Hodgkin Lymphoma (NHL) is a major category of haematological malignancies affecting the lymphatic system and requires accurate subtype classification for effective diagnosis and treatment. The field of medical image analysis and computational pathology has increasingly adopted artificial intelligence techniques to support clinical decision making through automated and data driven approaches.

Traditionally, NHL diagnosis has been based on histopathological examination and immunophenotyping performed by expert pathologists. With the advancement of digital pathology and the availability of large medical datasets, deep learning models have been introduced to automatically extract discriminative features for lymphoma classification. Several existing studies have demonstrated the effectiveness of deep neural networks in distinguishing subtypes such as Mantle Cell Lymphoma (MCL), Follicular Lymphoma (FL), and Chronic Lymphocytic Leukaemia (CLL).

However, most current deep learning-based solutions operate as black box systems, limiting their clinical acceptance due to the absence of interpretability. Recent developments in Explainable Artificial Intelligence (XAI), including Grad CAM and SHAP, provide mechanisms to visualize and quantify model decisions. These techniques enable transparent and reliable classification frameworks, forming the foundation for the proposed explainable deep neural network approach for NHL subtype classification.

1.2 MOTIVATIONS

The motivation for this project arises from the critical need for accurate and reliable classification of Non Hodgkin Lymphoma (NHL) subtypes to support early diagnosis and effective treatment planning. Manual diagnostic procedures are time consuming and subject to inter observer variability, which can lead to inconsistent clinical outcomes. An automated and intelligent classification system can assist medical professionals by improving efficiency and diagnostic precision.

From an academic perspective, this project is motivated by the growing research interest in applying deep learning techniques to medical diagnosis and by the necessity to address the challenge of interpretability in artificial intelligence models. While deep neural networks achieve high classification accuracy, their black box nature limits trust and adoption in clinical practice.

Furthermore, this work aligns with current trends in Explainable Artificial Intelligence (XAI), which emphasizes transparency and accountability in healthcare applications. By integrating classification with explainability using Grad CAM and SHAP, this project aims to contribute to the development of trustworthy and clinically meaningful AI based diagnostic systems for lymphoma classification.

1.3 SCOPE OF THE PROJECT

The scope of this project is limited to the development of an explainable deep neural network framework for the classification of selected Non Hodgkin Lymphoma subtypes, namely Mantle Cell Lymphoma (MCL), Follicular Lymphoma (FL), and Chronic Lymphocytic Leukaemia (CLL). The project focuses on training and evaluating a deep learning model to perform multi class classification and integrating explainability techniques to interpret model predictions.

The project includes data preprocessing, model training, performance evaluation using standard classification metrics, and implementation of explainable artificial intelligence methods such as Grad CAM and SHAP for result interpretation. The system is intended as a decision support tool and not as a replacement for clinical diagnosis.

The project excludes real time clinical deployment, treatment recommendation systems, and analysis of other lymphoma subtypes beyond MCL, FL, and CLL. Limitations include dependence on the quality and size of the available dataset, computational constraints, and the potential variability in performance across different data sources. Additionally, the explainability results are restricted to the interpretability methods employed and may not capture all aspects of clinical reasoning.

CHAPTER 2

PROJECT DESCRIPTION AND GOALS

2.1 LITERATURE REVIEW

2.1.1 NHL BASED

No.	Title (APA)	Journal / Year	Method	Result (headline)	Dataset(s)	Pros	Cons	Future Work
1	Khelil, H., Zerari, A. E. M., & Djerou, L. (2022). Accurate diagnosis of non-Hodgkin lymphoma on whole-slide images using deep learning.	IEEE SETI T Conference / 2022	Improved CNN-based deep learning model for multi-class NHL classification	Achieved 98.7% classification accuracy	NIA-curated histopathology whole-slide image dataset	High accuracy and effective multi-class classification	Limited to a single dataset and lacks explainability	Integrative explainable AI techniques and validate on larger multi-center datasets
2	D. S. & Sekhar, M. C. (2025). Non-Hodgkin Lymphoma Classification Using Multi-Scale Attention Mechanism with Convolutional Neural Networks.	ICD SNS Conference / 2025	Multi-Scale Attention Mechanism integrated with CNN and GLCM feature extraction	Achieved 99.21% accuracy, outperforming DCGAN and SampEN approaches	Histopathological NHL image dataset with preprocessing and augmentation	High accuracy, attention-based feature learning, robust preprocessing pipeline	Computational complexity and lack of explainability analysis	Integrative explainable AI techniques and evaluate performance on larger and diverse clinical datasets

3	Sri, G. P., & Ananthajothi, K. (2025). Enhanced malignant lymphoma classification using an explainable Dilated MobileNet V2 and convolutional-recurrent neural network.	ICC SP Conference / 2025	Explai nable Dilated Mobile NetV2 combin ed with Convolutio nal - Recurr ent Neural Networ k (CRN N) for lympho ma classifi cation	Achiev ed high classifi cation accurac y with improv ed interpretabilit y	Publicly available malignan t lymphoma image datasets	Interpre table model, efficient feature extracti on, improve d clinical applicab ility	Evaluate d on limited datasets and lacks compariso n with multiple XAI techniqu es	Extend evaluati on to larger multi-center datasets and integrat e advanc ed explain able AI method s for deeper clinical insights
4	H. N., N. K., H. S., B., Junaid, M., S., S., S., R., & S., K. S. (2025). Comprehensive study on lymphoma detection using deep learning.	ICIC I Conference / 2025	System atic survey of deep learnin g models for lympho ma subtyp e detecti on and classifi cation from histopa thologi cal images	Summa rized state-of-the-art methods, benchm ark datasets , and researc h gaps in lympho ma classifi cation	Publicly available histopath ological lymphom a benchmark datasets	Compre hensive review, identifie s research gaps, compar es methods and datasets	Does not propose a new classific ation model or experim ental framew ork	Develo p novel explain able deep learnin g architec tures and validate them on diverse multi- class lympho ma datasets

5	Kumar, A., Nelson, L., & Arumugam, D. (2024). Blood cancer diagnosis using pretrained VGG16 transfer learning model with lymphoma dataset.	ACO IT Conference / 2024	Transfer learning using pretrained VGG16 model for lymphoma subtype classification	Achieved 96.19% validation accuracy and 95.20% test accuracy	Kaggle histopathological image dataset (CLL, FL, MCL)	Effective use of transfer learning, good accuracy, simple and efficient architecture	Limited to one pretrained model and lacks explainability analysis	Integrative explainable AI techniques and evaluate performance on larger and more diverse clinical datasets
6	Habijan, M., & Galić, I. (2024). Ensemble transfer learning for lymphoma classification.	IWS SIP Conference / 2024	Ensemble transfer learning using VGG-19, DenseNet201, MobileNetV3, and ResNet50V2 CNN models	Achieved 98.89% accuracy using ensemble model, outperforming individual networks	Histopathological lymphoma image dataset (CLL, FL, MCL)	High accuracy through ensemble learning, comparison of multiple pretrained CNNs	Increased computational complexity and lack of explainability analysis	Integrative explainable AI methods and optimize ensemble models for real-time clinical deployment

7	Riyanto, A. A. R., et al. (2024). Lymphoma sub-type classification using DenseNet169-based transfer learning architecture.	iSemantic Conference / 2024	Transfer learning using DenseNet169 with image augmentation for lymphoma subtype classification	Achieved 99.1% accuracy, outperforming previous studies	Digital pathology dataset (CLL, FL, MCL: 113, 139, 122 images) with augmentation	High accuracy on limited data, effective use of augmentation and transfer learning	Small dataset size and absence of explainability analysis	Apply explainable AI methods and validate the model on larger and multi-institutional datasets
8	Maghdid, S. S., Al-Atroshi, S. J., Salh, C. H., & Kareem, S. W. (2024). Deep learning for the detection of skin cancer: A comprehensive review.	IEC Conference / 2024	Review of deep learning methods (CNNs, RNNs) for medical image analysis	A summary of methods, challenges, advancements, and interpretability in skin cancer detection	Various dermatological image datasets	Comprehensive review, covers multiple architectures, emphasizes interpretability	Not specific to lymphoma; no experimental results for lymphoma	Apply similar deep learning and explainability methods to lymphoma classification tasks

9	Mohana, M., Rithika, R., & Sivasakthi, V. (2024). Revolutionizing lymphoma diagnosis with deep learning and natural language generation.	IC4 Conference / 2024	Ensemble of CNNs (VGG19, ResNet 50v2, MobileNetV3, DenseNet201) combined with GPT-2 for diagnostic report generation	Achieved improved classification accuracy and generated explanatory diagnostic reports	H+E-stained lymphoma biopsy image dataset	High accuracy, ensemble leverages strengths of multiple CNNs, provides automated report generation	Complex architecture, computationally intensive, limited dataset	Integrate broader datasets and extend multimodal approach to other oncological diagnoses
10	Kaur, A., Kukreja, V., Thapliyal, N., Manwal, M., & Sharma, R. (2024). An efficient fine-tuned GoogleNet model for multiclass classification of blood cell cancer.	IAT MSI Conference / 2024	Fine-tuned Google Net model for multiclass blood cell cancer classification	Achieved 99.83% accuracy with low loss at epoch 15	Broad blood cell image dataset	High precision, effective feature extraction, potential for clinical use	Limited to blood cell cancers, no explainability analysis	Apply explainable AI methods and extend to other hematologic malignancies and lymphoma subtypes

1 1	Aravind, K. R., Kumar, K. A., Reddy, V. V. D., & Yadlapalli, P. (2024). Leveraging convolutional neural networks for classifying lymphoma using histopathological images.	ICIT EIC S Conference / 2024	Customized pre-trained CNN model for lymphoma classification	Achieved 96.2% accuracy and 96.2% sensitivity on histopathology images	Multi Cancer Dataset (15,000 lymphoma images, 512×512 pixels)	High accuracy and sensitivity, effective for histopathologic diagnosis	Limited to one dataset, lacks explainability analysis	Apply explainable AI methods and validate on diverse multi-center lymphoma datasets
1 2	M., S., D., P. K., & P., S. (2023). Data mining approaches in healthcare industry.	ICIR CA Conference / 2023	Machine learning and data mining techniques with Min-Max normalization for lymphoma diagnosis	Improved lymphoma diagnosis accuracy using ML and data mining methods	Healthcare datasets including lymphoma-related patient data	Enhances diagnostic accuracy, low-cost approach, effective handling of complex data	Limited focus on deep learning or histopathology images	Integrate deep learning and explainable AI methods for automated lymphoma subtype classification

1 3	Gupta, R. K., et al. (2024). Classification and morphological analysis of DLBCL subtypes in H&E-stained slides.	BIB E Conference / 2024	Deep learning model for classification of DLBCL subtypes (ABC vs. GCB) with morphological feature analysis	Achieved average AUC of 87.4% ± 5.7%, high PPV, robust subtype classification	H&E-stained DLBCL histopathology slides	Enables precise subtype classification, includes morphological insights	Subtle visual differences make classification challenging, limited dataset	Extend model to larger multi-center datasets and integrate explainable AI for clinical interpretability
1 4	Archita, C., Prabha, C., Nath, A., & Singh, R. (2024). Enhancing lymphoma diagnosis: Transfer learning with DenseNet201 for subtype classification.	ICA AIC Conference / 2024	Transfer learning using DenseNet201 for classifying lymphoma subtypes (MCL, FL, CLL)	Achieved 98.53% accuracy and F1 score of 0.98	Blood cancer image dataset (13,500 images; training 10,800, validation 2,700, testing 1,500)	High accuracy, balanced precision and recall, robust model	Limited to one dataset, lacks interpretability	Integrate explainable AI and validate on larger multi-center datasets for clinical deployment

1 5	Jusman, Y., Ningrum, R. O., & Nurkholid, M. A. F. (2023). Leukemia cell image classification using CNN: AlexNet and GoogLeNe t.	ICE EIE Conf erenc e / 2023	CNN models (AlexN et, GoogL eNet) for leukem ia cell image classifi cation	Achiev ed 100% accuracy in training runs; AlexNe t faster in training than GoogL eNet	Leukemi a cell image datasets	High accurac y, efficient classific ation, compre hensive evaluati on metrics (precisi on, sensitivi ty, specifici ty, F- score)	Limited to leukemi a dataset, no explaina bility, not directly lympho ma related	Validat e on larger datasets and explore applica bility to lympho ma subtype classifi cation
--------	---	--	---	--	--	---	--	--

2.1.2 DEEP LEARNING BASED

No.	Title (APA)	Journal / Year	Method	Result (headline)	Dataset(s)	Pros	Cons	Future Work
16	M. Pasha, K. K. ATA and V. V. Kishore (2025). Optimized Ensemble Learning for Lung and Colon Cancer Classification using Histopathology Images from LC25000 Dataset. 2025 International Conference on Electronics and Renewable Systems (ICEARS), pp. 1874-1879.	ICE ARS / 2025	Ensemble deep learning (VGG16+ResNet50, VGG16+EfficientNetB0, ResNet50+EfficientNetB0)	High accuracy, precision, recall, F1-score (0.98 each)	LC 25000	Robust and adaptable; handles limited data	Reduced dataset; needs full-scale clinical validation	Further clinical validation with full dataset
17	H. H. Hai, S. Manoharan, S. D. M and K. Raja (2025). An Explainable Transfer Learning Framework for Breast Cancer Histopathology Classification Using CNNs. IEEE 7th International Conference on Computing, Communication and Automation (ICCCA), pp. 1-7.	ICC CA / 2025	Transfer learning with pre-trained CNNs (VGG16, ResNet 50, InceptionV3) and Grad-CAM	ResNet50 achieved 95.4% accuracy; Grad-CAM improves interpretability	Break His	Explainable model; effective automated classification	Black-box CNN nature remains a challenge	Enhanced interpretability; test on larger, multi-source datasets

1 8	P. Subramaniyam & G. H. Krishnan (2025). Fusion Learning Framework for Malignant and Benign Classification in Histopathology Images. 3rd International Conference on Intelligent Data Communication Technologies and IoT (IDCIoT), pp. 1692-1696.	IDCIoT / 2025	Ensemble CNN (EfficientNetB0 + ResNet 50, EfficientNetB0 + VGG16, VGG16 + ResNet 50)	High accuracy, precision, recall, F1-score; EfficientNetB0 + VGG16 ensemble best	Break His	Combines complementary CNNs; robust classification	Needs large-scale validation	Optimize architectures; larger dataset validation
1 9	G. Kaur & N. Sharma (2024). Invasive Ductal Carcinoma Detection Using CNN Architecture from Breast Histopathology Images. Second International Conference on Intelligent Cyber Physical Systems and IoT (ICoICI), pp. 1470-1475.	ICoICI / 2024	CNN with convolution, pooling, batch norm, dropout	Testing accuracy 94.73%; training accuracy 99.12%	24, 678 breast histopathology images	High precision and recall; robust IDC detection	Moderate testing accuracy; small augmentation scope	Expanded dataset; improve augmentation techniques

2 0	V. P. V, P. Chattu, K. Sivasankari, D. T. Pisal, B. Renuka Sai & D. Suganthi (2024). Exploring Convolution Neural Networks for Image Classification in Medical Imaging. International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE), pp. 1-4.	IITC EE / 2024	CNNs, transfer learning, custom network	High classification accuracy: X-rays 95.2%, CT 92.7%, MRI 94.1%, histopathology 88.6%	X-ray s, CT, MR I, histopathology slides	Supports multiple modalities; uses transfer learning; interpretable via gradient visualization	Lower accuracy on histopathology; ethical/regulator y concern	Enhanc e histopatholog y performance; address privacy and compli ance
2 1	M. Jia, X. Yan & S. Fu (2019). Histopathologic Cancer Detection Based on Deep Multiple Instance Learning. 15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN), pp. 368-371.	MSN / 2019	Deep multiple instance learning (DCNN + MIL)	Higher accuracy than conventional methods	His top atholo gy ima ges	Handles complex high-resolution images	Requires multiple instance setup; computationally intensive	Test on larger datasets; optimize MIL approach

2 2	T. D. Pham (2025). Class Fusion of Support Vector Machines with Deep Learning Features for Oral Cancer Histopathology Classification. IEEE 22nd International Symposium on Biomedical Imaging (ISBI), pp. 1-4.	ISBI / 2025	Feature extraction from InceptionResNet-v2 and ViT; SVM fusion	Fusion improves balance; accuracy, precision, sensitivity, AUC	Oral cancer histopathology images	Addresses class imbalance; combines complementary classifiers	Complex fusion process; needs computational resources	Expanded to other cancer types; optimize fusion strategy
2 3	Y. Yari & H. Nguyen (2020). A State-of-the-art Deep Transfer Learning-Based Model for Accurate Breast Cancer Recognition in Histology Images. 20th International Conference on Bioinformatics and Bioengineering (BIBE), pp. 900-905.	BIBE / 2020	Transfer learning with DenseNet121 pre-trained on ImageNet; fine-tuned classifier	Multi-class accuracy 97%; binary classification accuracy 100%	Breast histology images / Break His	High performance; flexible and scalable	Dependent on ImageNet weights; needs domain-specific fine-tuning	Integrate other CNNs; test on other diseases

