HyMaC-Net: A Hybrid Lightweight Mamba-CNN Framework with Patch Embedding for Medical Image Classification

by

Sourav Das
24341223
Shibam Chakraborty
24241278
Sakibul Hasan Turja
24341222
Md.Tanvirul Islam Rifat
22101311

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
October 2025.

Declaration

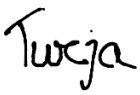
It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Sourav Das
24341223

Shibam Chakraborty
24241278



Sakibul Hasan Turja
24341222

Md.Tanvirul Islam Rifat
22101311

Approval

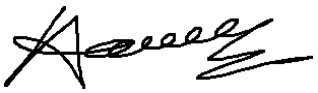
The thesis/project titled "HyMaC-Net: A Lightweight Mamba-CNN Framework with Patch Embedding for Medical Image Classification" submitted by

1. Sourav Das (24341223)
2. Shibam Chakraborty (24241278)
3. Sakibul Hasan Turja (24341222)
4. Md.Tanvirul Islam Rifat (22101311)

of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in Summer 2025.

Examining Committee:

Supervisor:
(Member)



Dr. Amitabha Chakrabarty
Professor
Department of Computer Science and Engineering
BRAC University

Co-Supervisor:
(Member)

Nirjhor Datta
Lecturer
Department of Computer Science and Engineering
BRAC University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Chairperson and Associate Professor
Department of Computer Science and Engineering
BRAC University

Abstract

Designing medical image classifiers that generalize across heterogeneous datasets while remaining computationally lean remains a challenge. Recent work explores hybrids that mix local convolutional priors with global sequence modeling (e.g., Transformers or state-space models), yet there is limited empirical guidance on which design choices actually matter under strict compute budgets. This study presents a systematic ablation study in 12 datasets (MedMNIST-2D subsets plus CPN X-ray and Kvasir) that probes key knob depth, tokenization granularity, normalization/regularization, pooling, and the use of Mamba state-space blocks for long-range dependency modeling. Rather than chasing single-dataset SOTA, our goal is to map the accuracy compute frontier with transparent metrics (ACC/AUC, parameters, GMACs, and inference time) and seed-robust statistics. The study yields two practical profiles (*Small* and *Base*) of the proposed HyMaC-Net a hybrid mamba-cnn architecture that deliver competitive performance across datasets while staying within modest compute envelopes. In particular, in the Kvasir dataset, both Base (84.67%) and Small (83.42%) models acquired 4-5% more precision than the existing models while having fewer parameters and GMACs. On the other hand, on the OrganAMNIST dataset, both the Base (97.8%) and Small (95.3%) models surpassed the existing model with a 2% accuracy increment. Results are consistent with the premise that Mamba SSM blocks can replace attention to capture global context efficiently and that careful architectural pruning (fewer blocks, fused pooling, auxiliary guidance) preserves accuracy at substantially reduced cost. Furthermore, we include an interpretability check using Grad-CAM to ensure that model predictions are based on clinically relevant features. We release per-dataset metrics (OA, AUC, precision, recall, specificity, F1, Cohen's κ, Dice) and ablation logs to support reproducibility and fair comparison. The resulting guidelines are intended to help physicians build deployable medical classifiers without exhaustive tuning.

Keywords: Medical Image Classification, Hybrid Models, Mamba (State-Space Models), Self-attention, Compute Efficiency, Generalization, MedMNIST, GRAD-CAM, Interpretability.

Acknowledgement

We would first like to express our deepest gratitude to the Almighty, whose infinite grace, strength, and wisdom have guided us throughout every stage of this endeavor. Without His divine blessings, this accomplishment would not have been possible. We are profoundly thankful to our esteemed supervisor, Dr. Amitabha Chakraborty, whose exceptional guidance, insightful advice, and unwavering support have been invaluable to the success of this research. Our sincere appreciation also goes to our co-supervisor, Nirjhor Datta, for his thoughtful suggestions, encouragement, and continuous motivation, which have significantly contributed to enhancing the quality of our work.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Medical imaging is now central to clinical diagnosis and treatment planning. Modalities such as CT, MRI, X-ray, ultrasound, and pathology slides produce vast amounts of data every day in hospitals. These images provide valuable insight into the human body, but they also create new challenges. Radiologists often face heavy workloads, and manual interpretation is slow, subjective, and prone to error [29],[27],[36]. This situation has made automated medical image classification an important research direction. Deep learning, especially Convolutional Neural Networks (CNNs), has shown strong success in this field. CNNs are good at extracting local features such as textures, shapes, and edges, which makes them effective for identifying tumors, lesions, or tissue patterns [19]. However, CNNs focus mainly on local regions and struggle to capture global context, which is critical when a diagnosis depends on understanding how different regions of an image relate to one another.

To overcome this limitation, Vision Transformers (ViTs) were introduced into computer vision and later adapted for medical imaging. Unlike CNNs, ViTs can capture long-range dependencies and global structures through self-attention. This makes them powerful in recognizing complex relationships in medical images [18],[24]. However, their practical use in healthcare is limited. ViTs are computationally expensive, with quadratic complexity in relation to image size, and they often require large annotated datasets. Creating such datasets in medicine is difficult because annotations need expert knowledge and are costly to obtain. Hybrid CNN–ViT models try to combine local and global learning, but they still suffer from inefficiency and often require large GPU resources that are not available in many hospitals.

Recently, state space models (SSMs) such as Mamba have provided a new path forward. These models can capture long-range dependencies like Transformers but do so with linear computational complexity, making them more efficient for high-resolution images [34]. This efficiency makes them promising for medical image classification. At the same time, patch embedding has become an effective way to reduce redundancy in large medical images by dividing them into smaller patches and turning them into compact feature vectors. Combining CNNs for local features, Mamba for global context, and patch embedding for efficiency creates a strong opportunity to design a lightweight and scalable model. This is the motivation for HyMEx-Net, a new framework that aims to meet the real-world needs of medical imaging: accuracy, robustness, and computational efficiency in a single architecture.

1.2 Motivation

Medical image classification demands models that are accurate, efficient, and reliable, yet current solutions fail to fully meet these needs. CNNs capture local features well but ignore global context, while Transformers model long-range dependencies but are computationally heavy and require large annotated datasets, which are difficult to obtain in healthcare [41]. Hybrid CNN–Transformer models improve performance but remain resource-intensive and less suitable for real clinical settings [45].

Although CNNs, Transformers, and SSMs have individually advanced medical image understanding, there remains a critical need for a unified, lightweight, and generalizable architecture capable of performing effectively across multiple modalities and datasets. Existing Transformer or CNN architectures often excel in specific domains but struggle to maintain consistent performance across diverse data distributions—an issue commonly known as domain shift. In medical imaging, such domain variability is unavoidable due to differences in imaging equipment, acquisition protocols, and patient demographics. Consequently, models that rely heavily on large datasets or high-resolution global attention mechanisms often fail to generalize when applied to unseen data.

This gap motivated the development of HyMaC-Net (Hybrid Mamba-Convolution Network), a model that combines CNN-based local feature extraction integrating patch embedding to reduce redundancy with Mamba-based global dependency modeling. The design aims to achieve the robustness of CNNs, the contextual awareness of Transformers, and the efficiency of state-space architectures all within a single, scalable framework. HyMaC-Net is intended to operate efficiently across datasets of varying sizes and complexity, providing a strong balance between computational cost and predictive performance.

1.3 Problem Statement

Despite significant progress in medical deep learning, existing methods face three fundamental challenges:

1. **Limited global context modeling:** Convolutional Neural Networks (CNNs) focus on local receptive fields, missing long-range feature relationships that are crucial for complex tissue or organ structures.

2. **High computational cost and data dependency:** Vision Transformers and large hybrid models demand extensive training data and GPU resources, limiting their use in real-world clinical settings.

3. **Weak cross-domain generalization:** Many models exhibit sharp performance degradation when tested on external datasets, mainly due to domain shift and overfitting to specific distributions.

Therefore, there is a need for a lightweight yet generalizable hybrid model that efficiently integrates local and global feature extraction while maintaining robustness and interpretability across different imaging modalities.

1.4 Objective

The main goal of this research is to design and validate a lightweight hybrid deep learning model (HyMaC-Net) that bridges convolutional locality and state-space global dependency modeling for generalized medical image classification. The specific objectives are:

1. To develop a hybrid CNN–SSM framework that combines convolutional stems and Vision-Mamba blocks with patch-embedding for efficient multi-scale representation learning.

2. To evaluate HyMaC-Net on diverse medical datasets, including the MedMNIST collection and other domain-specific benchmarks, covering modalities such as X-ray, pathology, CT, and endoscopy.

3. To compare HyMaC-Net with state-of-the-art architectures such as ResNet, MedViT, and MedMamba, assessing performance using accuracy, AUC, FLOPs, parameter count, and inference time.

4. To conduct statistical analysis, including seed-sensitivity evaluation and Wilcoxon signed-rank testing, for validating the model's reliability and significance of improvements.

5. To examine the interpretability of HyMaC-Net using Grad-CAM visualizations, highlighting class-discriminative regions and verifying clinical relevance of the learned representations.

6. To assess cross-dataset generalization and identify how efficiently the proposed architecture adapts to new medical imaging domains.

1.5 Methodology in Brief

The study follows a structured methodological pipeline encompassing data preparation, model development, training, and evaluation:

- **Dataset Preparation:** A collection of standardized and heterogeneous datasets from MedMNIST v2 and external sources (e.g., CPN X-ray, Kvasir, and Brain-Tumor) was curated. Preprocessing included resizing, normalization, and augmentation to ensure consistency and diversity.

- **Model Design:** The proposed HyMaC-Net integrates a multi-scale CNN stem for low-level texture learning, followed by patch embedding (4×4 patches) and Mamba-SSM blocks for long-range dependency modeling. An auxiliary classifier head is added to provide gradient supervision on the mid-layer module and to merge multi-level representations before the final classification head. Finally, two versions of HyMaC-Net (small & base) are proposed.

- **Training Configuration:** All models were trained using PyTorch with AdamW optimization, learning-rate scheduling, and early-stopping strategies. Experiments were repeated across multiple random seeds to ensure reproducibility.

- **Evaluation Metrics:** Model performance was quantified using metrics such as Accuracy, Precision, Recall, F1-score, AUC-ROC, Specificity and Cohen's Kapp-a. Efficiency metrics (parameters, FLOPs, and inference time) were also computed.
- **Statistical Validation:** Descriptive variability and Wilcoxon signed-rank tests were performed to assess the statistical significance of performance differences across random seeds and models.
- **Interpretability Check:** To enhance model transparency and clinical trustworthiness, post-hoc interpretability analysis was conducted using Grad-CAM. These visual explanations provided qualitative verification of whether the model focused on medically relevant anatomical or pathological regions.
- **Cross-Dataset Evaluation:** To assess generalization and robustness, a cross-dataset evaluation was conducted in which the model trained on one dataset was tested on distinct external datasets. This analysis evaluated model stability under domain-shift conditions and measured its ability to retain diagnostic discrimination across varied modalities.

This end-to-end methodology ensures a fair, reproducible, and statistically sound comparison between HyMaC-Net and competitive baselines.

1.6 Scopes and Challenges

The scope of this research spans generalized medical image classification, focusing on creating a versatile backbone applicable across different modalities and imaging conditions. HyMaC-Net is designed to serve as a deployable and interpretable framework capable of supporting tasks such as lesion recognition, tissue characterization, and diagnostic triage in clinical workflows.

However, several challenges remain. The domain shift problem continues to be a significant obstacle, as variations in imaging devices and acquisition settings lead to inconsistencies between training and deployment data. Furthermore, the limited availability of annotated medical datasets restricts the potential of data-hungry deep learning models. Another challenge lies in balancing model complexity and interpretability, ensuring that performance improvements do not compromise transparency, which is vital for clinical acceptance.

Despite these limitations, the proposed HyMaC-Net demonstrates a practical direction toward efficient, generalizable, and explainable deep learning solutions in healthcare imaging. The findings of this research contribute to the ongoing evolution of hybrid architectures that combine convolutional inductive biases with efficient global modeling, paving the way for real-time and resource-aware medical AI systems.

Chapter 2

Literature Review

2.1 Preliminaries

Medical image classification has advanced rapidly with the growth of deep learning, offering efficient alternatives to manual diagnosis. Computer-Aided Diagnosis (CAD) systems now use machine learning and hybrid architectures to analyze CT, MRI, X-ray, and pathology images with high accuracy. Some of the key concepts related to this research are:

1. **Feature Extraction:** It involves methods like texture extraction, geometric analysis, color space transformation, etc. For example, Linear Discriminant Analysis (LDA) methods, Discrete Orthonormal S-Transform (DOST) and Principal Component Analysis (PCA), are used to preprocess and efficient classification.

2. **Deep Learning Models:** Convolutional Neural Networks (CNNs), Vision Transformer (ViT) and hybrid architectures are widely used for feature learning and classification tasks. Also, pre-trained models like AlexNet and EfficientNet were fine-tuned for ALL diagnoses.

3. **Patch Embedding:** Dividing medical images into smaller patches and projecting them into feature vectors reduces redundancy and speeds up computation, making it especially useful for high-resolution data.

4. **Transfer learning:** Transfer learning allows to fine-tune pre-trained models on a smaller, domain-specific dataset. This is especially useful for limited data availability.