2 4	S. S. M. Khairi, M. A. A. Bakar, M. A. Alias, S. A. Bakar & C. -Y. Lioong (2021). A Preliminary Study of Convolutional Neural Network Architectures for Breast Cancer Image Classification. IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), pp. 1-5.	CSD E / 2021	Compare CNN architectures (AlexNet, Google Net, ResNet 18)	ResNe t18 achieved 94.8% accuracy; fastest training 70 min 31 sec	Bre ak His	Identifie s effective architecture; fast computation	Focused on limited CNNs; moderate dataset size	Test more architectures; increase dataset diversity
2 5	G. Kaur, N. Sharma & R. Gupta (2024). Enhanced Detection of Oral Squamous Cell Carcinoma from Histopathological Images Using CNN Architectures. Second International Conference on Intelligent Cyber Physical Systems and IoT (ICoICI), pp. 1525-1530.	ICoI CI / 2024	Customized CNN with data augmentation (rescaling, shearing, zooming, flipping)	Validation accuracy 86.67%; training 94.64%; test accuracy 72%	OS CC hist opathology images	Supports early detection; assists pathologists	Moderate test accuracy; small dataset	Further optimization; larger dataset ; improve generalization

2 6	Quinones, W. R., Ashraf, M., & Yi, M. Y. (2021). Impact of Patch Extraction Variables on Histopathological Imagery Classification Using Convolution Neural Networks. 2021 International Conference on Computational Science and Computational Intelligence (CSCI), 1176–1181.	2021	DenseNet CNN, patch extraction	Analyzed impact of patch size and magnification on histopathology classification	Stomach histopathology images	Guides patch extraction for WSI; improves model performance understanding	Limited to stomach images; only DenseNet	Extend to other cancer types and networks; larger datasets
2 7	Nguyen, P. T., Nguyen, T. T., Nguyen, N. C., & Le, T. T. (2019). Multiclass Breast Cancer Classification Using Convolutional Neural Network. 2019 International Symposium on Electrical and Electronics Engineering (ISEE), 130–134.	2019	CNN, image resizing	Multi-class classification of 8 breast cancer classes	Breast Hist (7, 909 images)	Handles 8-class classification; reduces human error	Limited preprocessing; fixed CNN architecture	Use transfer learning; larger datasets; data augmentation

2 8	P., S., V, S. K., & M. S., C. M. S. (2025). Deep Learning Based Multi Class Epithelial Ovarian Cancer Classification From Histopathological Images. 2025 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), 1-6.	2025	ResNet 50, VGG19, DenseNet121 CNNs	Accur acy: 87%, (VGG 19), 96% (ResNet50), 73% (DenseNet121)	0 ova rian can cer hist opathology ima ges	High accuracy with ResNet50; multiple architectures evaluated	Smaller dataset; ovarian cancer only	Expan ded dataset ; ensemble methods; other cancer types
2 9	Duan, H., Liu, Y., Yan, H., He, Q., He, Y., & Guan, T. (2022). Fourier ViT: A Multi-scale Vision Transformer with Fourier Transform for Histopathological Image Classification. 2022 7th International Conference on Automation, Control and Robotics Engineering (CACRE), 189-193.	2022	Fourier ViT, mixed attention	Outper forms classic CNNs and Transformers in fine-grained histopathology classification	Not specific ed	Combin es ViT and Fourier transform; better attention to detail	Needs more datasets ; computationally heavy	Apply to larger datasets; optimize computational efficiency

30	Mukadam, S. B., & Patil, H. Y. (2024). Fusion of ESRGAN, Adaptive NSCT, and Multi-Attention CNN With Wavelet Transform for Histopathological Image Classification. IEEE Access, 12, 129977–129993.	2024	ESRGA N + NSCT + Multi- Attention CNN + Wavelet	Accur acy 93.67 %, AUC 94%	PC am (32 7,6 80 ima ges)	Super- resolutio n + attention improve s feature extractio n	Comple x pipeline ; high computa tional cost	Further optimi ze attentio n mecha nisms; test other cancer types
31	Johny, A., Madhusoodanan, K. N., & Cyriac, S. (2022). Edge Computing Based Miniature Maps Using Embedded Webserver For Prediction of Malignancy. 2022 6th International Conference on Devices, Circuits and Systems (ICDCS), 268–271.	2022	Miniatu re probabil ity maps + CNN on edge devices	Faster on- device predict ion of malign ancy	His top ath olo gy WS I ima ges	Reduces latency; portable device deploym ent	Edge hardwar e limitatio ns; smaller models	Optimi ze model for more comple x WSI; real- time applica tions

3 2	Venugopal, D., Mesbah, M., Hmouz, R., Qureshi, A., & Ammari, A. (2025). Comparative Analysis of Diverse Activation Functions and Kernel Methods for Thyroid Cancer Classification. 2025 International Conference on Emerging Technologies in Engineering Applications (ICETEA), 1–6.	2025	CNNs + SVMs, multiple activation functions and kernels	CNN (ResNet50+ReLU) outperforms SVMs for thyroid cancer	Thyroid histopathology images	Sys. comparison of activation/kernel s; high accuracy	Single-institution dataset	Expanded dataset ; optimize hybrid architectures
3 3	Swarnkar, B., Maheshwari, P., Khare, N., & Gyanchandani, M. (2024). Early Diagnosis of Endometrial Cancer: An Ensemble-Based Deep Learning Approach. 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), 1–7.	2024	Ensemble of VGG16, DenseNet121, MobileNetV2	Accuracy 97.17 %, F-score 96.79 %	330 endomtrial histopathology images	High accuracy ; ensemble approach improves robustness	Relatively small dataset; limited generalization	Use larger datasets; optimize ensemble weightings

3 4	Varman, U., Bharti, V., Sharma, A., Kumar, A., & Singh, S. K. (2025). When Explainability Meets Vision AI: Analyzing CNNs, Transformers, and State-Space Models in Healthcare. 2025 International Joint Conference on Neural Networks (IJCNN), 1–8.	2025	CNNs, Transformers, State-space models, Grad-CAM/Attention explainability	Trade-offs between accuracy, efficiency, and interpretability	Breast Cancer X-ray, Retinal images	Explains model decisions; multiple datasets evaluated	Complex; computational complexity intensive	Further optimize explainability; integrate more imaging modalities
3 5	Yari, Y., Nguyen, T. V., & Nguyen, H. T. (2020). Deep Learning Applied for Histological Diagnosis of Breast Cancer. IEEE Access, 8, 162432–162448.	2020	Transfer learning : ResNet 50, DenseNet121	Binary : 100% accuracy, Multiclass: 98% accuracy	Breast Cancer His	High accuracy ; magnification-dependent & independent evaluation	Large model sizes; requires data augmentation	Expanded to other cancers ; optimize for edge deployment

3 6	Bhatt, N., Goswami, S., K., P., Swati, D., Srivalli, D., & Singh, S. (2025). Lightweight Deep Learning Model for Breast Cancer Malignancy Prediction Using CNN on Embedded Edge Devices. 2025 World Skills Conference on Universal Data Analytics and Sciences (WorldSUAS), 1–6.	2025	Lightweight CNN for edge devices	Accur- acy: max specifi- city 97.1% , negati- ve predic- tive value 96.8%	Not spe- cifi- ed	Real- time on- device inferenc- e; low latency	Limited model comple- xity; small datasets	Further test on large dataset s; optimi- ze lightwe- ight archite- cture
3 7	Singla, S., & Gupta, R. (2024). Optimizing Cancer Detection: A CNN Approach for Lung and Colon Cancer. 2024 4th International Conference on Ubiquitous Computing and Intelligent Information Systems (ICUIS), 62–67.	2024	CNN with dense layers, pooling	Improve- d classifi- cation of benign vs malign- ant tissue	Lu- ng and col- on histo- opa- tholo- gy ima- ges	Robust model; improve- s patholog- ist workflo- w	Single- instituti- on; limited generaliz- ation	Expan- d dataset ; integra- te with clinical workfl- ow

3 8	Qian, X., et al. (2024). SPCB-Net: A Multi-Scale Skin Cancer Image Identification Network Using Self-Interactive Attention Pyramid and Cross-Layer Bilinear-Trilinear Pooling. <i>IEEE Access</i> , 12, 2272–2287.	2024	SPCB-Net (ResNet 101) + SAP + cross-layer pooling	Accuracy: 97.10% (HAM 10000), 99.87% (NCT-CRC-HE-100K)	HAM10000, NC-T-CR-HE-100K	Multiscale fusion; state-of-the-art performance	Complex architecture; high computational cost	Optimize network for speed; apply to more dermatology datasets
3 9	Vanitha, K., et al. (2024). Attention-Based Feature Fusion With External Attention Transformers for Breast Cancer Histopathology Analysis. <i>IEEE Access</i> , 12, 126296–126312.	2024	External Attention Transformer (EAT)	Accuracy: 99%	BreastKHis	High accuracy; efficient transformer model	Transformer may need large compute	Further test on other datasets; optimize for edge deployment

2.1.3 EXPLAINABLE AI BASED

No.	Title (APA)	Journal / Year	Method	Result (headline)	Dataset(s)	Pros	Cons	Future Work
40	Agbley, B. L. Y., et al. (2024). Federated Fusion of Magnified Histopathological Images for Breast Tumor Classification in the Internet of Medical Things. IEEE Journal of Biomedical and Health Informatics, 28(6), 3389-3400.	IEEE J. Bio med. Heal th Infor m. / 2024	Federated Learning with residual network and multi-magnification fusion	Outperfor ms centr alize d learn ing and existi ng meth ods	Bre akH is	Priv acy-pres erving, high accu racy, expl aina ble via Grad - CA M	Requi res comp utatio nal resour ces for FL, compl ex integr ation	Extend to other cancers and real-time IoMT deployment
41	Das, I., et al. (2025). Improving Medical X-Ray Imaging Diagnosis With Attention Mechanisms and Robust Transfer Learning Techniques. IEEE Access, 13, 159002-159027.	IEEE Acce ss / 2025	EfficientNet B0 + CBAM with preprocessing and geometric augmentation	High test accur acies : 98.0 9%-99.5 1% acros s datas ets	Frac Atla s, Kne e, Lun g X-ray	Rob ust to nois e, inter pretabe le, high gene raliz ation	Comp lex preprocessin g, transf ormer comp arison comp utatio nally expen sive	Expand dataset diversit y and clinical validati on

4 2	Varam, D., et al. (2023). Wireless Capsule Endoscopy Image Classification: An Explainable AI Approach. IEEE Access, 11, 105262-105280.	IEEE Access / 2023	Transfer learning models + XAI techniques	Highest reported F1-score : 97% for Vision Transformer	Kvasir-capsule	Explainable AI, high accuracy, multiple models tested	Focus on top 9 classes only, limited dataset size	Extend to full dataset and real-world clinical testing
4 3	Shakor, M. Y., & Khaleel, M. I. (2025). Modern Deep Learning Techniques for Big Medical Data Processing in Cloud. IEEE Access, 13, 62005-62028.	IEEE Access / 2025	CNN, RNN in cloud-based big data analytics	20% improved diagnostic s, reduced computation time	Healthcare big medical data sets	Scalable, integrated DL with cloud, high diagnostic improvement	Complexity in real-time deployment	Apply federated learning, edge computing, explainable AI

4 4	Rehman, Z. U., et al. (2025). Efficient and Interpretable Otoscopic Image Classification via Distilled CNN With Adaptive Channel Attention. IEEE Access, 13, 151082-151096.	IEEE Access / 2025	Knowledge distillation + ACA-enhanced EfficientNet-B0	Accuracy 98.75%, AUC 0.992-1.0	Otoscopy image data set	Lighweight, interpretable, clinically relevant features	Requires teacher model training	Extend to other ear pathologies and telemedicine applications
4 5	ELwahsh, H., et al. (2025). Explainable Artificial Intelligence in Malignant Lymphoma Classification: Optimized DenseNet121 Deep Learning Approach With Particle Swarm Optimization and Genetic Algorithm. IEEE Access, 13, 98639-98655.	IEEE Access / 2025	DenseNet121 + PSO + GA for feature reduction and optimization	Accuracy 96.77%, high precision and F1 scores	H&E lymphoma biopsy images	Explainable AI, resouce-efficient, high accuracy	Requires optimization algorithms tuning	Apply to other cancer types, improve generalizability, real-time CAD integration

4 6	Kothadiya, D. R., et al. (2023). SignExplainer: An Explainable AI-Enabled Framework for Sign Language Recognition With Ensemble Learning. IEEE Access, 11, 47410-47419.	IEEE Access / 2023	Attention-based ensemble learning with ResNet50 + Self Attention	Accuracy 98.20%	Sign language data sets	Interpretable ensemble, high accuracy, visual relevance explanation	Focused on sign language only	Adapt for other CV tasks and larger gesture datasets
4 7	Inamdar, M. A., et al. (2025). A Dual-Stream Deep Learning Architecture With Adaptive Random Vector Functional Link for Multi-Center Ischemic Stroke Classification. IEEE Access, 13, 46638-46658.	IEEE Access / 2025	Dual-stream CNN with dual attention and ARVFL layer	Accuracy 98.83% (single-center), 92.42% (multi-center)	CT stroke images	High accuracy, computational efficiency, explainable	Multi-center generalization slightly lower	Apply to MRI data, improve multi-center generalization, integrate XAI for clinical deployment

4 8	Bissoonauth-Daiboo, P., et al. (2025). Exploring Vision Transformers and Explainable AI for Enhanced Artefact Classification in Esophageal Endoscopic Images. IEEE Access, 13, 176221-176244.	IEEE Access / 2025	Vision Transformer (ViT) + XAI for artefact classification	Accuracy 93.46%, surpasses CNN baseline	Eso phageal end oscopy images	High interpretability, outperforms CN N, handles artefacts	Focused on colour misalignment only	Extended to other artefact types and real-time endoscopy
4 9	Radhakrishnan, M., et al. (2024). Advancing Ovarian Cancer Diagnosis Through Deep Learning and eXplainable AI: A Multiclassification Approach. IEEE Access, 12, 116968-116986.	IEEE Access / 2024	Multiple DL models + XAI (Grad-CAM, DeepLift, etc.)	Highest accuracy 97.96% using Inception V3	Ovarian cancer histopathology	High accuracy, interpretable, multiclass classification	Focused on subtleties only	Includes larger datasets, clinical validation, other modalities

5 0	Saraswat, D., et al. (2022). Explainable AI for Healthcare 5.0: Opportunities and Challenges. IEEE Access, 10, 84486-84517.	IEEE Access / 2022	Survey on EXAI in Healthcare 5.0	Framework for EXAI validation in CT and ECG demonstrated	Healthcare systems	Provides architecture, private-preserving, operational guidance	Primarily conceptual, less experimental	Apply framework to real clinical deployments, expand datasets
5 1	Sangnark, S., et al. (2024). Explainable Multi-Modal Deep Learning With Cross-Modal Attention for Diagnosis of Dyssynergic Defecation Using Abdominal X-Ray Images and Symptom Questionnaire. IEEE Access, 12, 78132-78147.	IEEE Access / 2024	Multi-modal DL + cross-modal attention + XAI	Accuracy 82.27%, output for ms singl e-modal & human experts	Abdominal X-rays + symptom questionair e	High sensitivity and interpretability, aids clinicians	Lower absolute accuracy vs. other tasks	Extend to other gastrointestinal disorders, optimize multi-modal fusion

5 2	Karthik, R., et al. (2024). An Explainable Deep Learning Network for Environmental Microorganism Classification Using Attention-Enhanced Semi-Local Features. IEEE Access, 12, 151770-151784.	IEEE Access / 2024	DenseNet-169 + EHCR, DASA, NA blocks + Grad-CAM	Accuracy 82.74%	EM DS-6 environment mental microbial organic dataset	Semi-local attention improves feature extraction, interpretable	Model rate accuracy, limited dataset	Scale to larger microbial datasets, optimize attention modules
5 3	Amin, A., Hasan, K., & Hossain, M. S. (2025). XAI-Empowered MRI Analysis for Consumer Electronic Health. IEEE Trans. on Consumer Electronics, 71(1), 1423-1431.	IEEE Trans. on Consum. Electron. / 2025	PIDL + deep learning + XAI for MRI analysis	Precision 96%	Brain tumor or MR I images	Integrate phys ics principle s, interpretable, consumer- health oriented	Small dataset, limited to MRI brain tumors	Apply to other imaging modalities and larger datasets, real-world validation

5 4	Aravinda, C. V., Joseph, E. R., & Alasmari, S. (2025). Optimized DenseNet121 and Quantum PennyLane Fusion for Explainable Skin Disease Recognition and Classification. IEEE Access, 13, 167957-167969.	IEEE Access / 2025	DenseNet1 + quantum classifier + XAI (SHAP, LIME, Grad-CAM, XRAI)	Validation accuracy 91.5 %	Skin lesion images (770)	Hybrid classification approach, interpretable, high trust	Relatively small dataset, quant um component complex	Scale dataset, optimize quantum fusion, extend to other skin conditions
5 5	Haque, F., et al. (2025). An End-to-End Concatenated CNN Attention Model for the Classification of Lung Cancer With XAI Techniques. IEEE Access, 13, 96317-96336.	IEEE Access / 2025	Concatenated CNNs + MLP + multi-head attention + XAI	Accuracy 99.5 4%, F1 99.6 6%, AUC 99.9 7%	Lung cancer CT images	Highly accurate, interpretable, high light s critical regions	Complex architecture, high computational demand	Deploy in real-time clinical workflows, extend to multi-class lung diseases

2.2 GAPS IDENTIFIED

- Limited Interpretability in Existing Models :

Most deep learning models developed for lymphoma classification achieve high predictive accuracy but lack interpretability, which restricts their acceptance in clinical environments. Ribeiro et al. [56] emphasized that black box predictions without explanations cannot be trusted in high stakes domains such as healthcare. Similarly, Samek et al. [57] highlighted that clinicians require transparent decision mechanisms to validate automated outcomes against medical knowledge. In the context of NHL classification, many studies prioritize performance metrics while neglecting interpretability, creating a significant gap between algorithmic success and clinical usability.