2.2 Review of Existing Research

CNN Architectures

Convolutional Neural Networks (CNNs) have profoundly reshaped the landscape of computer vision and medical image analysis since the seminal work of Krizhevsky et al. (2012)[2], who introduced AlexNet, a deep CNN trained end-to-end on 1.2 million images from the ImageNet dataset (1000 classes). The model, with 60 million parameters and eight layers (five convolutional and three fully connected), achieved

state-of-the-art results (37.5% top-1 and 17.0% top-5 error), pioneering innovations such as ReLU activation, dropout regularization, local response normalization, and multi-GPU training, which made deep CNNs computationally feasible and scalable. Despite high computational demands and dependency on large labeled data, AlexNet marked the turning point for deep learning in vision, inspiring successors like VGG, GoogLeNet, and ResNet. Later, CNN applications extended toward medical imaging, as demonstrated by Yadav and Jadhav (2019)[15], who applied CNN-based transfer learning (VGG16, InceptionV3) to pneumonia classification using chest X-rays (5,232 training and 624 test images). Their study compared classical ORB+SVM and Capsule Networks against CNNs and found that moderate transfer learning with selective unfreezing of deeper convolutional blocks and data augmentation yielded superior generalization, while over-parameterization caused overfitting. Although limited by small, imbalanced datasets and a lack of cross-site validation, the work emphasized the importance of data augmentation and architecture scaling in medical diagnostics. More recently, Azeem et al. (2024)[32] proposed SkinLesNet, a lightweight four-layer CNN for classifying smartphone-acquired dermoscopic images (melanoma, nevus, seborrheic keratosis) from the PAD-UFES-20-Modified, HAM10000, and ISIC2017 datasets. Using augmentation, standardized 224×224 preprocessing, and Adam optimization, SkinLesNet achieved up to 96% accuracy, showing strong potential for real-world dermatological triage. Despite its simplicity and high accuracy, its limitations include reliance on small, non-diverse cohorts and lack of explainability—motivating future directions such as Grad-CAM–based interpretability, self-supervised learning, and scalable data collection. Collectively, these studies trace the evolution of CNNs from large-scale generic vision systems to specialized, lightweight, and clinically adaptable models for healthcare applications.

Transformers

Vision Transformers (ViTs) mark a fundamental departure from convolution-based architectures by treating images as sequences of patches and modeling global dependencies through self-attention rather than local receptive fields. Touvron et al. (2021)[22] introduced DeiT, a data-efficient Vision Transformer trained solely on ImageNet-1k without external pretraining by employing a carefully designed augmentation and regularization pipeline (RandAugment, Mixup/CutMix, Random Erasing, stochastic depth, repeated augmentation, EMA) and a novel distillation token that learns directly from a convolutional teacher (e.g., RegNetY-16GF). DeiT-B achieved 84.5% top-1 accuracy (384 px), demonstrating that ViTs can rival state-of-the-art CNNs with standard data budgets, though its sensitivity to recipe ablations and limited robustness under long-tail distributions remain concerns. Extending transformer utility to medical domains, Halder et al. (2024)[43] fine-tuned a ViT-Base/16 model (224×224, ImageNet-pretrained) on subsets of MedMNISTv2 BloodMNIST, BreastMNIST, PathMNIST, RetinaMNIST achieving accuracies of 97.9%, 90.38%, 94.62%, and 57.0% respectively, outperforming CNN baselines and illustrating that ViTs can address CNNs' locality bias while generalizing across imaging modalities. However, challenges such as small-scale datasets, class imbalance, and domain sensitivity remain, suggesting the need for federated and self-supervised learning to enhance clinical robustness. Similarly, Ji et al. (2025)[48] proposed ScaleFormer, a grouped multi-scale Vision Transformer integrating a Local Per-

ception Unit (LPU) and Grouped Multi-Scale Attention (GMSA) with Inter-Scale Attention (ISA) to fuse local and global contextual features efficiently. Evaluated on Synapse (CT), ACDC (MRI), and ISIC2018 (dermoscopy), ScaleFormer achieved superior Dice, HD95, and mIoU scores compared to U-Net, TransUNet, and Swin-UNet, validating the benefit of grouped multi-scale modeling, though its primarily 2D scope and limited domain-shift testing highlight future opportunities in 3D extensions and uncertainty-aware interpretability. Complementing these architectural advances, Komorowski et al. (CVPRW 2023)[35] examined explainability in ViTs for COVID-19 classification (COVID-QU-Ex dataset, 33,920 images) by comparing LIME, attention rollout, and TransLRP attribution methods, where TransLRP achieved the highest faithfulness (0.16 ± 0.18) and lowest sensitivity (0.06 ± 0.03), establishing a quantitative framework for ViT explainability despite its single-modality limitation and absence of radiologist-verified masks. Together, these studies establish Transformers as a transformative paradigm that combines data efficiency, cross-domain adaptability, multi-scale contextual reasoning, and growing interpretability—positioning them as the next frontier for reliable and explainable medical imaging intelligence.

Hybrid Models

Hybrid models have recently emerged as an effective middle ground between CNNs and Transformers, leveraging the local inductive biases of convolutions and the global dependency modeling of attention or state-space mechanisms. Tang et al. (CVPR 2022)[30] proposed a self-supervised volumetric transformer framework using a 3D Swin UNETR encoder–decoder architecture pre-trained on 5,050 multi-organ CT volumes via anatomy-oriented proxy tasks—masked volume inpainting, 3D rotation prediction, and contrastive learning on region-of-interest (ROI) body parts. This strategy aligns proxy objectives with radiologic structures, enabling superior transfer learning performance on BTCV (13 abdominal organs) and MSD segmentation tasks compared to ImageNet-style or purely supervised pretraining. Despite its scalability and open-source reproducibility (MONAI), its reliance on CT-only pretraining, fixed proxy task weighting, and high 3D computational cost remain open challenges. Building upon the hybridization concept, Manzari et al. (2023)[37] introduced MedViT, a CNN–Transformer hybrid optimized for the 12 MedMNIST-2D benchmarks, employing Efficient Convolution Blocks (ECB) for multi-head convolutional attention, depthwise Local Feed-Forward Networks (LFFN) for locality, and Efficient Self-Attention (ESA) for global modeling. With the addition of a Patch Momentum Changer (PMC) to stabilize training, MedViT achieved AUC values around 0.94 across modalities and exhibited robust adversarial resistance on OctMNIST, TissueMNIST, and RetinaMNIST. While demonstrating strong cross-modality generalization and transparent protocols, its evaluation remains constrained to 2D data and limited domain-shift robustness. Hu et al. (2024)[44] advanced this integration by presenting MambaConvT, a three-branch hybrid model combining a CNN pathway for local texture extraction, a state-space module (SS2D) for long-range dependency modeling, and a Transformer stage for global aggregation. Trained under a standardized protocol on CPN-CX, Brain Tumor MRI, ISIC2018, and Kvasir datasets, MambaConvT consistently outperformed CNNs (ResNet, ConvNeXt, VGG19) and Transformers (ViT-B, Swin-T) across all metrics, with ablations confirming the

complementary contributions of each branch. Nonetheless, most experiments remain 2D, with limited efficiency analysis and robustness assessment, prompting future extensions toward 3D/temporal modeling, self-supervised pretraining, and uncertainty-aware interpretability. Yue and Li (2024)[46] further evolved this trend through MedMamba, replacing self-attention with state-space models (SSMs) that preserve convolutional locality while enabling linear-time sequence modeling. Its four-stage SS–Conv–SSM hierarchy fuses convolutional and sequential dynamics, achieving superior OA/AUC on multi-modal datasets—85.6%/0.953 (Cervical-US), 95.3%/0.997 (PathMNIST), and 97.1%/0.995 (CPN X-ray)—while maintaining low computational overhead. MedMamba's strengths include high scalability, robust cross-modality generalization, and public reproducibility, though its focus remains 2D and requires broader domain-shift and corruption testing. Lastly, Rafiq et al. (2023)[40] demonstrated the utility of fusion-based CNNs for breast cancer histopathology classification (PatchCamelyon dataset), where progressively deeper multi-branch CNNs (1-CNN, 2-CNN, 3-CNN) achieved 90–98% accuracy, showing that parallel convolutional feature fusion enhances discrimination. Despite strong performance, dependence on augmentation and lack of interpretability restricts generalization, motivating the inclusion of explainable AI (Grad-CAM) and domain-agnostic testing. Collectively, these studies illustrate the growing dominance of hybrid paradigms—uniting convolutional, transformer, and state-space mechanisms—to achieve efficient, robust, and generalizable modeling across diverse medical imaging modalities.

Ref.	**Dataset**	**Models**	**Evaluation Metrics**	**Comments**
CNN-Based Models				
[2]	ImageNet ILSVRC	AlexNet (8-layer CNN)	Top-1: 37.5%, Top-5: 17%	High compute; large dataset
[15]	Chest X-ray (5k)	ORB+SVM, VGG16, IncepV3, CapsNet	VGG16 TL best	Small, unbalanced; no cross-site
[32]	PAD-UFES-20, HAM10000, ISIC2017	SkinLesNet (4-layer CNN)	Acc: 90–96%	Small sets; low diversity; no XAI
Transformer-Based Models				
[22]	ImageNet-1k	DeiT (Vision Transformer)	Top-1: 81.8–84.5%	Heavy aug; sensitive setup
[35]	COVID-QU-Ex (33k CXR)	ViT-Base + XAI	Acc: 95.4%	Single set; limited explainability
[43]	MedMNISTv2 (Blood, Breast, Path, Retina)	ViT-Base/16 (pretrained)	Acc: 57–98%	Weak on Retina; small-scale

Ref.	Dataset	Models	Evaluation Metrics	Comments
[48]	Synapse (CT), ACDC (MRI), ISIC2018	ScaleFormer (multi-scale ViT)	Dice/mIoU competitive	Mostly 2D; limited shift
Hybrid Models				
[30]	BTCV, MSD (5k CT vols)	Swin UNETR (3D ViT+CNN)	Dice: 0.918; MSD: 78.7%	CT-only pretrain; high 3D cost
[40]	PCam (Histopathology)	Fusion CNNs (1–3 branches)	Acc: 90–98%	Heavy aug; dataset-specific
[37]	MedMNIST-2D (12 sets)	MedViT (CNN–ViT hybrid)	Acc $\approx$ 0.85; AUC $\approx$ 0.94	2D only; limited shift
[46]	16 sets, 10 modalities (411k imgs)	MedMamba (CNN+SSM hybrid)	Acc: 85–97%; AUC: 0.95–0.99	2D only; uneven robustness
[44]	ISIC 2018, Kvasir, CPN-CX, Brain Tumor (+2 private)	MambaConvT	Acc: 97.3%, F1: 97.0%, AUC: 0.976	High training cost; 2D focus; limited speed/param trade-off

Table 2.1: Summary of Related Works on CNN, Transformer, and Hybrid Models for Medical Image Classification.

2.3 Summary of Key Findings

The reviewed literature demonstrates a coherent trajectory from early convolutional architectures to transformer and hybrid paradigms that progressively expand receptive scope, integrate multi-scale reasoning, and incorporate domain-aware or self-supervised learning for medical imaging. While these advances enhance generalization and robustness, persistent challenges remain in data scarcity, imbalance, computational efficiency, and clinical interpretability.

- **Evolution of CNNs:** Early convolutional networks such as AlexNet established scalable deep learning through ReLU activation, dropout, data augmentation, and multi-GPU training, forming the foundation for subsequent medical imaging applications.
- **Transfer and capacity control:** Medical adaptations of CNNs (e.g., VGG16, InceptionV3) achieved reliable performance when model capacity and layer unfreezing were carefully tuned to small, imbalanced datasets, outperforming handcrafted baselines.

- **Fusion and lightweight designs:** Multi-branch CNNs enhanced histopathology classification via parallel feature fusion, and compact dermoscopy models generalized across cohorts, though both remained constrained by dataset specificity and limited explainability.
- **Rise of Transformers:** Data-efficient training strategies such as DeiT showed that vision transformers can rival CNNs under standard supervision, while direct ViT fine-tuning on MedMNIST confirmed their strength in capturing global dependencies but exposed sensitivity to small inputs and class imbalance.
- **3D and multi-scale learning:** Domain-specific self-supervision (Swin-UNETR) and grouped multi-scale attention (ScaleFormer) improved radiologic segmentation by fusing fine detail with holistic context, outperforming conventional pretraining schemes.
- **Explainability studies:** Quantitative analyses of ViT explanations (e.g., COVID-19 CXR) revealed moderate faithfulness and underscored the need for standardized, clinically interpretable XAI evaluation frameworks.
- **Hybrid and state-space models:** Recent architectures such as MedViT, MambaConvT, and MedMamba integrate convolutional locality with transformer or state-space global modeling, achieving efficiency, robustness, and cross-modality adaptability.
- **Overall implication:** Future medical image analysis should pursue lightweight hybrid frameworks that unify local and global representation learning, employ anatomy-aware or self-supervised pretraining, and embed calibration and interpretability to ensure clinically reliable deployment.