- Insufficient Integration of Explainable AI Techniques :

Although Grad CAM and SHAP have been individually proposed as explanation tools, few studies integrate both approaches within a unified lymphoma classification framework. Selvaraju et al. [58] demonstrated that Grad CAM provides visual localization of important regions, while Lundberg and Lee [59] showed that SHAP offers feature level contribution analysis. However, most existing research employs only one explainability technique, limiting comprehensive interpretation. The lack of combined visual and quantitative explanations represents a gap in delivering robust and clinically meaningful insights.

- Focus on Binary Classification Rather Than Multi Class Subtypes :

A significant portion of existing research addresses binary classification problems such as cancer vs. non cancer detection rather than multi class differentiation among lymphoma subtypes. Litjens et al. [60] noted that many medical imaging studies simplify diagnostic tasks to binary decisions for computational convenience. This approach does not reflect real clinical scenarios, where accurate differentiation between MCL, FL, and CLL is essential. Consequently, there exists a gap in developing deep learning models capable of reliable multi class NHL subtype classification.

- Lack of Standardized Evaluation Metrics :

Different studies employ varying evaluation metrics such as accuracy, sensitivity, specificity, and AUC, making cross study comparison difficult. Esteva et al. [61] argued that inconsistent evaluation standards hinder reproducibility and objective benchmarking in healthcare AI research. Without standardized metrics, it becomes challenging to assess

the true effectiveness of NHL classification models. This inconsistency constitutes a methodological gap that affects the reliability and generalization of reported results.

- Limited Use of Feature Level Explanations :

Most current explainability approaches in medical imaging rely on heatmaps or attention maps without quantifying the contribution of individual features. Lundberg et al. [59] demonstrated that SHAP values provide mathematically grounded feature attributions that enhance interpretability. However, existing lymphoma classification systems rarely incorporate such feature level explanations, thereby restricting the depth of model understanding. This gap limits clinicians' ability to assess which specific attributes influence classification outcomes.

- Dependence on Small or Homogeneous Datasets :

Many deep learning models for lymphoma classification are trained on small and homogeneous datasets, which reduces their ability to generalize across diverse populations. LeCun et al. [62] emphasized that deep learning systems require large and diverse datasets to achieve robust performance. The limited availability of annotated medical datasets further exacerbates this issue, creating a gap between experimental success and real world applicability.

- Minimal Clinical Validation of Explanations :

Although explainability techniques generate visual and numerical interpretations, few studies validate these explanations against expert medical opinion. Holzinger et al. [63] stressed that explainable AI in medicine must be evaluated not only technically but also clinically. Without validation by pathologists or clinicians, explanations may not align with established diagnostic criteria. This lack of clinical interpretability assessment forms a critical gap in current research.

- High Computational Complexity of Existing Models :

Deep neural networks often demand substantial computational resources for training and inference, which limits their deployment in clinical settings with constrained infrastructure. Goodfellow et al. [64] reported that deep architectures require extensive processing power and memory. As a result, many existing lymphoma classification models remain confined to research environments, revealing a gap in designing efficient and scalable systems suitable for healthcare institutions.

- Scarcity of End to End Explainable Diagnostic Systems :

Current solutions frequently treat classification and explanation as separate processes rather than as an integrated pipeline. Doshi Velez and Kim [65] argued for interpretable machine learning systems that embed explanation mechanisms directly into decision making workflows. The absence of end to end explainable diagnostic frameworks for NHL classification represents a gap in system design and practical deployment.

- Limited Emphasis on Trustworthy AI in Lymphoma Diagnosis :

Trust, accountability, and transparency are essential requirements for medical AI systems. The European Commission's guidelines on trustworthy AI [66] emphasize the need for ethical and transparent models in healthcare. However, existing lymphoma classification studies rarely address these aspects systematically. This lack of focus on trustworthy AI principles constitutes a gap that hinders long term adoption of intelligent diagnostic tools.

2.3 OBJECTIVES

- i. To develop a deep neural network model for multi-class classification of NHL subtypes (MCL, FL, and CLL). : This objective focuses on designing and training a robust deep learning model capable of accurately distinguishing between the three selected Non-Hodgkin Lymphoma subtypes. The model aims to learn discriminative patterns from the input data and provide reliable predictions that can assist in automated diagnostic support.
- ii. To preprocess and standardize the dataset for effective model training and evaluation.: This objective involves applying data preprocessing techniques such as normalization, noise reduction, and augmentation to improve data quality and consistency. Proper preprocessing ensures that the model generalizes well and reduces bias caused by variations in input data.
- iii. To evaluate the performance of the classification model using standard metrics: This objective aims to assess the effectiveness of the proposed model using quantitative measures such as accuracy, precision, recall, F1-score, and confusion matrix analysis. These metrics provide a comprehensive evaluation of classification reliability across different lymphoma subtypes.
- iv. To integrate Grad-CAM for visual interpretation of model predictions. : This objective seeks to implement Gradient-weighted Class Activation Mapping to highlight the regions of the input data that contribute most significantly to

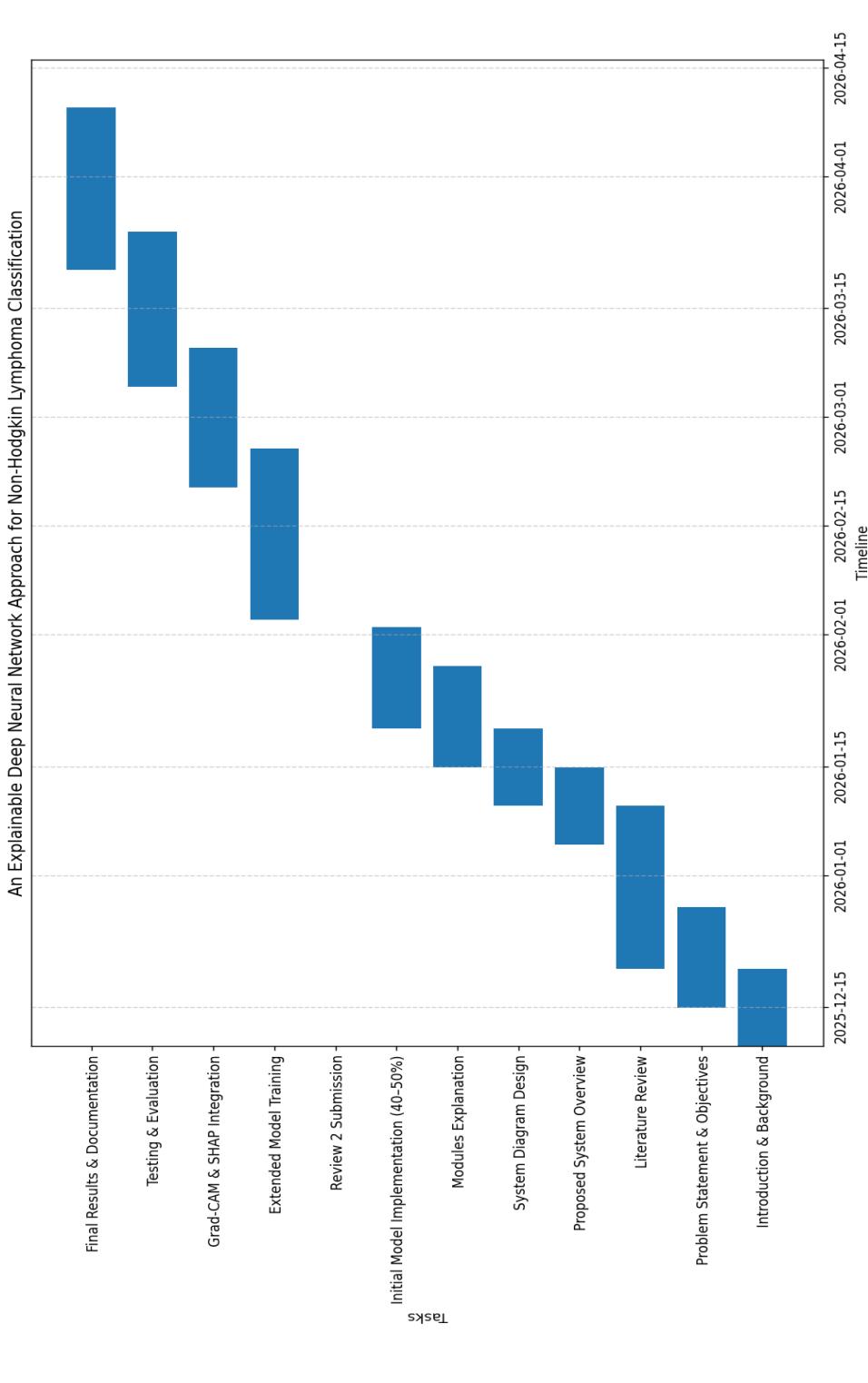
- classification decisions. Visual explanations enable validation of whether the model focuses on clinically relevant features.
- v. To incorporate SHAP for feature-level explanation of classification outcomes.: This objective emphasizes the use of SHAP values to quantify the contribution of individual features toward each prediction. Feature-level explanations enhance transparency and allow deeper understanding of model behavior beyond visual attention maps.
 - vi. To design a two-stage framework combining classification and explainability. : This objective aims to establish a structured pipeline in which the first stage performs accurate NHL subtype classification and the second stage provides interpretable explanations. The framework ensures both predictive performance and transparency within a unified system.
 - vii. To compare the results of different explainability techniques.: This objective involves analyzing and comparing Grad-CAM and SHAP outputs to assess their effectiveness in explaining model decisions. Such comparison provides insights into the strengths and limitations of each method for clinical interpretation.
 - viii. To enhance model reliability and trustworthiness for clinical decision support.: This objective focuses on improving user confidence by providing interpretable results that can be examined by medical professionals. Transparent predictions support responsible use of artificial intelligence in healthcare environments.
 - ix. To identify limitations and challenges in explainable deep learning for lymphoma classification. : This objective aims to analyze potential constraints such as data quality, computational complexity, and interpretability issues. Understanding these challenges provides direction for future improvements and research extensions.
 - x. To demonstrate the applicability of explainable AI in medical diagnostic systems. : This objective seeks to show that combining deep neural networks with explainability techniques can serve as an effective approach for medical diagnosis. The study aims to contribute to the advancement of explainable artificial intelligence in oncology and computational pathology.

2.4 PROBLEM STATEMENT

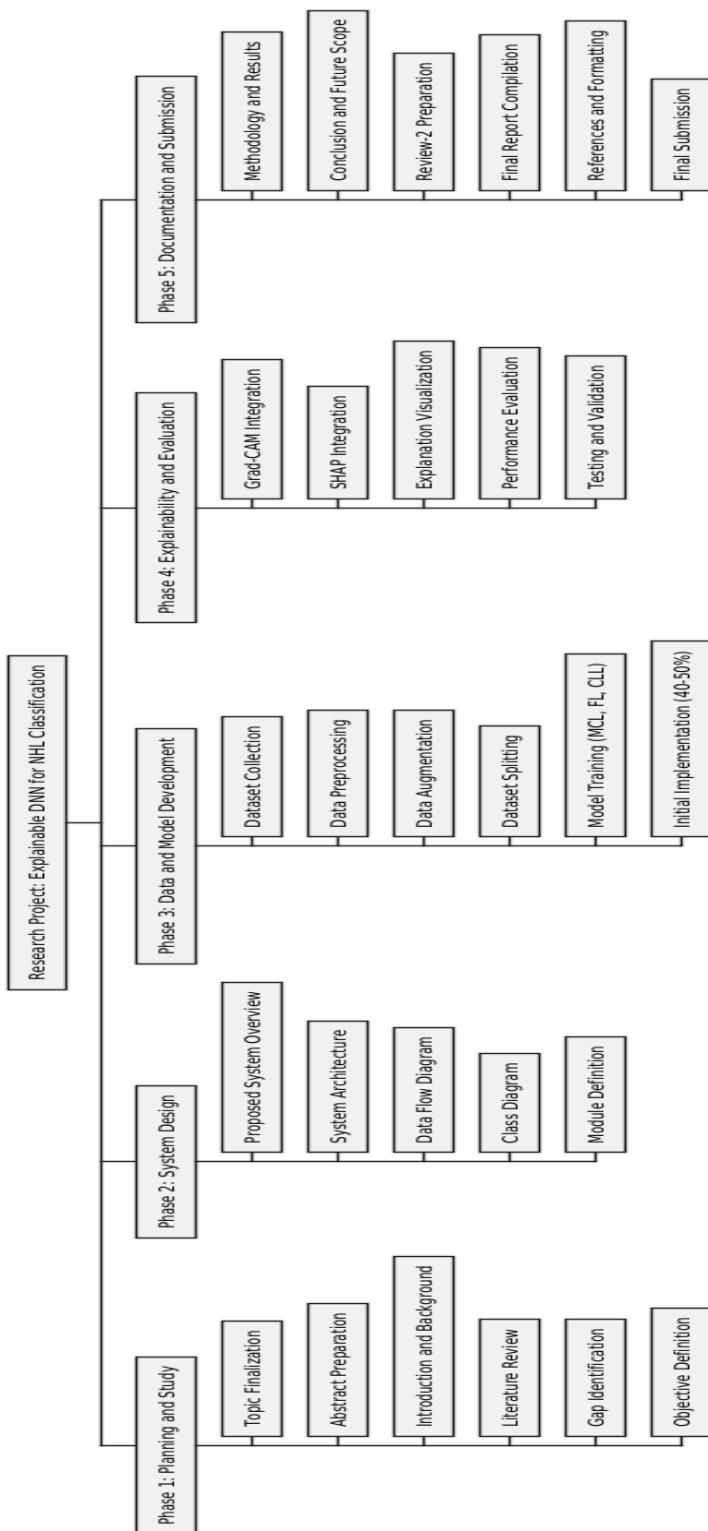
Accurate classification of Non-Hodgkin Lymphoma subtypes remains challenging due to the complexity of medical data and the lack of interpretability in existing deep learning models. This project aims to develop an explainable deep neural network framework to reliably classify MCL, FL, and CLL while providing transparent and clinically meaningful predictions.