Chapter 3

Requirements, Impacts and Constraints

3.1 Final Specifications and Requirements

This thesis develops and rigorously evaluates a compact hybrid CNN-Mamba architecture for medical image classification across various 2D datasets. The system ingests grayscale and RGB images that are standardized to 224×224 pixels and accommodates different label spaces through a unified data pipeline. The final configuration of the model was chosen after iterative ablation studies that varied patch sizes, stem depth, and the number of residual and Mamba blocks. The selection balances accuracy, generalization, and efficiency: the network remains lightweight (on the order of tens of millions of parameters with roughly one to two GFLOPs), while delivering competitive performance against strong CNN/Transformer baselines.
The software stack consists of Python and PyTorch with CUDA acceleration, torchvision utilities for preprocessing, and a standard scientific Python toolkit for metrics, plotting, and logging. The experiments are run on both local device and cloud settings.

For local setup:

- OS: Linux Ubuntu
- CPU: i7-14700K
- GPU: 4080 Super 16GB
- RAM: 64GB DDR5
- SSD: 1 TB
- HDD: 1 TB

For cloud setup:

- Kaggle
- GPU: 2 T4 GPU (used parallel)

Evaluation emphasizes both effectiveness and reliability. Core metrics include overall accuracy and AUC, supplemented by precision, recall, F1, specificity and Cohen's κ. To reduce the risk of over-interpreting single runs, results are averaged over multiple random seeds, and pairwise statistical tests (e.g., Wilcoxon signed-rank) are used to assess whether observed improvements are consistent rather than incidental.

3.2 Societal Impact

The intended societal value of this work is to augment clinical workflows with faster triage and second-opinion support, especially in settings where expert capacity or imaging infrastructure is limited. By building on public, de-identified datasets and emphasizing compact models, the thesis lowers barriers to adoption and education while encouraging reproducible research practices.
At the same time, risks must be acknowledged. Models trained on specific cohorts or devices may not generalize to under-represented populations or new acquisition settings, potentially leading to uneven performance. There is also a risk that automated outputs could be over-trusted if presented without uncertainty or appropriate caveats. The project mitigates these concerns by reporting results across multiple datasets and seeds, explicitly stating limitations, and positioning the system as decision support rather than a replacement for clinical expertise. Where feasible, clinician-in-the-loop usage and external validation are encouraged to maintain accountability.

3.3 Environmental Impact

The environmental impact of this project is primarily related to the computational resources required for model training and evaluation. To minimize this footprint, the entire design process emphasized efficiency over scale. The chosen CNN–Vision Mamba architecture is lightweight, with only tens of millions of parameters and relatively low computational demand compared to very large Transformer models. By fixing the input resolution at 224×224 and avoiding unnecessarily deep or wide model variants, the project reduced the overall energy consumed per training run.
Experiments were conducted in two settings: a high-performance local workstation equipped with an RTX 4080 Super GPU and a Kaggle cloud environment with dual T4 GPUs. While the Kaggle setup uses more GPUs in parallel, this significantly shortens training time, ensuring that total GPU usage remains reasonable. The local machine, with its modern and efficient hardware, further supports energy-conscious experimentation.
Additional steps were taken to keep the impact under control. Mixed-precision training was used where stable, reducing both runtime and power draw. Ablation studies were carefully planned to focus only on meaningful variations, avoiding an excessive number of redundant experiments. A single preprocessing pipeline was reused across all datasets, preventing repeated data conversions and storage overhead. Checkpoint saving was also limited to essential models to reduce storage and I/O energy use.
Overall, by prioritizing compact model design, disciplined experimentation, and hardware-efficient practices, the project achieved its research objectives without

incurring the environmental cost typically associated with large-scale deep learning projects.

3.4 Ethical Issues

Ethical practice in medical AI begins with data stewardship. This work uses only public, de-identified datasets under appropriate licenses and avoids any form of personally identifiable information. Dataset provenance, preprocessing, and splits are documented to minimize the risk of leakage or optimistic bias. Fairness is considered through the lens of distributional shift and class imbalance; known sensitivities—such as differences between grayscale and RGB modalities—are described with guidance on when performance may degrade.
Safety is addressed by clearly stating that the model is a research artifact, not a medical device, and must not be used for autonomous diagnosis or treatment decisions. Interpretability outputs are provided to support human scrutiny but are never presented as definitive explanations. Reproducibility and transparency are emphasized through open configurations, seeds, and saved models, enabling independent verification and facilitating responsible downstream use.

3.5 Standards - if applicable

While formal clinical device standards are out of scope, the thesis aligns with community norms for trustworthy medical-imaging research. Data and code practices follow FAIR principles where feasible; experiments use consistent input resolution, clearly defined train/val/test splits, and a standard metric suite. Statistical robustness is addressed through multi-seed evaluation and paired non-parametric testing. Concise "model card" style documentation accompanies the final configuration to communicate intended use, limitations, and potential failure modes. Together, these practices mirror the spirit of emerging ML transparency guidelines even in the absence of regulatory certification.

3.6 Project Management Plan

The work proceeds in phased iterations. First, datasets are curated and standardized to 224×224 with verified labels and deterministic splits. Next, strong baselines are trained to establish reference performance and computational costs. Building on this foundation, the hybrid architecture is implemented with profiling hooks to measure parameters, GMACs, throughput, and latency. Systematic ablations then explore the design space—varying patch sizes, stem depth, and the number or placement of residual/Mamba blocks—until a small set of finalists emerges.
With candidates selected, multi-dataset benchmarking quantifies accuracy and AUC across modalities, followed by robustness analysis using multiple seeds and statistical tests. Interpretability studies examine representative successes and errors to provide qualitative insight. After that, explainability is also checked to validate the model's interpretability. Finally, cross-dataset evaluations probe generalization by training on one dataset and testing on a related but distinct dataset. The last stage

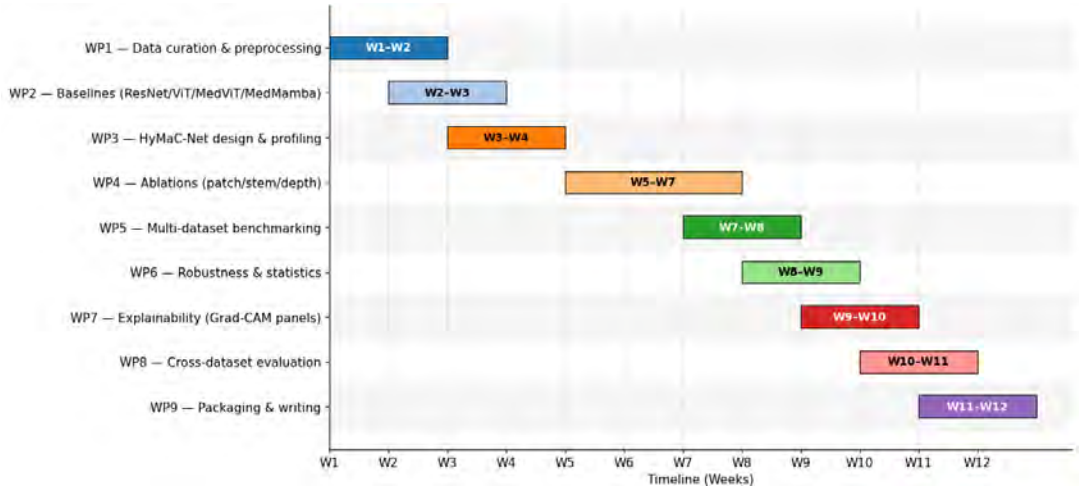


Figure 3.1: Gantt chart illustrating the project schedule and milestones.

consolidates results, artifacts, and narrative into the thesis chapter and an accompanying reproducibility bundle. Decision gates after the ablation and benchmarking phases ensure that only configurations with the best accuracy-to-efficiency trade-offs proceed to comprehensive testing. All of these stages are depicted in Fig. 3.1 for visual representations.

3.7 Risk Management

Several risks are associated with the development and evaluation of a medical imaging model across multiple datasets, and each has been carefully considered in the design of this project. One of the major risks lies in dataset bias and distributional shift. Public datasets often differ in collection methods, imaging devices, and population demographics, which can affect how well a model generalizes. To address this, the project evaluates performance across diverse datasets and also performs cross-dataset testing to reveal potential weaknesses and document limitations.
Another potential risk is related to computational constraints. Training deep learning models can be resource-intensive, and limited GPU memory or excessive training time may disrupt experiments. This was mitigated by deliberately designing a compact architecture that remains within practical bounds of GPU capacity, by applying early downsampling, and by using mixed-precision training where possible.
Finally, there are risks linked to reproducibility and interpretability. Without clear documentation, reproducing results may be difficult. Similarly, misinterpretation of model explanations, such as saliency maps, could lead to misleading conclusions. To mitigate these, the project maintains strict versioning of code and configurations, releases seeds and checkpoints, and clearly frames interpretability outputs as qualitative insights rather than definitive explanations.

3.8 Economic Analysis

Economic costs were primarily related to GPU usage, storage, and researcher time. By adopting a lightweight CNN–Vision Mamba design, the project reduced training

complexity and energy demand. Experiments ran efficiently on both a local workstation (RTX 4080 Super) and free Kaggle GPUs, which minimized financial costs. Storage use was also optimized by standardizing preprocessing pipelines, caching datasets, and saving only essential checkpoints. These measures ensured the project remained cost-effective, achieving robust results without the high expenses often associated with large-scale deep learning research.

Chapter 4

Proposed Methodology

4.1 Methodology Overview

Our proposed methodology has been designed to build a generalizable and interpretable deep learning architecture for medical image classification that can perform reliably across different datasets and imaging modalities. Our approach, named HyMaC-Net, follows a structured pipeline that begins with careful data preparation, progresses through architecture design and ablation-driven optimization, and finally incorporates explainable AI to validate the decision-making process.

At the initial stage, multiple datasets were considered to ensure the generalization ability of the proposed framework. The primary dataset used was PathMNIST, a colorectal cancer dataset with nine classes, complemented by additional datasets from the MedMNIST benchmark suite such as BloodMNIST, OrganAMNIST, DermaMNIST, and others. Since these datasets vary in terms of modality, image resolution, and number of channels (RGB vs. grayscale), we applied a uniform preprocessing strategy involving resizing, normalization, and in some cases augmentation. This step was crucial for maintaining consistency across datasets and allowing cross-dataset evaluation later in the study.

The second stage of the methodology focused on the design of our proposed HyMaC-Net. The initial architecture was built upon a CNN stem for extracting low-level spatial features, which were then passed into Mamba blocks for advanced sequence modeling and long-range dependency learning. To improve efficiency and accuracy, we carried out patch embedding optimization, experimenting with patch sizes of 28, 14, 7, and 4. This process revealed that smaller patch sizes improve classification performance but at the cost of increased computational complexity. To overcome this trade-off, we modified the CNN stem to perform additional downsampling, thereby reducing FLOPs while maintaining high accuracy.

To further refine the architecture, we conducted ablation studies by systematically adding, removing, or modifying components such as residual (RE) blocks, Atrous Spatial Pyramid Pooling (ASPP), dropout layers, and pooling strategies. These experiments revealed that removing certain residual connections not only reduced architecture complexity but also stabilized accuracy. Through this process, lightweight yet high-performing variants were identified with Strategy 20 and 21 achieving the best balance between accuracy, architecture size, and inference time. After that we have also tried replacing mamba-ssm blocks with linformer, though we did not achieve better results from it that's why we did not continue with this stage. The

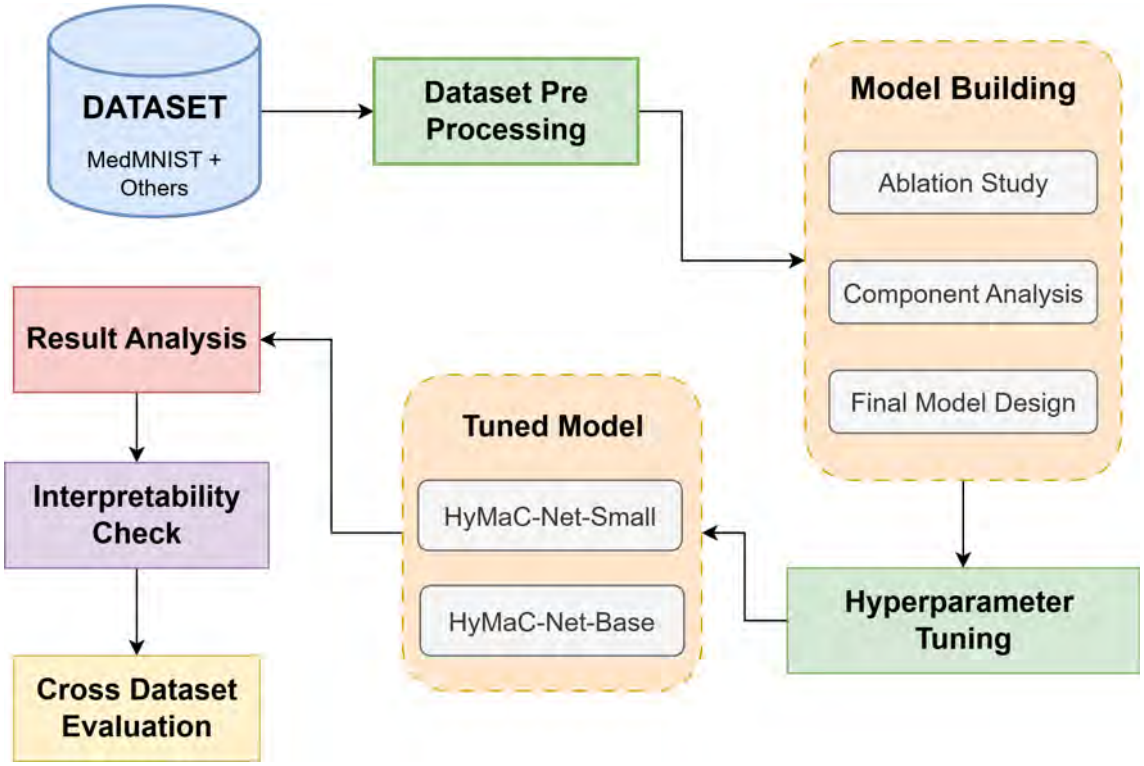


Figure 4.1: Methodology diagram of the overall workflow of our thesis; starting from the dataset and ending at cross-dataset evaluation testing.