2.5 PROJECT PLAN

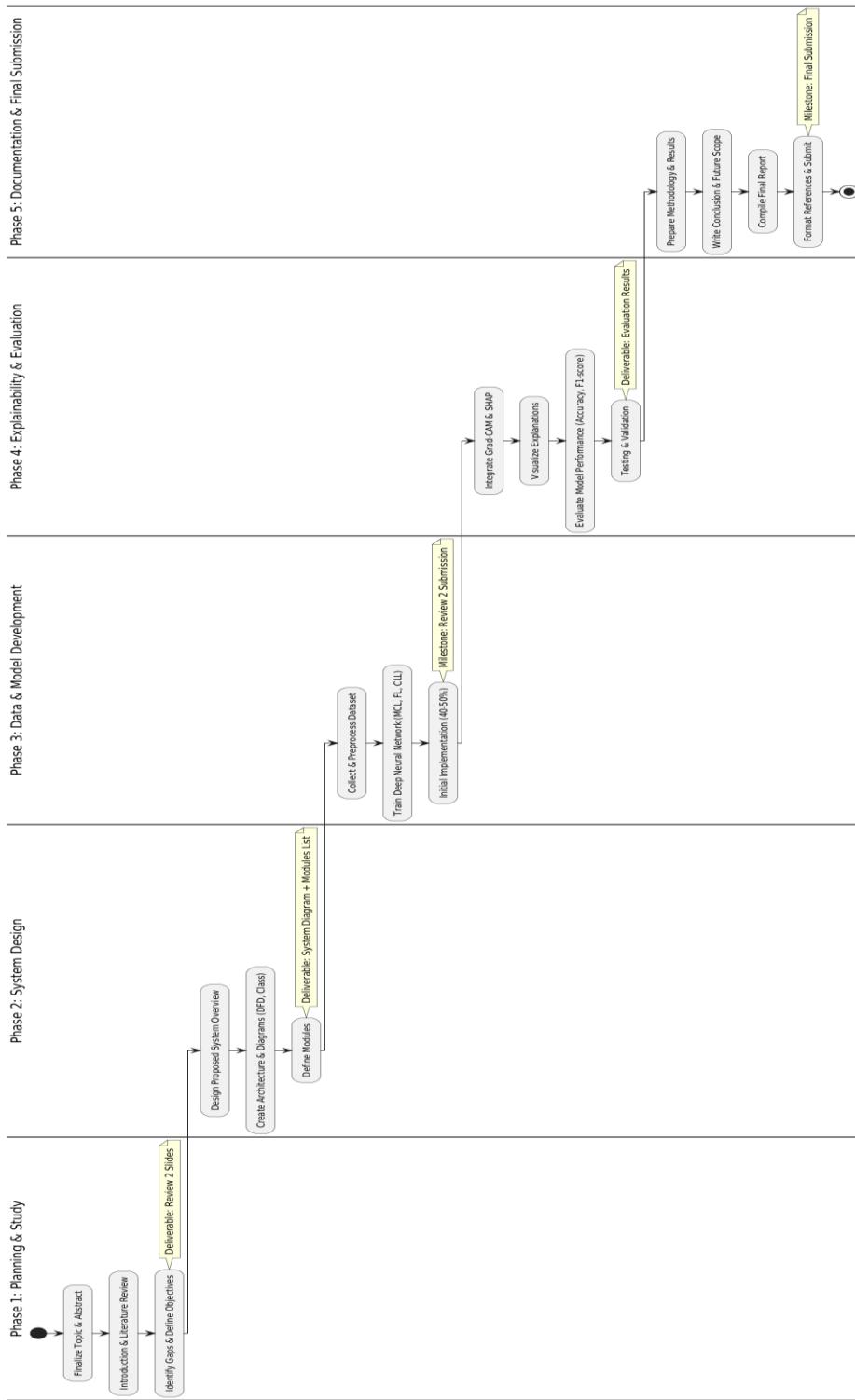
2.5.1 GANTT CHART



2.5.2 WORK BREAKDOWN STRUCTURE



2.5.3 ACTIVITY CHART



CHAPTER 3

TECHNICAL SPECIFICATION

3.1 REQUIREMENTS

3.1.1 FUNCTIONAL

- Accept histopathological image datasets organized into training, validation, and testing splits
- Support high-resolution RGB image inputs for feature extraction
- Perform image resizing, normalization, and label encoding during preprocessing
- Apply data augmentation techniques to improve generalization and reduce overfitting
- Extract discriminative visual features using a Hybrid Vision Transformer (H-ViT) combining CNN and Transformer architectures
- Train deep learning models using supervised learning approaches
- Evaluate model performance using accuracy, precision, recall, F1-score, and ROC-AUC metrics
- Save trained models and best-performing checkpoints for reuse without retraining
- Load saved models for inference and further analysis
- Generate performance visualizations such as accuracy, loss, and ROC curves
- Provide feature-space visualizations (e.g., t-SNE) for interpretability
- Output predicted lymphoma subtype labels along with probability scores

3.1.2 NON-FUNCTIONAL

- Ensure reliable classification accuracy within acceptable training time constraints
- Provide efficient inference on unseen images using saved models
- Support scalability to accommodate larger datasets and additional lymphoma subtypes
- Maintain robustness through model checkpointing to handle runtime interruptions
- Ensure stable execution across multiple training and evaluation sessions
- Use a modular and flexible code structure to enable easy experimentation
- Maintain clear separation between preprocessing, training, evaluation, and visualization components

- Provide well-documented and maintainable modules for future enhancements and research extensions

3.2 FEASIBILITY STUDY

3.2.1 TECHNICAL FEASIBILITY

- Availability of powerful deep learning frameworks such as TensorFlow and Keras supports efficient model development
- Use of pretrained CNN backbones like MobileNetV2 reduces training time and computational complexity
- Adoption of hybrid architectures combining CNN, Transformer, and GNN enables effective local and global feature learning
- Execution on Google Colab with GPU support allows training of large-scale models without the need for high-end local hardware
- Compatibility with existing software tools and libraries ensures smooth system implementation
- The proposed system can be effectively developed using current technologies and infrastructure

3.2.2 ECONOMIC FEASIBILITY

- Utilizes open-source libraries such as TensorFlow, Keras, NumPy, and Scikit-Learn
- Google Colab provides free or low-cost access to GPU resources
- Does not require proprietary software, licensed datasets, or paid platforms
- Minimal hardware investment is needed, making the project cost-efficient
- Suitable for academic, research, and educational use with limited financial resource

3.2.3 SOCIAL FEASIBILITY

- Assists in early and accurate diagnosis of Non-Hodgkin Lymphoma
- Reduces manual workload on pathologists by offering AI-based decision support
- Encourages adoption of artificial intelligence in healthcare for improved diagnostic consistency
- Enhances patient outcomes through faster and more reliable disease classification

- Acts as a supportive diagnostic tool and does not replace medical professionals

3.3 SYSTEM SPECIFICATION

3.3.1 HARDWARE SPECIFICATION

Component	Specification
Processor	Intel i5 / AMD Ryzen 5 or higher
RAM	Minimum 8 GB (Recommended 16 GB)
GPU	NVIDIA GPU (T4 / RTX preferred)
Storage	Minimum 50 GB free disk space
Platform	Google Colab / Local Machine

3.3.2 SOFTWARE SPECIFICATION

Software	Specification
Operating System	Windows / Linux
Programming Language	Python 3.10 or higher
Deep Learning Framework	TensorFlow, Keras
Data Processing Libraries	NumPy, Pandas
Visualization Libraries	Matplotlib, Seaborn
Machine Learning Tools	Scikit-Learn
Development Environment	Jupyter Notebook / Google Colab

CHAPTER 4

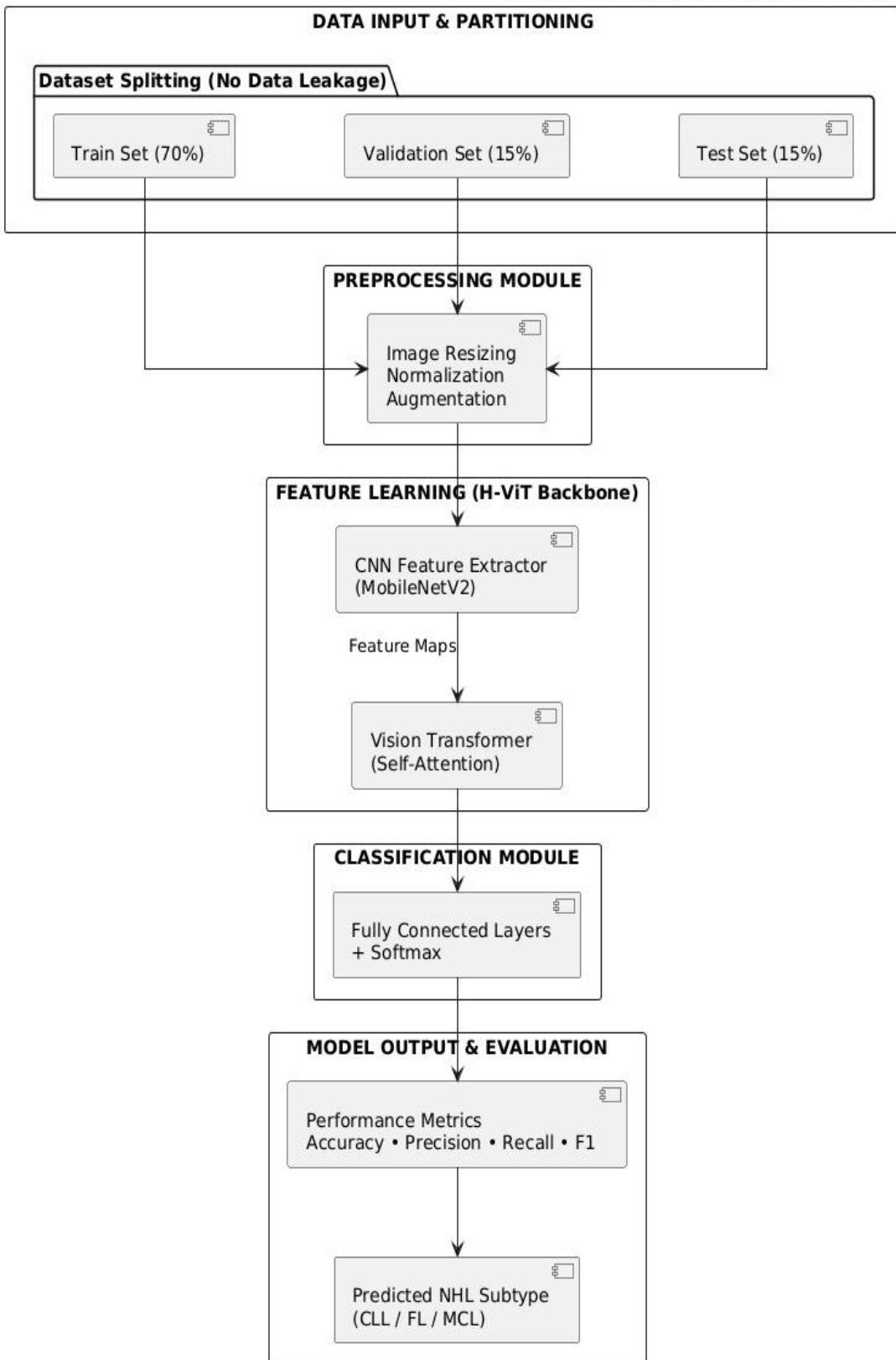
DESIGN APPROACH AND DETAILS

4.1 SYSTEM ARCHITECTURE

The proposed system architecture is designed to perform reliable and structured classification of Non-Hodgkin Lymphoma (NHL) subtypes using a hybrid deep learning approach. The workflow begins with the input histopathology image dataset, which is systematically partitioned into training (70%), validation (15%), and test (15%) sets to ensure unbiased learning and prevent data leakage. This strict dataset separation enables accurate model evaluation on unseen data and improves the generalization capability of the system.

Following dataset partitioning, the images undergo a preprocessing stage that includes resizing, normalization, and data augmentation. These operations standardize the input images and enhance robustness against variations in staining, scale, and orientation commonly observed in histopathological data. The preprocessed images are then passed to the feature learning stage, which employs a Hybrid Vision Transformer (H-ViT) backbone. In this stage, a CNN-based feature extractor (MobileNetV2) captures local morphological patterns, while a Vision Transformer module models global contextual relationships using self-attention mechanisms.

The extracted features are forwarded to the classification module, consisting of fully connected layers and a softmax classifier, which predicts the NHL subtype as Mantle Cell Lymphoma (MCL), Follicular Lymphoma (FL), or Chronic Lymphocytic Leukaemia (CLL). The final stage evaluates model performance using standard classification metrics such as accuracy, precision, recall, and F1-score. The architecture emphasizes modularity, scalability, and consistency with clinical diagnostic workflows, making it suitable for use as a decision-support system in automated lymphoma subtype classification.



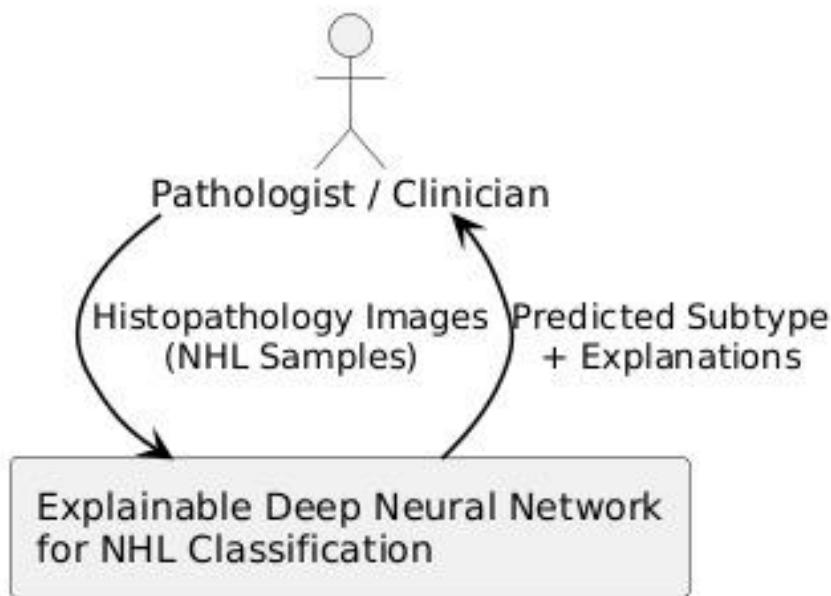
4.2 DESIGN

4.2.1 DATA FLOW DIAGRAM

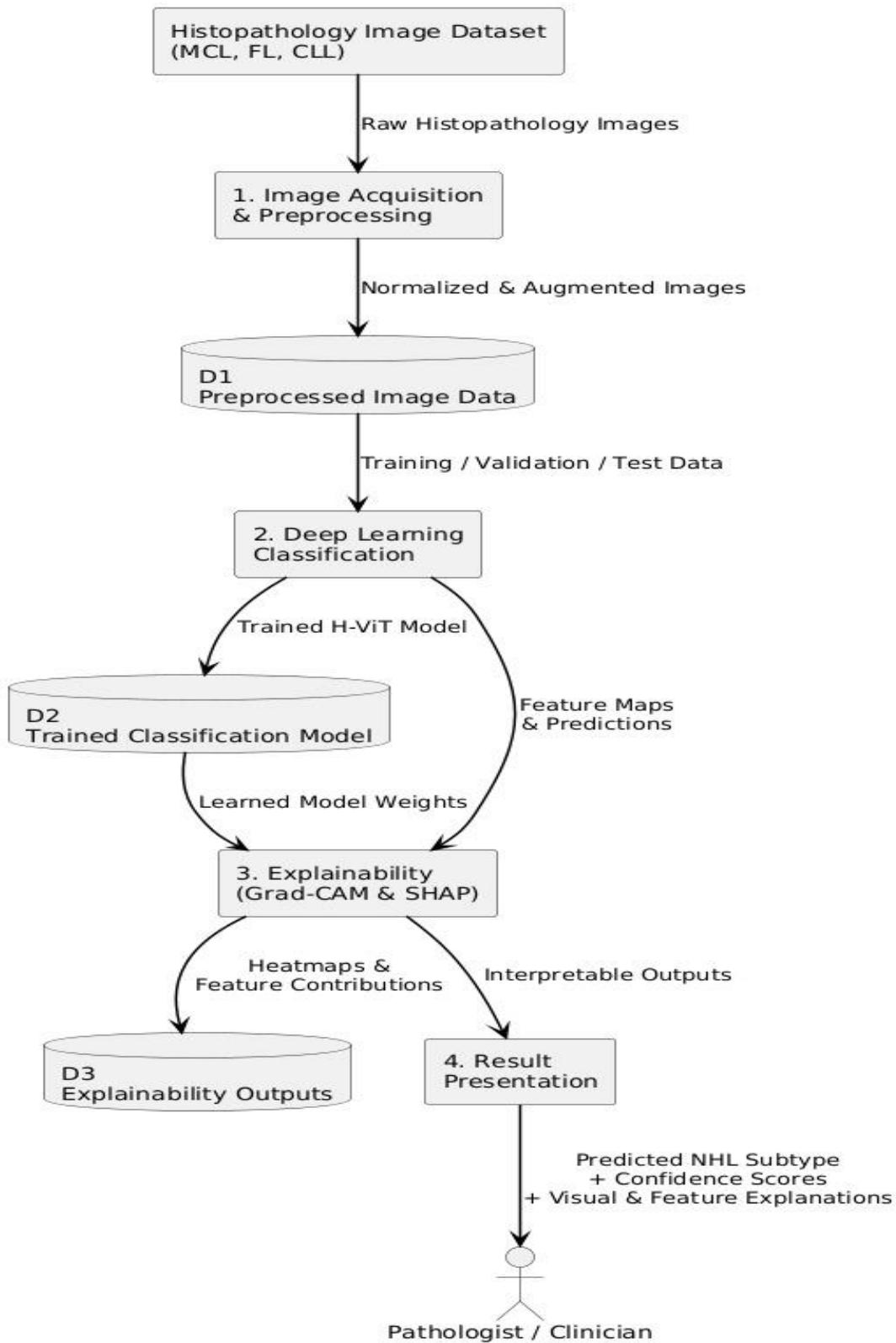
This DFD Level 1 shows the detailed data flow process of the proposed Non-Hodgkin Lymphoma classification system, which involves the processing, analysis, and presentation of histopathology images to the end user. The data flow starts with the histopathology image dataset as the external data source, which is inputted into the image acquisition and preprocessing process where resizing, normalization, and augmentation are done and stored as preprocessed image data. The preprocessed data is then inputted into the deep learning classification process, which uses a trained hybrid CNN-Vision Transformer model to produce subtype predictions. The trained model and feature representations are then inputted into the explainability process, which uses Grad-CAM and SHAP to produce visual and feature level explanations of the trained model's predictions, stored as explainability outputs. Finally, the result presentation process takes the classification outputs and explanations and provides interpretable diagnostic outputs to the clinician.