third stage of the methodology involved training and evaluation of the proposed architecture and its variants. To ensure robustness, our architecture were trained using multiple random seeds, and their performance was averaged and compared with ResNet-50. Proposed architecture design evaluation is carried out using several key metrics, including Overall Accuracy (OA), Area Under the Curve (AUC), precision, recall, F1 score, inference time, and architecture size. The results were compared against strong baselines such as ResNet-50, Vision Transformer (ViT), MedViT, MedMamba. This benchmarking demonstrated that HyMaC-Net consistently outperformed or matched these state-of-the-art architectures while remaining computationally efficient.

In the final stage, we addressed the critical aspect of explainability. Since medical imaging requires interpretability to gain the trust of clinicians, we integrated Grad-CAM visualizations to highlight the regions of input images that influenced the architecture's predictions. These heatmaps confirmed that the proposed architecture focuses on clinically relevant areas, thereby validating both the reliability and the interpretability of HyMaC-Net.

Overall, our methodology integrates data preprocessing, hybrid architecture design, ablation-based refinement, rigorous evaluation, and explainable AI into a unified framework presented in the Fig. 4.1. This structured approach not only ensures high classification accuracy but also demonstrates generalization across multiple datasets and provides interpretability, which is crucial for real-world medical applications.

4.2 Dataset

4.2.1 MedMNIST V2

To evaluate the generalization capability of our proposed HyMaC-Net architecture, we employed the MedMNIST V2 [42] benchmark suite, which extends the MedMNIST collection. This benchmark includes twelve 2D biomedical image datasets that cover a broad spectrum of modalities, such as histopathology, chest X-rays, dermatoscopy, optical coherence tomography (OCT), ultrasound, fundus imaging, blood cell microscopy, and abdominal CT scans. Each dataset poses unique challenges by varying in image modality (RGB vs. grayscale), classification type (binary, multi-class or ordinal regression), and dataset scale (from hundreds to hundreds of thousands of images).

DermaMNIST

DermaMNIST is built on the HAM10000 [11],[10] dataset, which contains 10,015 dermatoscopic images of pigmented skin lesions. It poses a 7-class classification task, including melanoma, basal cell carcinoma, benign keratosis, dermatofibroma, and vascular lesions. Images are 224×224 RGB, making color and texture critical features for diagnosis. This dataset specifically tests the HyMaCNet's ability to extract color- and texture-based features, which are essential in dermatological analysis.

PathMNIST

PathMNIST is derived from NCT-CRC-HE-100K and CRC-VAL-HE-7K and aggregates colorectal H&E pathology tiles into 9 tissue classes (e.g., normal mucosa, stroma, tumor epithelium, necrosis, lymphocytes, adipose) [7]. The total set contains 107,180 images standardized in MedMNIST, while we use 224×224 RGB inputs to preserve nuclei and glandular textures. The task probes fine-grained histological morphology and color variations typical of H&E slides. The PathMNIST dataset was selected as the primary benchmark for developing HyMaC-Net through ablation and component study. It was chosen as the primary dataset because of its clinical relevance, large scale, and multi-class nature, which makes it ideal for testing both the accuracy and generalization ability of the proposed architecture.

OCTMNIST

OCTMNIST is built from a large retinal OCT collection [8],[9] of 109,309 B-scans and frames a 4-class diagnosis: choroidal neovascularization, diabetic macular edema, drusen, and normal. We use 224×224 inputs (grayscale content) to maintain layer boundaries and fluid pockets. The dataset stresses sensitivity to small structural disruptions in the retinal cross-section.

PneumoniaMNIST

PneumoniaMNIST comes from a pediatric chest X-ray cohort with 5,856 images and is formulated as binary classification (pneumonia vs. normal) [8],[9]. Images are