x

DFD Level 0 - Explainable NHL Classification System

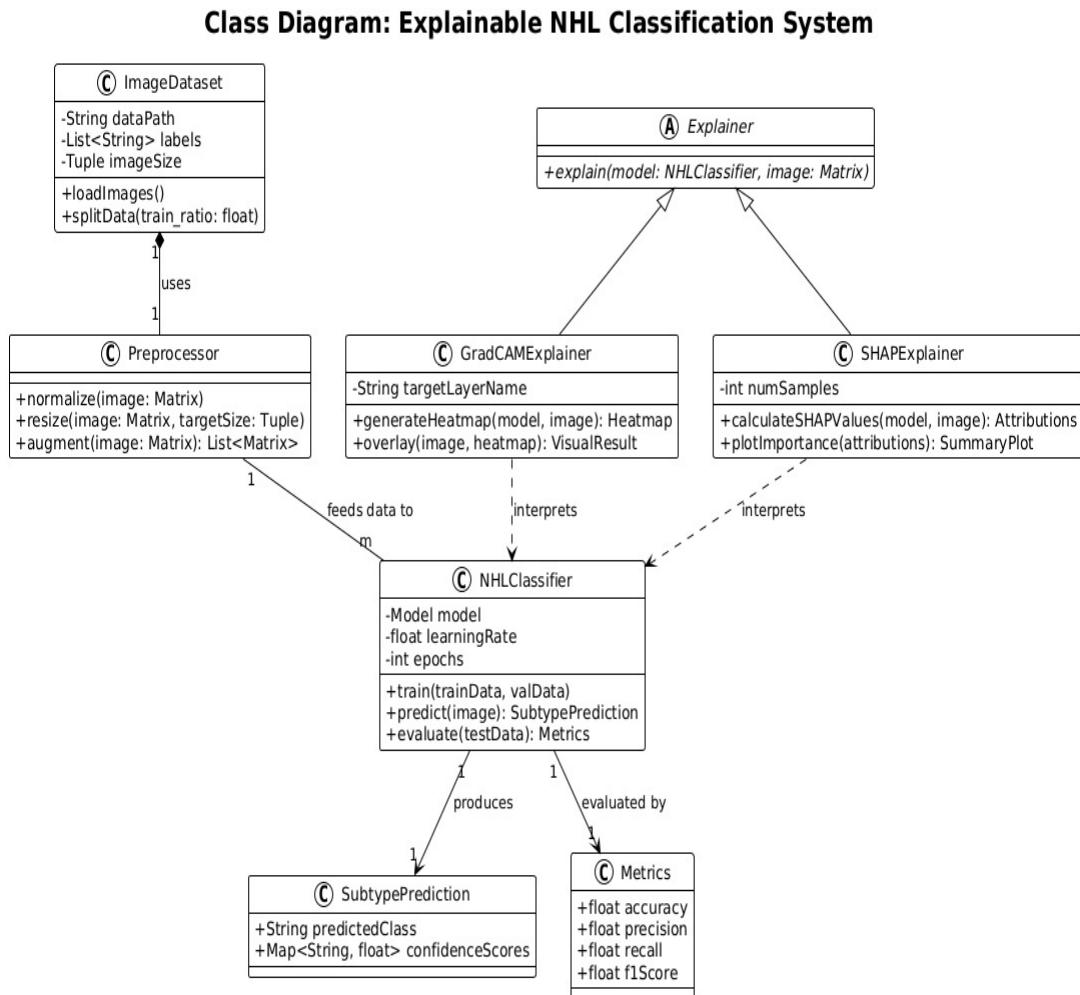


DFD Level 1 - Explainable Deep Learning Framework for Non-Hodgkin Lymphoma Classification



4.2.2 CLASS DIAGRAM

The class diagram above illustrates the structural design of the proposed explainable Non-Hodgkin Lymphoma classification system by showing the key software components and their relationships. The diagram above illustrates the workflow from the ImageDataset class, which is responsible for the image loading and dataset splitting, followed by the Preprocessor class, which is responsible for the histopathology image resizing, normalization, and augmentation. The key classification functionality is implemented in the NHLClassifier class, which implements the training, prediction, and evaluation of the classification model, resulting in the SubtypePrediction and Metrics objects, which store the classification results and metrics. To improve the interpretability of the classification results, the diagram above proposes the abstract Explainer class with its concrete implementations GradCAMExplainer and SHAPExplainer classes, which produce visual and feature level explanations of the classification results.



CHAPTER 5

CODE AND RESULTS

5.1 CODE

```
!pip install -U tensorflow==2.19.0

!pip install -q scikit-learn matplotlib seaborn opencv-python-headless
Pillow scipy networkx

from tensorflow.keras import mixed_precision
mixed_precision.set_global_policy("mixed_float16")
print("✓ Mixed precision enabled")

import os
import shutil
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from pathlib import Path
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers, models
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau,
ModelCheckpoint
from sklearn.metrics import confusion_matrix, classification_report,
roc_auc_score, roc_curve
from sklearn.metrics import precision_recall_fscore_support
from sklearn.preprocessing import label_binarize
from sklearn.utils.class_weight import compute_class_weight
import cv2
from google.colab import drive
import networkx as nx
from scipy.spatial.distance import cdist
from keras.saving import register_keras_serializable

np.random.seed(42)
tf.random.set_seed(42)

gpus = tf.config.list_physical_devices('GPU')
if gpus:
```

```

try:
    for gpu in gpus:
        tf.config.experimental.set_memory_growth(gpu, True)
    print(f"✓ GPU available: {len(gpus)} device(s)")
except RuntimeError as e:
    print(e)
else:
    print("⚠️ No GPU detected. Training will be slower.")

print(f"TensorFlow version: {tf.__version__}")

drive.mount('/content/drive')

BASE_PATH = '/content/drive/MyDrive/NHL_DATA/Multi_Cancer/Multi_Cancer/Lymphoma'

if os.path.exists(BASE_PATH):
    print(f"✓ Data directory found: {BASE_PATH}")
    subdirs = [d for d in os.listdir(BASE_PATH) if os.path.isdir(os.path.join(BASE_PATH, d))]
    print(f" Subdirectories: {subdirs}")
else:
    print(f"✗ Data directory not found: {BASE_PATH}")
    print(" Please update BASE_PATH to match your data location")

MODEL_DIR = "/content/drive/MyDrive/NHL_Project/models"
!mkdir -p $MODEL_DIR

CONFIG = {
    'IMG_SIZE': (128, 128),
    'BATCH_SIZE': 32,
    'EPOCHS': 10,
    'LEARNING_RATE': 1e-4,
    'TRAIN_SPLIT': 0.70,
    'VAL_SPLIT': 0.15,
    'TEST_SPLIT': 0.15,
    'NUM_CLASSES': 3,
    'CLASS_NAMES': ['CLL', 'FL', 'MCL'],
    'CLASS_LABELS': {
        'lymph_cll': 'Chronic Lymphocytic Leukemia (CLL)',
        'lymph_fl': 'Follicular Lymphoma (FL)',
        'lymph_mcl': 'Mantle Cell Lymphoma (MCL)'
    },
    'RANDOM_SEED': 42
}

```

```
print("Configuration:")
for key, value in CONFIG.items():
    print(f"  {key}: {value}")

def create_train_val_test_split(source_dir, dest_dir, train_ratio=0.70,
val_ratio=0.15, test_ratio=0.15, seed=42):
    np.random.seed(seed)

    for split in ['train', 'val', 'test']:
        os.makedirs(os.path.join(dest_dir, split), exist_ok=True)

    class_dirs = [d for d in os.listdir(source_dir)
                 if os.path.isdir(os.path.join(source_dir, d))]

    split_info = {}

    for class_dir in class_dirs:
        class_path = os.path.join(source_dir, class_dir)
        all_files = [f for f in os.listdir(class_path)
                     if f.lower().endswith('.jpg', '.jpeg', '.png',
                     '.bmp'))]

        np.random.shuffle(all_files)

        n_total = len(all_files)
        n_train = int(n_total * train_ratio)
        n_val = int(n_total * val_ratio)

        train_files = all_files[:n_train]
        val_files = all_files[n_train:n_train + n_val]
        test_files = all_files[n_train + n_val:]

        for split in ['train', 'val', 'test']:
            os.makedirs(os.path.join(dest_dir, split, class_dir),
exist_ok=True)

            for fname in train_files:
                src = os.path.join(class_path, fname)
                dst = os.path.join(dest_dir, 'train', class_dir, fname)
                shutil.copy2(src, dst)

            for fname in val_files:
                src = os.path.join(class_path, fname)
                dst = os.path.join(dest_dir, 'val', class_dir, fname)
                shutil.copy2(src, dst)
```

```

        for fname in test_files:
            src = os.path.join(class_path, fname)
            dst = os.path.join(dest_dir, 'test', class_dir, fname)
            shutil.copy2(src, dst)

        split_info[class_dir] = {
            'total': n_total,
            'train': len(train_files),
            'val': len(val_files),
            'test': len(test_files)
        }

    return split_info

SPLIT_DIR = '/content/drive/MyDrive/NHL_Project/data/lymphoma_split'

print("Creating train/val/test split...")
print("This ensures NO data leakage between sets.\n")

split_info = create_train_val_test_split(
    source_dir=BASE_PATH,
    dest_dir=SPLIT_DIR,
    train_ratio=CONFIG['TRAIN_SPLIT'],
    val_ratio=CONFIG['VAL_SPLIT'],
    test_ratio=CONFIG['TEST_SPLIT'],
    seed=CONFIG['RANDOM_SEED']
)

print("\n" + "="*70)
print("DATA SPLIT STATISTICS (NO LEAKAGE)")
print("="*70)
for class_name, counts in split_info.items():
    print(f"\n{CONFIG['CLASS_LABELS'].get(class_name, class_name)}:")
    print(f"  Total: {counts['total']} images")
    print(f"  Train: {counts['train']}")
    print(f"  ({counts['train']}/{counts['total'])*100:.1f}%)")
    print(f"  Val: {counts['val']}")
    print(f"  ({counts['val']}/{counts['total'])*100:.1f}%)")
    print(f"  Test: {counts['test']}")
    print(f"  ({counts['test']}/{counts['total'])*100:.1f}%)")
    print("="*70)

print("\n✅ Split complete! Train/Val/Test are now completely
independent.")

```

```

print("  Test set is held-out and will ONLY be used for final
evaluation.")

plt.figure(figsize=(8,5))
sns.barplot(x=list(split_info.keys()), y=[split_info[c]['total'] for c in
split_info])
plt.title("Class Distribution (Total Images per Class)", fontsize=14,
fontweight='bold')
plt.ylabel("Number of Images")
plt.xlabel("NHL Subtype")
plt.show()

fig, axes = plt.subplots(1, 3, figsize=(15,5))
for i, class_dir in enumerate(split_info.keys()):
    sample_img_path = os.path.join(BASE_PATH, class_dir,
os.listdir(os.path.join(BASE_PATH, class_dir))[0])
    img = cv2.imread(sample_img_path)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    axes[i].imshow(img)
    axes[i].set_title(CONFIG['CLASS_LABELS'][class_dir])
    axes[i].axis('off')
plt.suptitle("Sample Histopathology Images per NHL Subtype", fontsize=16,
fontweight='bold')
plt.show()

import os
import matplotlib.pyplot as plt
import seaborn as sns

SPLIT_DIR = '/content/drive/MyDrive/NHL_Project/data/lymphoma_split'

class_folders = ['lymph_cll', 'lymph_fl', 'lymph_mcl']

splits = ['train', 'val', 'test']

split_counts = {}
for split in splits:
    split_path = os.path.join(SPLIT_DIR, split)
    counts = {folder: len(os.listdir(os.path.join(split_path, folder))) for
folder in class_folders}
    split_counts[split] = counts

import pandas as pd
df_counts = pd.DataFrame(split_counts).T
df_counts.index.name = 'Split'
df_counts.columns = ['CLL', 'FL', 'MCL']

```

```

print("Image counts per split:")
print(df_counts)

df_counts.plot(kind='bar', figsize=(8,6))
plt.title('Number of Images per Class in Each Split')
plt.ylabel('Number of Images')
plt.xlabel('Data Split')
plt.xticks(rotation=0)
plt.show()

train_counts = df_counts.loc['train']
plt.figure(figsize=(6,6))
plt.pie(train_counts, labels=train_counts.index, autopct='%1.1f%%',
startangle=140, colors=['#ff9999','#66b3ff','#99ff99'])
plt.title('Class Distribution in Training Set')
plt.show()

import random
from PIL import Image
image_sizes = {cls: [] for cls in class_folders}

for cls in class_folders:
    cls_path = os.path.join(SPLIT_DIR, 'train', cls)
    for img_name in os.listdir(cls_path):
        img_path = os.path.join(cls_path, img_name)
        img = Image.open(img_path)
        image_sizes[cls].append(img.size)

sizes_df = pd.DataFrame({cls: [h for w,h in image_sizes[cls]] for cls in
class_folders})
plt.figure(figsize=(8,6))
sns.boxplot(data=sizes_df)
plt.title('Image Height Distribution per Class (Train Set)')
plt.ylabel('Height (pixels)')
plt.show()

for split in splits:
    counts = [len(os.listdir(os.path.join(SPLIT_DIR, split, cls))) for cls
in class_folders]
    plt.figure(figsize=(6,6))
    plt.pie(counts, labels=[cls.upper() for cls in class_folders],
autopct='%1.1f%%', startangle=140)
    plt.title(f'Class Distribution in {split.capitalize()} Set')
    plt.show()

```

```

train_datagen = ImageDataGenerator(
    rescale=1./255,
    rotation_range=10,
    width_shift_range=0.1,
    height_shift_range=0.1,
    zoom_range=0.1,
    horizontal_flip=True,
    fill_mode='nearest'
)

val_test_datagen = ImageDataGenerator(rescale=1./255)

print("✓ Data augmentation configured")
print(" Training: rotation, shift, zoom, horizontal flip")
print(" Val/Test: rescaling only (no augmentation)")

train_generator = train_datagen.flow_from_directory(
    os.path.join(SPLIT_DIR, 'train'),
    target_size=CONFIG['IMG_SIZE'],
    batch_size=CONFIG['BATCH_SIZE'],
    class_mode='categorical',
    shuffle=True,
    seed=CONFIG['RANDOM_SEED']
)

validation_generator = val_test_datagen.flow_from_directory(
    os.path.join(SPLIT_DIR, 'val'),
    target_size=CONFIG['IMG_SIZE'],
    batch_size=CONFIG['BATCH_SIZE'],
    class_mode='categorical',
    shuffle=False,
    seed=CONFIG['RANDOM_SEED']
)

test_generator = val_test_datagen.flow_from_directory(
    os.path.join(SPLIT_DIR, 'test'),
    target_size=CONFIG['IMG_SIZE'],
    batch_size=CONFIG['BATCH_SIZE'],
    class_mode='categorical',
    shuffle=False,
    seed=CONFIG['RANDOM_SEED']
)

class_indices = train_generator.class_indices
index_to_class = {v: k for k, v in class_indices.items()}

```

```

print("\n" + "="*70)
print("DATA GENERATORS (FROM INDEPENDENT DIRECTORIES)")
print("=*70)
print(f"Training samples: {train_generator.samples}")
print(f"Validation samples: {validation_generator.samples}")
print(f"Test samples: {test_generator.samples}")
print(f"\nClass mapping: {class_indices}")
print("=*70)
print("\n All generators use SEPARATE directories - NO DATA LEAKAGE")

class_weights_array = compute_class_weight(
    class_weight='balanced',
    classes=np.unique(train_generator.classes),
    y=train_generator.classes
)

class_weights = dict(enumerate(class_weights_array))
print("Class Weights (to handle imbalance):")
for idx, weight in class_weights.items():
    print(f" {index_to_class[idx]}: {weight:.3f}")

@register_keras_serializable(package="Custom")
class TransformerBlock(layers.Layer):

    def __init__(self, embed_dim, num_heads, ff_dim, dropout_rate=0.1,
                 **kwargs):
        super().__init__(**kwargs)
        self.embed_dim = embed_dim
        self.num_heads = num_heads
        self.ff_dim = ff_dim
        self.dropout_rate = dropout_rate

        self.att = layers.MultiHeadAttention(
            num_heads=num_heads,
            key_dim=embed_dim,
            dropout=dropout_rate
        )
        self.ffn = models.Sequential([
            layers.Dense(ff_dim, activation='gelu'),
            layers.Dropout(dropout_rate),
            layers.Dense(embed_dim)
        ])
        self.layernorm1 = layers.LayerNormalization(epsilon=1e-6)
        self.layernorm2 = layers.LayerNormalization(epsilon=1e-6)

```

```

        self.dropout1 = layers.Dropout(dropout_rate)
        self.dropout2 = layers.Dropout(dropout_rate)

    def call(self, inputs, training=False):
        attn_output = self.att(inputs, inputs)
        attn_output = self.dropout1(attn_output, training=training)
        out1 = self.layernorm1(inputs + attn_output)

        ffn_output = self.ffn(out1)
        ffn_output = self.dropout2(ffn_output, training=training)
        return self.layernorm2(out1 + ffn_output)

    def get_config(self):
        config = super().get_config()
        config.update({
            'embed_dim': self.embed_dim,
            'num_heads': self.num_heads,
            'ff_dim': self.ff_dim,
            'dropout_rate': self.dropout_rate
        })
        return config

def create_hvit_nhl_model(input_shape=(128, 128, 3), num_classes=3,
                           transformer_blocks=1, embed_dim=128,
                           num_heads=2):
    inputs = layers.Input(shape=input_shape, name='input_image')

    base_model = MobileNetV2(
        include_top=False,
        weights='imagenet',
        input_tensor=inputs,
        pooling=None
    )

    for layer in base_model.layers:
        layer.trainable = False

    x = base_model.output

    H, W, C = x.shape[1], x.shape[2], x.shape[3]
    x = layers.Reshape((H * W, C))(x)
    x = layers.Dense(embed_dim)(x)

    positions = tf.range(H * W)
    position_embedding = layers.Embedding(

```

```

        input_dim=H * W,
        output_dim=embed_dim
    )(positions)
    x = x + position_embedding

    for i in range(transformer_blocks):
        x = TransformerBlock(
            embed_dim=embed_dim,
            num_heads=num_heads,
            ff_dim=embed_dim * 2
        )(x)

    x = layers.GlobalAveragePooling1D()(x)
    x = layers.Dense(128, activation='relu')(x)
    x = layers.Dropout(0.3)(x)
    x = layers.Dense(64, activation='relu')(x)
    x = layers.Dropout(0.2)(x)
    outputs = layers.Dense(num_classes, activation='softmax',
                           dtype='float32')(x)

    model = models.Model(inputs=inputs, outputs=outputs, name='H-ViT-NHL')
    return model

baseline_model = create_hvit_nhl_model(
    input_shape=(*CONFIG['IMG_SIZE'], 3),
    num_classes=CONFIG['NUM_CLASSES'],
    transformer_blocks=1,
    embed_dim=128,
    num_heads=2
)

print("\n" + "*70)
print("BASELINE MODEL: Hybrid CNN + Vision Transformer (H-ViT-NHL)")
print("*70)
baseline_model.summary()
print("*70)

trainable_params = np.sum([tf.keras.backend.count_params(w)
                          for w in baseline_model.trainable_weights])
non_trainable_params = np.sum([tf.keras.backend.count_params(w)
                               for w in
baseline_model.non_trainable_weights])
total_params = trainable_params + non_trainable_params

print("Model Parameters:")

```

```

print(f" Total: {total_params:,}")
print(f" Trainable: {trainable_params:,}")
print(f" Non-trainable: {non_trainable_params:,}")
print(f" Trainable ratio: {trainable_params/total_params*100:.1f}%")

baseline_model.compile(
    optimizer=keras.optimizers.Adam(learning_rate=CONFIG['LEARNING_RATE']),
    loss='categorical_crossentropy',
    metrics=[
        'accuracy',
        keras.metrics.AUC(name='auc', multi_label=False)
    ]
)

print("✓ Baseline model compiled")
print(f" Optimizer: Adam (lr={CONFIG['LEARNING_RATE']} )")
print(f" Loss: Categorical Cross-Entropy")
print(f" Metrics: Accuracy, AUC")

callbacks = [
    EarlyStopping(
        monitor='val_loss',
        patience=10,
        restore_best_weights=True,
        verbose=1
    ),
    ReduceLROnPlateau(
        monitor='val_loss',
        factor=0.5,
        patience=5,
        min_lr=1e-7,
        verbose=1
    ),
    ModelCheckpoint(
        filepath=f"{MODEL_DIR}/best_hvit_baseline.keras",
        monitor='val_accuracy',
        save_best_only=True,
        verbose=1
    )
]

print("✓ Training callbacks configured")

history = baseline_model.fit(
    train_generator,

```

```

    epochs=CONFIG['EPOCHS'],
    validation_data=validation_generator,
    callbacks=callbacks,
    class_weight=class_weights
)

import os
os.listdir("/content/drive/MyDrive/NHL_Project/models")

def plot_training_history(history, title="Training History"):
    fig, axes = plt.subplots(1, 3, figsize=(18, 5))

    axes[0].plot(history.history['loss'], label='Training', linewidth=2)
    axes[0].plot(history.history['val_loss'], label='Validation',
    linewidth=2)
    axes[0].set_xlabel('Epoch', fontsize=12, fontweight='bold')
    axes[0].set_ylabel('Loss', fontsize=12, fontweight='bold')
    axes[0].set_title('Loss', fontsize=14, fontweight='bold')
    axes[0].legend()
    axes[0].grid(alpha=0.3)

    axes[1].plot(history.history['accuracy'], label='Training',
    linewidth=2)
    axes[1].plot(history.history['val_accuracy'], label='Validation',
    linewidth=2)
    axes[1].set_xlabel('Epoch', fontsize=12, fontweight='bold')
    axes[1].set_ylabel('Accuracy', fontsize=12, fontweight='bold')
    axes[1].set_title('Accuracy', fontsize=14, fontweight='bold')
    axes[1].legend()
    axes[1].grid(alpha=0.3)

    axes[2].plot(history.history['auc'], label='Training', linewidth=2)
    axes[2].plot(history.history['val_auc'], label='Validation',
    linewidth=2)
    axes[2].set_xlabel('Epoch', fontsize=12, fontweight='bold')
    axes[2].set_ylabel('AUC', fontsize=12, fontweight='bold')
    axes[2].set_title('AUC', fontsize=14, fontweight='bold')
    axes[2].legend()
    axes[2].grid(alpha=0.3)

    plt.suptitle(title, fontsize=16, fontweight='bold', y=1.02)
    plt.tight_layout()
    plt.show()

plot_training_history(history, "Baseline H-ViT-NHL Training")

```

```

from tensorflow import keras

baseline_model = keras.models.load_model(
    f"{MODEL_DIR}/best_hvit_baseline.keras",
    compile=False
)

from tensorflow.keras.layers import GlobalAveragePooling1D

gap_layer = [layer for layer in baseline_model.layers if isinstance(layer,
GlobalAveragePooling1D)][0]

feature_extractor = keras.Model(
    inputs=baseline_model.input,
    outputs=gap_layer.output
)

import numpy as np
dummy_input = np.random.rand(1, 128, 128, 3).astype(np.float32)
features = feature_extractor({'input_image': dummy_input})
print("Feature shape:", features.shape)

print("Evaluating baseline on TRULY HELD-OUT test set...\n")

test_steps = test_generator.samples // CONFIG['BATCH_SIZE']
y_pred_probs_baseline = baseline_model.predict(test_generator,
steps=test_steps, verbose=1)
y_pred_classes_baseline = np.argmax(y_pred_probs_baseline, axis=1)
y_true_classes = test_generator.classes[:len(y_pred_classes_baseline)]

baseline_accuracy = np.mean(y_pred_classes_baseline == y_true_classes)

print(f"\n\ufe0f Baseline Test Accuracy: {baseline_accuracy:.4f}\n({{baseline_accuracy*100:.2f}}%)")
print(f"    Evaluated on {len(y_pred_classes_baseline)} truly held-out test samples")

cm_baseline = confusion_matrix(y_true_classes, y_pred_classes_baseline)
cm_baseline_norm = confusion_matrix(y_true_classes,
y_pred_classes_baseline, normalize='true')

fig, axes = plt.subplots(1, 2, figsize=(16, 6))

sns.heatmap(cm_baseline, annot=True, fmt='d', cmap='Blues',
            xticklabels=CONFIG['CLASS_NAMES'],

```

```

        yticklabels=CONFIG['CLASS_NAMES'],
        ax=axes[0])
axes[0].set_xlabel('Predicted', fontsize=12, fontweight='bold')
axes[0].set_ylabel('True', fontsize=12, fontweight='bold')
axes[0].set_title('Confusion Matrix (Counts)', fontsize=14,
fontweight='bold')

sns.heatmap(cm_baseline_norm, annot=True, fmt='.%2', cmap='Blues',
            xticklabels=CONFIG['CLASS_NAMES'],
            yticklabels=CONFIG['CLASS_NAMES'],
            ax=axes[1])
axes[1].set_xlabel('Predicted', fontsize=12, fontweight='bold')
axes[1].set_ylabel('True', fontsize=12, fontweight='bold')
axes[1].set_title('Confusion Matrix (Normalized)', fontsize=14,
fontweight='bold')

plt.suptitle('Baseline H-ViT-NHL Performance (Held-Out Test Set)',
             fontsize=16, fontweight='bold', y=1.02)
plt.tight_layout()
plt.show()

sns.heatmap(cm_baseline_norm, annot=True, fmt='.%2', cmap='Blues',
            xticklabels=CONFIG['CLASS_NAMES'],
            yticklabels=CONFIG['CLASS_NAMES'])
plt.title("Baseline H-ViT Confusion Matrix (Normalized)")
plt.show()

plt.figure(figsize=(8,6))
for i, class_name in enumerate(CONFIG['CLASS_NAMES']):
    fpr, tpr, _ = roc_curve(y_true_bin[:, i], y_pred_probs_baseline[:, i])
    plt.plot(fpr, tpr, label=f'{class_name} (AUC = {baseline_roc_auc[i]:.2f})')
    plt.plot([0,1],[0,1],'k--')
    plt.xlabel("False Positive Rate")
    plt.ylabel("True Positive Rate")
    plt.title("ROC Curves - Baseline H-ViT")
    plt.legend()
    plt.grid(alpha=0.3)
    plt.show()

from sklearn.manifold import TSNE
import matplotlib.pyplot as plt
import numpy as np

X_features = feature_extractor.predict(test_generator, steps=test_steps,
verbose=1)

```

```

y_true = test_generator.classes[:X_features.shape[0]]

n_samples = X_features.shape[0]
perplexity = min(30, max(5, n_samples // 3))

print(f"t-SNE: n_samples={n_samples}, using perplexity={perplexity}")

tsne = TSNE(n_components=2, random_state=42, perplexity=perplexity)
X_2d = tsne.fit_transform(X_features)

plt.figure(figsize=(8,6))
for idx, class_name in enumerate(CONFIG['CLASS_NAMES']):
    mask = (y_true == idx)
    plt.scatter(X_2d[mask, 0], X_2d[mask, 1], label=class_name, alpha=0.7,
s=50)

plt.title("t-SNE Visualization of Test Set Features")
plt.xlabel("t-SNE Component 1")
plt.ylabel("t-SNE Component 2")
plt.legend()
plt.grid(alpha=0.3)
plt.show()

print("\n" + "="*70)
print("BASELINE CLASSIFICATION REPORT (HELD-OUT TEST SET)")
print("="*70)
report_baseline = classification_report(
    y_true_classes,
    y_pred_classes_baseline,
    target_names=CONFIG['CLASS_NAMES'],
    digits=4
)
print(report_baseline)
print("="*70)

baseline_precision, baseline_recall, baseline_f1, _ =
precision_recall_fscore_support(
    y_true_classes, y_pred_classes_baseline, average='macro'
)

y_true_bin = label_binarize(y_true_classes, classes=[0, 1, 2])

baseline_roc_auc = {}
for i in range(CONFIG['NUM_CLASSES']):

```

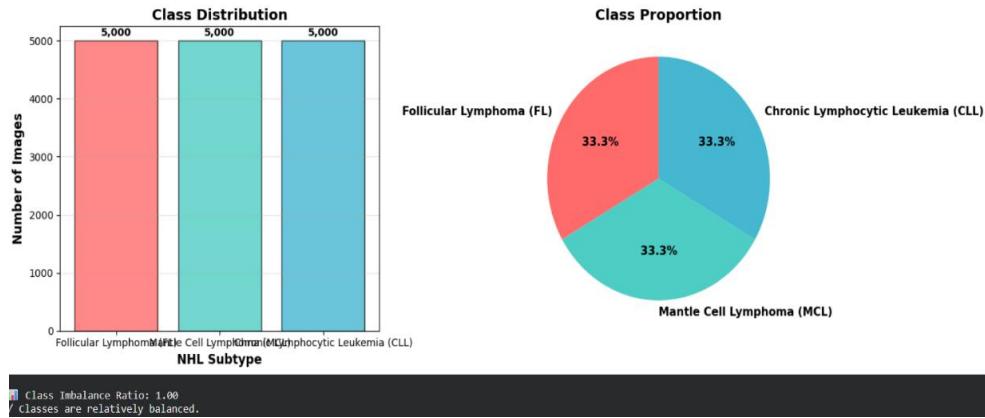
```
    baseline_roc_auc[i] = roc_auc_score(y_true_bin[:, i],
y_pred_probs_baseline[:, i])

baseline_macro_auc = roc_auc_score(y_true_bin, y_pred_probs_baseline,
average='macro')

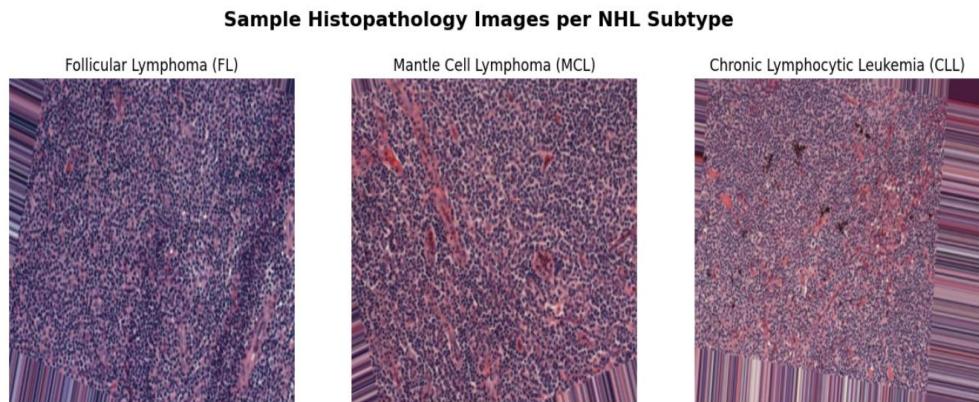
print("\nBaseline ROC-AUC (Held-Out Test):")
for i, class_name in enumerate(CONFIG['CLASS_NAMES']):
    print(f" {class_name}: {baseline_roc_auc[i]:.4f}")
print(f" Macro-Average: {baseline_macro_auc:.4f}")
```

5.2 RESULTS

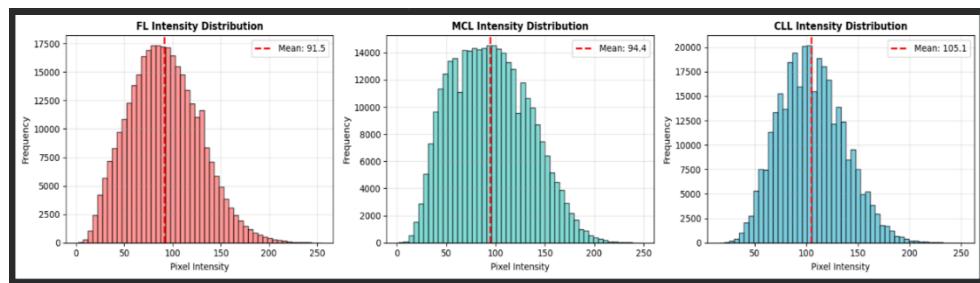
5.2.1 Dataset Overview



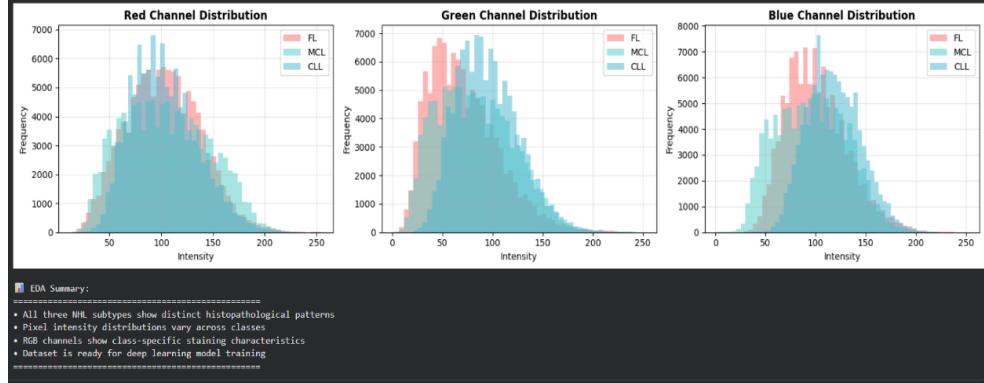
5.2.2 Sample Histopathology images by NHL Subtype



5.2.3 Image Intensity Distribution Analysis



5.2.4 Colour Channel Distribution Analysis



5.2.5 Model Configuration

```
... Configuration:  
    IMG_SIZE: (128, 128)  
    BATCH_SIZE: 32  
    EPOCHS: 10  
    LEARNING RATE: 0.0001  
    TRAIN_SPLIT: 0.7  
    VAL_SPLIT: 0.15  
    TEST_SPLIT: 0.15  
    NUM_CLASSES: 3  
    CLASS NAMES: ['CLL', 'FL', 'MCL']  
    CLASS_LABELS: {'lymph_cll': 'Chronic Lymphocytic Leukemia (CLL)', 'lymph_fl': 'Follicular Lymphoma (FL)', 'lymph_mcl': 'Mantle Cell Lymphoma (MCL)'}  
    RANDOM_SEED: 42
```

5.2.6 Dataset Split Summary

```
Creating train/val/test split...  
... This ensures NO data leakage between sets.  
  
=====  
DATA SPLIT STATISTICS (NO LEAKAGE)  
=====  
  
Follicular Lymphoma (FL):  
    Total: 5000 images  
    Train: 3500 (70.0%)  
    Val: 750 (15.0%)  
    Test: 750 (15.0%)  
  
Mantle Cell Lymphoma (MCL):  
    Total: 5000 images  
    Train: 3500 (70.0%)  
    Val: 750 (15.0%)  
    Test: 750 (15.0%)  
  
Chronic Lymphocytic Leukemia (CLL):  
    Total: 5000 images  
    Train: 3500 (70.0%)  
    Val: 750 (15.0%)  
    Test: 750 (15.0%)  
  
=====  
✓ Split complete! Train/Val/Test are now completely independent.  
Test set is held-out and will ONLY be used for final evaluation.
```

5.2.7 Model Architecture Summa

=====			
BASELINE MODEL: Hybrid CNN + Vision Transformer (H-ViT-NHL)			
=====			
Model: "H-ViT-NHL"			
Layer (type)	Output Shape	Param #	Connected to
input_image (InputLayer)	(None, 128, 128, 3)	0	-
Conv1 (Conv2D)	(None, 64, 64, 32)	864	input_image[0][0]
bn_Conv1 (BatchNormalization)	(None, 64, 64, 32)	128	Conv1[0][0]
Conv1_relu (ReLU)	(None, 64, 64, 32)	0	bn_Conv1[0][0]
expanded_conv_dept... (DepthwiseConv2D)	(None, 64, 64, 32)	288	Conv1_relu[0][0]
expanded_conv_dept... (BatchNormalization)	(None, 64, 64, 32)	128	expanded_conv_de...
expanded_conv_dept... (ReLU)	(None, 64, 64, 32)	0	expanded_conv_de...
expanded_conv_proj... (Conv2D)	(None, 64, 64, 16)	512	expanded_conv_de...
expanded_conv_proj... (BatchNormalization)	(None, 64, 64, 16)	64	expanded_conv_pr...
(BatchNormalizatio...	(None, 1280)	0	-
out_relu (ReLU)	(None, 4, 4, 1280)	0	Conv_1_bn[0][0]
reshape_1 (Reshape)	(None, 16, 1280)	0	out_relu[0][0]
dense_6 (Dense)	(None, 16, 128)	163,968	reshape_1[0][0]
add_1 (Add)	(None, 16, 128)	0	dense_6[0][0]
transformer_block_1 (TransformerBlock)	(None, 16, 128)	198,400	add_1[0][0]
global_average_poo... (GlobalAveragePool...)	(None, 128)	0	transformer_bloc...
dense_9 (Dense)	(None, 128)	16,512	global_average_p...
dropout_10 (Dropout)	(None, 128)	0	dense_9[0][0]
dense_10 (Dense)	(None, 64)	8,256	dropout_10[0][0]
dropout_11 (Dropout)	(None, 64)	0	dense_10[0][0]
dense_11 (Dense)	(None, 3)	195	dropout_11[0][0]
Total params: 2,645,315 (10.09 MB)			
Trainable params: 387,331 (1.48 MB)			
Non-trainable params: 2,257,984 (8.61 MB)			
=====			

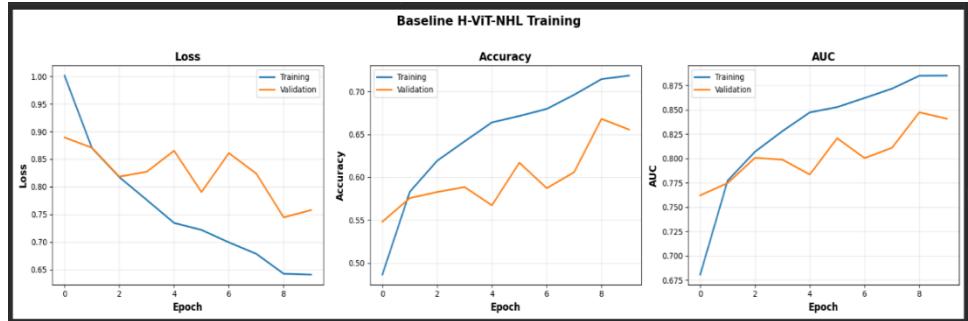
5.2.8 Model Training Process

```

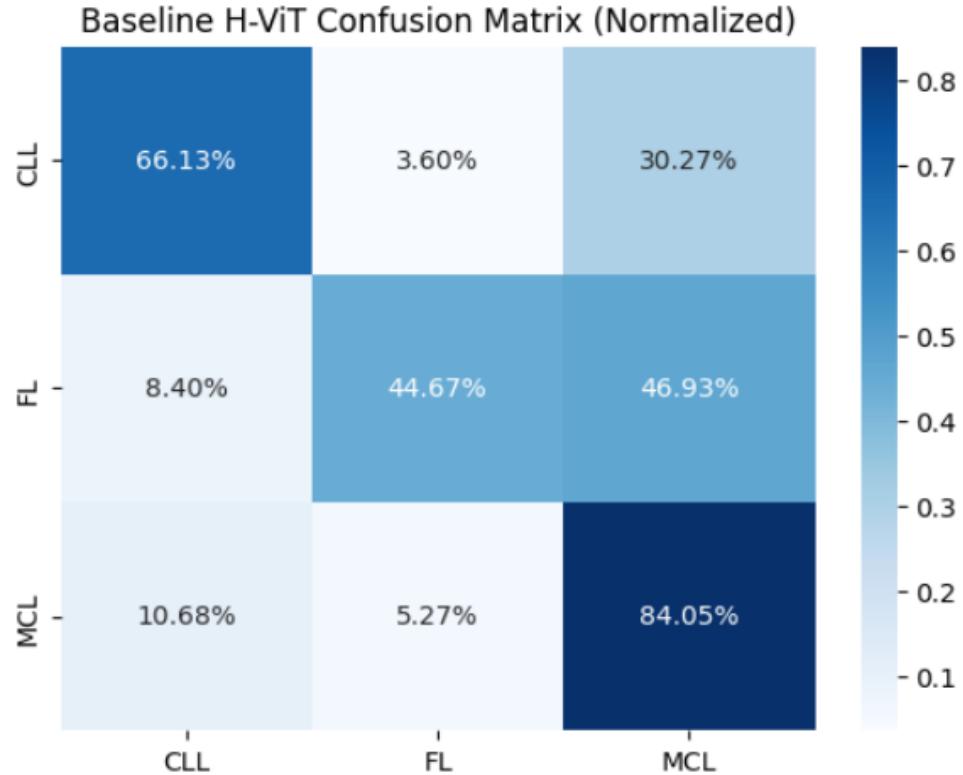
Epoch 1/10
329/329   0s 3s/step - accuracy: 0.4296 - auc: 0.6154 - loss: 1.0644
Epoch 1: val_accuracy improved from -inf to 0.54800, saving model to /content/drive/MyDrive/NHL_Project/models/best_hvit_baseline.keras
329/329   1119s 3s/step - accuracy: 0.4298 - auc: 0.6156 - loss: 1.0642 - val_accuracy: 0.5480 - val_auc: 0.7621 - val_loss: 0.8893 - learning_rate: 1.0000e-04
Epoch 2/10
329/329   0s 3s/step - accuracy: 0.5697 - auc: 0.7642 - loss: 0.8897
Epoch 2: val_accuracy improved from 0.54800 to 0.57556, saving model to /content/drive/MyDrive/NHL_Project/models/best_hvit_baseline.keras
329/329   1012s 3s/step - accuracy: 0.5697 - auc: 0.7642 - loss: 0.8897 - val_accuracy: 0.5756 - val_auc: 0.7747 - val_loss: 0.8707 - learning_rate: 1.0000e-04
Epoch 3/10
329/329   0s 3s/step - accuracy: 0.6070 - auc: 0.7977 - loss: 0.8330
Epoch 3: val_accuracy improved from 0.57556 to 0.62827, saving model to /content/drive/MyDrive/NHL_Project/models/best_hvit_baseline.keras
329/329   1044s 3s/step - accuracy: 0.6071 - auc: 0.7977 - loss: 0.8330 - val_accuracy: 0.6282 - val_auc: 0.8006 - val_loss: 0.8183 - learning_rate: 1.0000e-04
Epoch 4/10
329/329   0s 3s/step - accuracy: 0.6358 - auc: 0.8255 - loss: 0.7882
Epoch 4: val_accuracy improved from 0.62827 to 0.63844, saving model to /content/drive/MyDrive/NHL_Project/models/best_hvit_baseline.keras
329/329   1015s 3s/step - accuracy: 0.6358 - auc: 0.8255 - loss: 0.7882 - val_accuracy: 0.6384 - val_auc: 0.7988 - val_loss: 0.8272 - learning_rate: 1.0000e-04
Epoch 5/10
329/329   0s 3s/step - accuracy: 0.6596 - auc: 0.8423 - loss: 0.7485
Epoch 5: val_accuracy did not improve from 0.58844
329/329   1044s 3s/step - accuracy: 0.6596 - auc: 0.8423 - loss: 0.7484 - val_accuracy: 0.5671 - val_auc: 0.7834 - val_loss: 0.8653 - learning_rate: 1.0000e-04
Epoch 6/10
329/329   0s 3s/step - accuracy: 0.6540 - auc: 0.8518 - loss: 0.7241
Epoch 6: val_accuracy improved from 0.58844 to 0.61689, saving model to /content/drive/MyDrive/NHL_Project/models/best_hvit_baseline.keras
329/329   1045s 3s/step - accuracy: 0.6540 - auc: 0.8510 - loss: 0.7241 - val_accuracy: 0.6169 - val_auc: 0.8207 - val_loss: 0.7904 - learning_rate: 1.0000e-04
Epoch 7/10
329/329   0s 3s/step - accuracy: 0.6815 - auc: 0.8633 - loss: 0.6977
Epoch 7: val_accuracy did not improve from 0.61689
329/329   993s 3s/step - accuracy: 0.6815 - auc: 0.8633 - loss: 0.6977 - val_accuracy: 0.5871 - val_auc: 0.8003 - val_loss: 0.8611 - learning_rate: 1.0000e-04
Epoch 8/10
329/329   0s 3s/step - accuracy: 0.6902 - auc: 0.8670 - loss: 0.6915
Epoch 8: val_accuracy did not improve from 0.61689
329/329   993s 3s/step - accuracy: 0.6902 - auc: 0.8670 - loss: 0.6914 - val_accuracy: 0.6058 - val_auc: 0.8109 - val_loss: 0.8238 - learning_rate: 1.0000e-04
Epoch 9/10
329/329   0s 3s/step - accuracy: 0.7091 - auc: 0.8881 - loss: 0.6568
Epoch 9: val_accuracy improved from 0.61689 to 0.66800, saving model to /content/drive/MyDrive/NHL_Project/models/best_hvit_baseline.keras
329/329   997s 3s/step - accuracy: 0.7091 - auc: 0.8892 - loss: 0.6568 - val_accuracy: 0.6680 - val_auc: 0.8473 - val_loss: 0.7445 - learning_rate: 1.0000e-04
Epoch 10/10
329/329   0s 3s/step - accuracy: 0.7124 - auc: 0.8811 - loss: 0.6513
329/329   1082s 3s/step - accuracy: 0.7124 - auc: 0.8811 - loss: 0.6513 - val_accuracy: 0.6556 - val_auc: 0.8488 - val_loss: 0.7577 - learning_rate: 1.0000e-04
Restoring model weights from the end of the best epoch: 9.

```

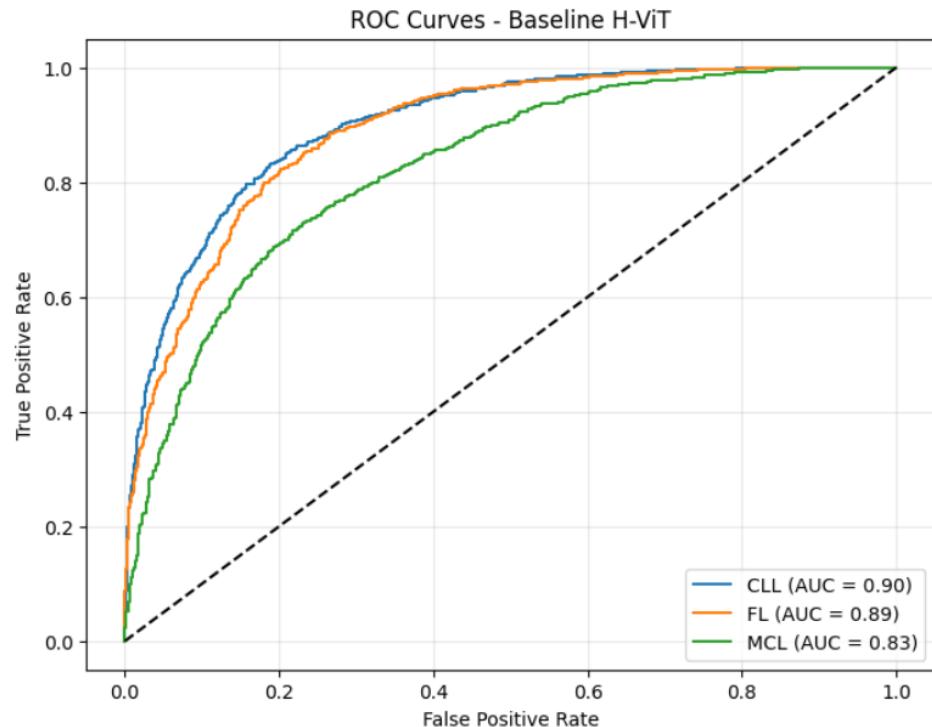
5.2.9 Training and Validation Performance Curves



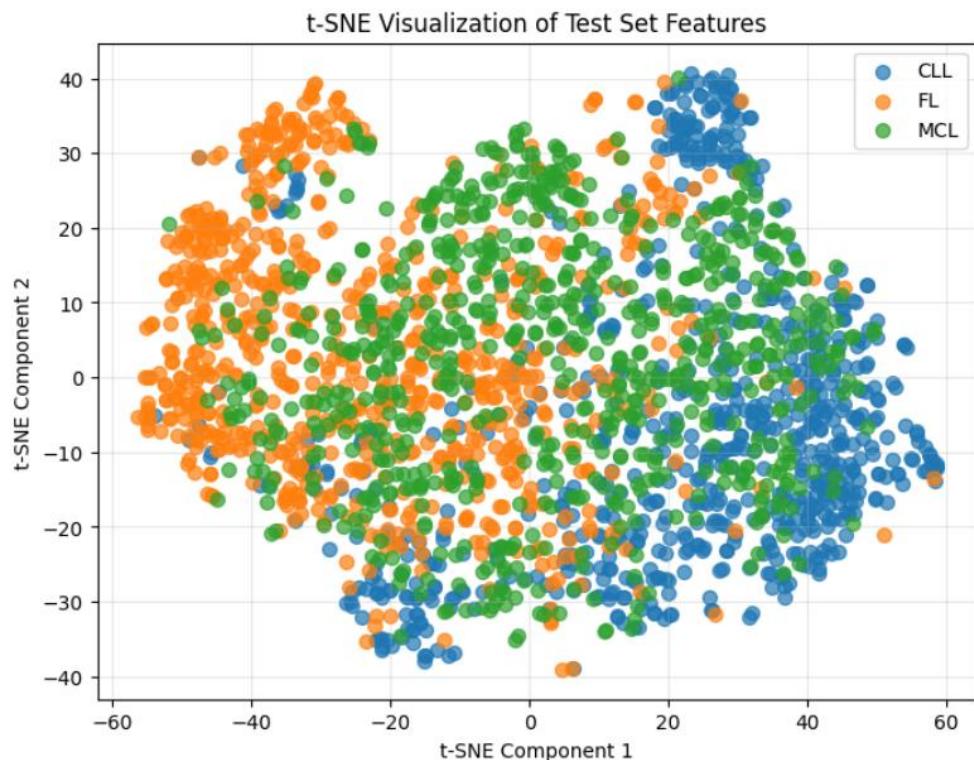
5.2.10 Confusion Matrix Analysis



5.2.11 ROC Curve Analysis



5.2.12 t-SNE Feature Space Visualization



5.2.13 Classification Performance Report

```
=====
BASELINE CLASSIFICATION REPORT (HELD-OUT TEST SET)
=====
precision    recall   f1-score   support
CLL      0.7774   0.6613   0.7147    750
FL       0.8354   0.4467   0.5821    750
MCL      0.5179   0.8405   0.6409    740
accuracy          0.6487    2240
macro avg     0.7102   0.6495   0.6459    2240
weighted avg   0.7111   0.6487   0.6459    2240
=====
```

5.2.14 Baseline Model ROC-AUC Scores

```
Baseline ROC-AUC (Held-Out Test):
CLL: 0.9010
FL: 0.8894
MCL: 0.8263
Macro-Average: 0.8722
```

REFERENCES

- [1] H. Khelil, A. El Moumene Zerari, and L. Djerou, "Accurate diagnosis of non-Hodgkin lymphoma on whole-slide images using deep learning," 2022 IEEE 9th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), Hammamet, Tunisia, 2022, pp. 447-451, doi: 10.1109/SETIT54465.2022.9875482.
- [2] D. S and M. Chandra Sekhar, "Non-Hodgkin Lymphoma Classification Using Multi-Scale Attention Mechanism with Convolutional Neural Networks," 2025 3rd International Conference on Data Science and Network Security (ICDSNS), Tiptur, India, 2025, pp. 1-5, doi: 10.1109/ICDSNS65743.2025.11168566.
- [3] G. P. Sri and K. Ananthajothi, "Enhanced Malignant Lymphoma Classification Using an Explainable Dilated MobileNetV2 and Convolutional-Recurrent Neural Network," 2025 11th International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 2025, pp. 1386-1391, doi: 10.1109/ICCSP64183.2025.11088498.
- [4] N. K. H N, B. H S, M. Junaid, S. S, R. S and S. K S, "Comprehensive Study on Lymphoma Detection Using Deep Learning," 2025 3rd International Conference on Inventive Computing and Informatics (ICICI), Bangalore, India, 2025, pp. 923-927, doi: 10.1109/ICICI65870.2025.11069739.
- [5] A. Kumar, L. Nelson, and D. Arumugam, "Blood Cancer Diagnosis Using Pretrained VGG16 Transfer Learning Model with Lymphoma Dataset," 2024 Asian Conference on Intelligent Technologies (ACOIT), KOLAR, India, 2024, pp. 1-6, doi: 10.1109/ACOIT62457.2024.10941174.
- [6] M. Habijan and I. Galić, "Ensemble Transfer Learning for Lymphoma Classification," 2024 31st International Conference on Systems, Signals and Image Processing (IWSSIP), Graz, Austria, 2024, pp. 1-6, doi: 10.1109/IWSSIP62407.2024.10634025.
- [7] A. A. R. Riyanto et al., "Lymphoma sub-type classification using DenseNet 169-based Transfer Learning Architecture," 2024 International Seminar on Application for

Technology of Information and Communication (iSemantic), Semarang, Indonesia, 2024, pp. 509-514, doi: 10.1109/iSemantic63362.2024.10762354.

[8] S. S. Maghdid, S. Jamal Al-Atroshi, C. H. Salh, and S. W. Kareem, "Deep Learning for the Detection of Skin Cancer: A Comprehensive Review," 2024 10th International Engineering Conference on Advances in Computer and Civil Engineering (IEC), Erbil, Iraq, 2024, pp. 155-160, doi: 10.1109/IEC61018.2024.11063749.

[9] M. Mohana, R. Rithika, and V. Sivasakthi, "Revolutionizing Lymphoma Diagnosis with Deep Learning and Natural Language Generation," 2024 2nd International Conference on Computer, Communication and Control (IC4), Indore, India, 2024, pp. 1-11, doi: 10.1109/IC457434.2024.10486592.

[10] A. Kaur, V. Kukreja, N. Thapliyal, M. Manwal, and R. Sharma, "An Efficient Fine-tuned GoogleNet Model for Multiclass Classification of Blood Cell Cancer," 2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), Gwalior, India, 2024, pp. 1-5, doi: 10.1109/IATMSI60426.2024.10502531.

[11] K. R. Aravind, K. A. Kumar, V. V. D. Reddy, and P. Yadlapalli, "Leveraging Convolutional Neural Networks for Classifying Lymphoma Using Histopathological Images," 2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS), Bangalore, India, 2024, pp. 1-6, doi: 10.1109/ICITEICS61368.2024.10625210.

[12] S. M, P. K. D, and S. P, "Data Mining Approaches in Healthcare Industry," 2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2023, pp. 1019-1025, doi: 10.1109/ICIRCA57980.2023.10220705.

[13] R. K. Gupta et al., "Classification and Morphological Analysis of DLBCL Subtypes in H&E-Stained Slides," 2024 IEEE 24th International Conference on Bioinformatics and Bioengineering (BIBE), Kragujevac, Serbia, 2024, pp. 1-8, doi: 10.1109/BIBE63649.2024.10820463.

[14] Archita, C. Prabha, A. Nath, and R. Singh, "Enhancing Lymphoma Diagnosis: Transfer Learning with DenseNet201 for Subtype Classification," 2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2024, pp. 664-669, doi: 10.1109/ICAAIC60222.2024.10575531.

[15] Y. Jusman, R. O. Ningrum, and M. A. Fawwaz Nurkholid, "Leukemia Cell Image Classification Using CNN: AlexNet and GoogLeNet," 2023 8th International Conference on Electrical, Electronics and Information Engineering (ICEEIE), Malang City, Indonesia, 2023, pp. 1-6, doi: 10.1109/ICEEIE59078.2023.10334867.

[16] M. Pasha, K. K. Ata, and V. V. Kishore, "Optimized ensemble learning for lung and colon cancer classification using histopathology images from LC25000 dataset," in 2025 International Conference on Electronics and Renewable Systems (ICEARS), 2025, pp. 1874–1879.

[17] H. H. Hai, S. Manoharan, S. D. M., and K. Raja, "An explainable transfer learning framework for breast cancer histopathology classification using CNNs," in IEEE 7th International Conference on Computing, Communication and Automation (ICCCA), 2025, pp. 1–7.

[18] P. Subramaniyam and G. H. Krishnan, "Fusion learning framework for malignant and benign classification in histopathology images," in 3rd International Conference on Intelligent Data Communication Technologies and IoT (IDCIoT), 2025, pp. 1692–1696.

[19] G. Kaur and N. Sharma, "Invasive ductal carcinoma detection using CNN architecture from breast histopathology images," in Second International Conference on Intelligent Cyber Physical Systems and IoT (ICoICI), 2024, pp. 1470–1475.

[20] V. P. V., P. Chhattu, K. Sivasankari, D. T. Pisal, B. Renuka Sai, and D. Suganthi, "Exploring convolution neural networks for image classification in medical imaging," in International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE), 2024, pp. 1–4.

[21] M. Jia, X. Yan, and S. Fu, "Histopathologic cancer detection based on deep multiple instance learning," in 15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN), 2019, pp. 368–371.

[22] T. D. Pham, "Class fusion of support vector machines with deep learning features for oral cancer histopathology classification," in IEEE 22nd International Symposium on Biomedical Imaging (ISBI), 2025, pp. 1–4.

[23] Y. Yari and H. Nguyen, "A state-of-the-art deep transfer learning-based model for accurate breast cancer recognition in histology images," in 20th International Conference on Bioinformatics and Bioengineering (BIBE), 2020, pp. 900–905.

[24] S. S. M. Khairi, M. A. A. Bakar, M. A. Alias, S. A. Bakar, and C.-Y. Liong, "A preliminary study of convolutional neural network architectures for breast cancer image classification," in IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), 2021, pp. 1–5.

[25] G. Kaur, N. Sharma, and R. Gupta, "Enhanced detection of oral squamous cell carcinoma from histopathological images using CNN architectures," in Second International Conference on Intelligent Cyber Physical Systems and IoT (ICoICI), 2024, pp. 1525–1530.

[26] W. R. Quinones, M. Ashraf, and M. Y. Yi, "Impact of patch extraction variables on histopathological imagery classification using convolution neural networks," in 2021 International Conference on Computational Science and Computational Intelligence (CSCI), 2021, pp. 1176–1181.

[27] P. T. Nguyen, T. T. Nguyen, N. C. Nguyen, and T. T. Le, "Multiclass breast cancer classification using convolutional neural network," in 2019 International Symposium on Electrical and Electronics Engineering (ISEE), 2019, pp. 130–134.

[28] S. P., S. K. V., and S. M. C. M. S., "Deep learning based multi class epithelial ovarian cancer classification from histopathological images," in 2025 IEEE International

Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), 2025, pp. 1–6.

[29] H. Duan, Y. Liu, H. Yan, Q. He, Y. He, and T. Guan, "Fourier ViT: A multi-scale vision transformer with Fourier transform for histopathological image classification," in 2022 7th International Conference on Automation, Control and Robotics Engineering (CACRE), 2022, pp. 189–193.

[30] S. B. Mukadam and H. Y. Patil, "Fusion of ESRGAN, adaptive NSCT, and multi-attention CNN with wavelet transform for histopathological image classification," IEEE Access, vol. 12, pp. 129977–129993, 2024.

[31] A. Johny, K. N. Madhusoodanan, and S. Cyriac, "Edge computing based miniature maps using embedded webserver for prediction of malignancy," in 2022 6th International Conference on Devices, Circuits and Systems (ICDCS), 2022, pp. 268–271.

[32] D. Venugopal, M. Mesbah, R. Hmouz, A. Qureshi, and A. Ammari, "Comparative analysis of diverse activation functions and kernel methods for thyroid cancer classification," in 2025 International Conference on Emerging Technologies in Engineering Applications (ICETEA), 2025, pp. 1–6.

[33] B. Swarnkar, P. Maheshwari, N. Khare, and M. Gyanchandani, "Early diagnosis of endometrial cancer: An ensemble-based deep learning approach," in 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), 2024, pp. 1–7.

[34] U. Varman, V. Bharti, A. Sharma, A. Kumar, and S. K. Singh, "When explainability meets vision AI: Analyzing CNNs, transformers, and state-space models in healthcare," in 2025 International Joint Conference on Neural Networks (IJCNN), 2025, pp. 1–8.

[35] Y. Yari, T. V. Nguyen, and H. T. Nguyen, "Deep learning applied for histological diagnosis of breast cancer," IEEE Access, vol. 8, pp. 162432–162448, 2020.

- [36] N. Bhatt, S. Goswami, P. K, Swati, D. Srivalli, and S. Singh, "Lightweight deep learning model for breast cancer malignancy prediction using CNN on embedded edge devices," in 2025 World Skills Conference on Universal Data Analytics and Sciences (WorldSUAS), 2025, pp. 1–6.
- [37] S. Singla and R. Gupta, "Optimizing cancer detection: A CNN approach for lung and colon cancer," in 2024 4th International Conference on Ubiquitous Computing and Intelligent Information Systems (ICUIS), 2024, pp. 62–67.
- [38] X. Qian et al., "SPCB-Net: A multi-scale skin cancer image identification network using self-interactive attention pyramid and cross-layer bilinear-trilinear pooling," IEEE Access, vol. 12, pp. 2272–2287, 2024.
- [39] K. Vanitha et al., "Attention-based feature fusion with external attention transformers for breast cancer histopathology analysis," IEEE Access, vol. 12, pp. 126296–126312, 2024.
- [40] B. L. Y. Agbley, M. F. Nartey, S. I. Akwaboa, C. N. Osei, and S. S. N. Antwi, "Federated Fusion of Magnified Histopathological Images for Breast Tumor Classification in the Internet of Medical Things," IEEE J. Biomed. Health Inform., vol. 28, no. 6, pp. 3389–3400, Jun. 2024, doi: 10.1109/JBHI.2023.3256974.
- [41] I. Das, S. S. Kale, P. Gupta, and R. K. Sharma, "Improving Medical X-Ray Imaging Diagnosis With Attention Mechanisms and Robust Transfer Learning Techniques," IEEE Access, vol. 13, pp. 159002–159027, 2025, doi: 10.1109/ACCESS.2025.3607639.
- [42] D. Varam, K. S. R. Anjaneyulu, A. A. Khan, and S. A. Hussain, "Wireless Capsule Endoscopy Image Classification: An Explainable AI Approach," IEEE Access, vol. 11, pp. 105262–105280, 2023, doi: 10.1109/ACCESS.2023.3319068.
- [43] M. Y. Shakor and M. I. Khaleel, "Modern Deep Learning Techniques for Big Medical Data Processing in Cloud," IEEE Access, vol. 13, pp. 62005–62028, 2025, doi: 10.1109/ACCESS.2025.3556327.

[44] Z. U. Rehman, M. F. A. Fauzi, F. N. I. Lokman, M. Touhami, and L. Saim, "Efficient and Interpretable Otoscopic Image Classification via Distilled CNN With Adaptive Channel Attention," IEEE Access, vol. 13, pp. 151082–151096, 2025, doi: 10.1109/ACCESS.2025.3597769.

[45] H. ELwahsh, A. H. A. El-Latif, A. K. Jha, and M. S. Al-Ahmadi, "Explainable Artificial Intelligence in Malignant Lymphoma Classification: Optimized DenseNet121 Deep Learning Approach With Particle Swarm Optimization and Genetic Algorithm," IEEE Access, vol. 13, pp. 98639–98655, 2025, doi: 10.1109/ACCESS.2025.3575364.

[46] D. R. Kothadiya, C. M. Bhatt, A. Rehman, F. S. Alamri, and T. Saba, "SignExplainer: An Explainable AI-Enabled Framework for Sign Language Recognition With Ensemble Learning," IEEE Access, vol. 11, pp. 47410–47419, 2023, doi: 10.1109/ACCESS.2023.3274851.

[47] M. A. Inamdar, A. R. Patil, S. K. Patil, and S. R. Patil, "A Dual-Stream Deep Learning Architecture With Adaptive Random Vector Functional Link for Multi-Center Ischemic Stroke Classification," IEEE Access, vol. 13, pp. 46638–46658, 2025, doi: 10.1109/ACCESS.2025.3550344.

[48] P. Bissoonauth-Daiboo, A. R. Hossain, S. M. Rahman, and T. Al-Salman, "Exploring Vision Transformers and Explainable AI for Enhanced Artefact Classification in Esophageal Endoscopic Images," IEEE Access, vol. 13, pp. 176221–176244, 2025, doi: 10.1109/ACCESS.2025.3616796.

[49] M. Radhakrishnan, N. Sampathila, H. Muralikrishna, and K. S. Swathi, "Advancing Ovarian Cancer Diagnosis Through Deep Learning and eXplainable AI: A Multiclassification Approach," IEEE Access, vol. 12, pp. 116968–116986, 2024, doi: 10.1109/ACCESS.2024.3448219.

[50] D. Saraswat, A. Sharma, R. Gupta, and S. Singh, "Explainable AI for Healthcare 5.0: Opportunities and Challenges," IEEE Access, vol. 10, pp. 84486–84517, 2022, doi: 10.1109/ACCESS.2022.3197671.

[51] S. Sangnark, P. Rattanachaisit, T. Patchatrakul, and P. Vateekul, "Explainable Multi-Modal Deep Learning With Cross-Modal Attention for Diagnosis of Dyssynergic Defecation Using Abdominal X-Ray Images and Symptom Questionnaire," IEEE Access, vol. 12, pp. 78132–78147, 2024, doi: 10.1109/ACCESS.2024.3409077.

[52] R. Karthik, A. Ajay, A. S. Bisht, J. Cho, and V. E. Sathishkumar, "An Explainable Deep Learning Network for Environmental Microorganism Classification Using Attention-Enhanced Semi-Local Features," IEEE Access, vol. 12, pp. 151770–151784, 2024, doi: 10.1109/ACCESS.2024.3462592.

[53] A. Amin, K. Hasan, and M. S. Hossain, "XAI-Empowered MRI Analysis for Consumer Electronic Health," IEEE Trans. Consum. Electron., vol. 71, no. 1, pp. 1423–1431, Feb. 2025, doi: 10.1109/TCE.2024.3443203.

[54] C. V. Aravinda, E. R. Joseph, and S. Alasmari, "Optimized DenseNet121 and Quantum PennyLane Fusion for Explainable Skin Disease Recognition and Classification," IEEE Access, vol. 13, pp. 167957–167969, 2025, doi: 10.1109/ACCESS.2025.3608217.

[55] F. Haque, M. R. Uddin, K. Hasan, and M. S. Hossain, "An End-to-End Concatenated CNN Attention Model for the Classification of Lung Cancer With XAI Techniques," IEEE Access, vol. 13, pp. 96317–96336, 2025, doi: 10.1109/ACCESS.2025.3572423.

[56] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you? Explaining the predictions of any classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, San Francisco, CA, USA, 2016, pp. 1135–1144.

[57] G. Samek, T. Wiegand, and K.-R. Müller, "Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models," *ITU Journal: ICT Discoveries*, vol. 1, no. 1, pp. 1–10, 2017.

[58] R. R. Selvaraju et al., "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 618–626.

- [59] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, pp. 4765–4774, 2017.
- [60] G. Litjens et al., “A survey on deep learning in medical image analysis,” *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.
- [61] A. Esteva et al., “A guide to deep learning in healthcare,” *Nature Medicine*, vol. 25, no. 1, pp. 24–29, 2019.
- [62] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [63] A. Holzinger et al., “What do we need to build explainable AI systems for the medical domain?” *arXiv preprint arXiv:1712.09923*, 2017.
- [64] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [65] F. Doshi-Velez and B. Kim, “Towards a rigorous science of interpretable machine learning,” *arXiv preprint arXiv:1702.08608*, 2017.
- [66] European Commission High-Level Expert Group on Artificial Intelligence, *Ethics Guidelines for Trustworthy AI*. Brussels, Belgium: European Union, 2019.

Web links:

- [1] <https://github.com/preyanshnahar/AN-EXPLAINABLE-DEEP-NEURAL-NETWORK-APPROACH-FOR-NON-HODGKIN-LYMPHOMA-CLASSIFICATION>