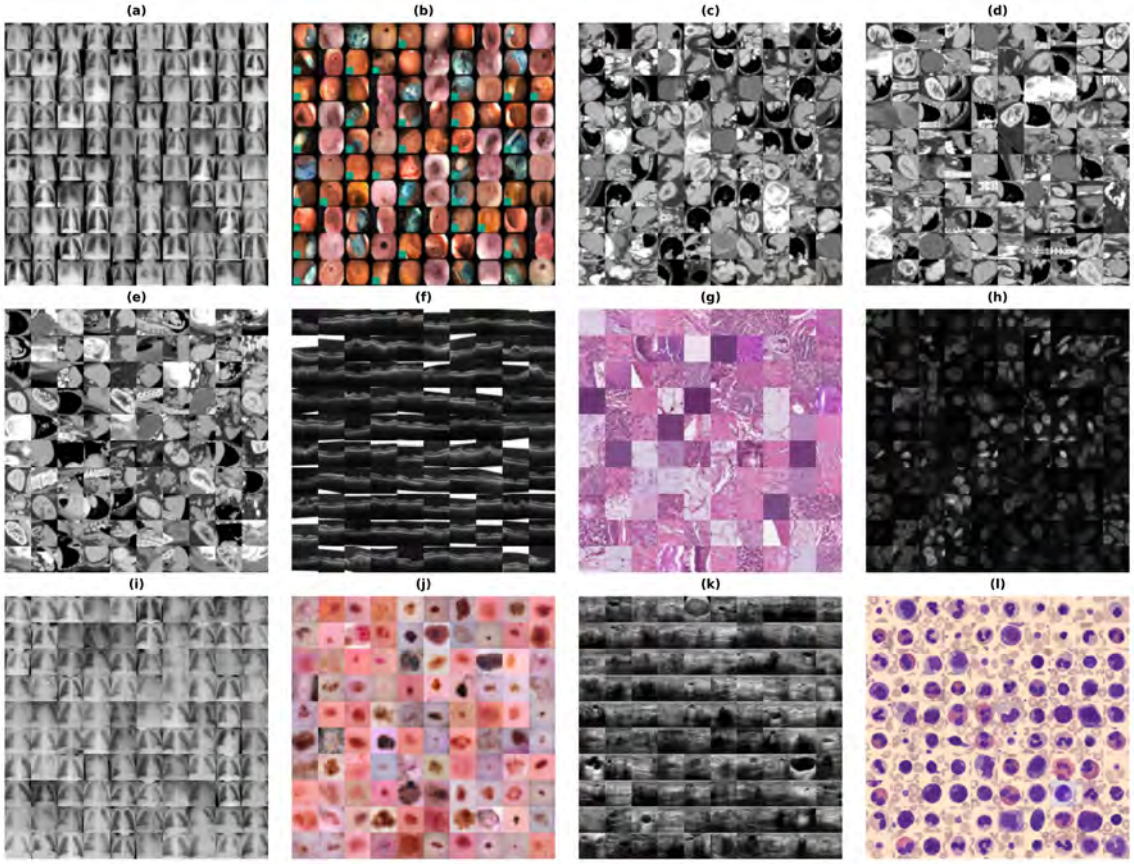


Figure 4.2: Typical samples of Used 12 different datasets with different imaging modalities; (a) CPN X-ray, (b) Kvasir, (c) OrganAMNIST, (d) OrganCMNIST, (e) OrganSMNIST, (f) OCTMNIST, (g) PathMNIST, (h) TissueMNIST, (i) PneumoniaMNIST, (j) DermaMNIST, (k) BreastMNIST, (l) BloodMNIST.

handled as 224×224 grayscale in our pipeline. It is useful for assessing performance and overfitting behavior on a smaller, clinically focused two-class task.

BreastMNIST

BreastMNIST is constructed from a 780-image breast ultrasound set (normal, benign, malignant) [17] and is simplified in MedMNIST to binary classification by grouping normal + benign vs. malignant. We treat images as 224×224 grayscale, which captures typical ultrasound artifacts like speckle and low contrast. The dataset probes noise-robust feature extraction under limited data.

BloodMNIST

BloodMNIST contains 17,092 microscopy images of peripheral blood cells divided into 8 cell-type classes [16]. We use 224×224 RGB inputs to retain staining and morphological cues such as nucleus shape and cytoplasm texture. This dataset evaluates fine-grained cellular recognition where inter-class differences are subtle.

TissueMNIST

TissueMNIST is sourced from BBBC051 [25] (human kidney cortex cells) and includes 236,386 instances grouped into 8 categories via 2D projections. We process them as 224×224 grayscale to standardize across datasets. It stresses scalability on large, single-channel bioimage sets with relatively homogeneous textures.

Organ[A,C,S]MNIST

The Organ[A,C,S]MNIST datasets are derived from the LiTS abdominal CT collection [33] and represent three orthogonal anatomical planes: axial (A), coronal (C), and sagittal (S). Each variant contains 2D slices labeled across 11 abdominal organs, forming a comprehensive multi-view benchmark for organ recognition in medical imaging. All inputs are standardized to 224×224 grayscale resolution. OrganAMNIST evaluates axial organ localization, OrganCMNIST tests cross-plane generalization in the coronal view, and OrganSMNIST complements them with sagittal slices.Together, they assess the robustness and feature consistency of machine learning and deep learning architectures across different anatomical orientations.

4.2.2 Other Datasets

Besides MedMNIST V2, we have also validated our proposed architecture on two additional publicly available medical imaging datasets CPN X-ray [28] and Kvasir [6]. These datasets extend the evaluation beyond the MedMNIST suite, enabling assessment of the SOTA architecture's generalization across different imaging modalities, including chest X-ray and gastrointestinal endoscopy. Both serve as complementary benchmarks for analyzing performance consistency under diverse visual and diagnostic conditions.

CPN X-ray

CPN X-ray is a tri-class chest-X-ray collection with 5,228 images across COVID-19 (1,626), Normal (1,802), and Pneumonia (1,800) [28]. We standardize to 224×224 grayscale. It serves as a simpler 3-way counterpart to ChestMNIST's multi-label task and supports evaluation of respiratory screening under balanced class counts.

Kvasir

Kvasir is a GI endoscopy dataset [6] with 8 classes covering anatomical landmarks and common findings: Z-line, pylorus, cecum, esophagitis, polyps, dyed & lifted polyp, dyed resection margins, and ulcerative colitis. The original Kvasir classification release contains 4,000 images (500 per class); a larger Kvasir-v2 (8,000 images) also appears in the literature—our experiments follow the classic 8-class configuration. We use 224×224 RGB to capture mucosal color/texture and illumination changes.

In Table 4.1, we can see the detailed description of the 12 datasets and Fig. 4.2 visually illustrates representative samples from the twelve medical imaging datasets used in this study, highlighting the diversity of modalities, textures, and anatomical structures.

Figure 4.3: Class Distribution of 12 used Dataset; (a) CPN X-ray, (b) Kvasir, (c) OrganAMNIST, (d) OrganCMNIST, (e) OrganSMNIST, (f) OCTMNIST, (g) PathMNIST, (h) TissueMNIST, (i) PneumoniaMNIST, (j) DermaMNIST, (k) BreastMNIST, (l) BloodMNIST.

Name	Imaging Modality	Task (Classes)	Dataset Size	Training / Validation / Test	Data Availability
PathMNIST	Colon Pathology (H&E)	MC (9)	107,180	89,996 / 10,004 / 7,180	Public
DermaMNIST	Dermatoscope	MC (7)	10,015	7,007 / 1,003 / 2,005	Public
OCTMNIST	Retinal OCT	MC (4)	109,309	97,477 / 10,832 / 1,000	Public
PneumoniaMNIST	Chest X-Ray	BC (2)	5,856	4,708 / 524 / 624	Public
BreastMNIST	Breast Ultrasound	BC (2)	780	546 / 78 / 156	Public
BloodMNIST	Blood Cell Microscope	MC (8)	17,092	11,959 / 1,712 / 3,421	Public
OrganAMNIST	Abdominal CT (axial)	MC (11)	58,850	34,581 / 6,491 / 17,778	Public
OrganCMNIST	Abdominal CT (coronal)	MC (11)	23,660	13,000 / 2,392 / 8,268	Public
OrganSMNIST	Abdominal CT (sagittal)	MC (11)	25,221	13,940 / 2,452 / 8,829	Public
TissueMNIST	Human Kidney Cortex Cells (BBBC051)	MC (8)	236,386	165,470 / 35,458 / 35,458	Public
CPN X-ray	Chest X-Ray (COVID/Normal/Pneumonia)	MC (3)	5,228	3,660 / 784 / 784	Public
Kvasir	Gastrointestinal Endoscope	MC (8)	4,000	2,800 / 600 / 600	Public

Table 4.1: Detailed descriptions of the 12 datasets.

Cross-Dataset Evaluation Dataset

To evaluate the generalization ability of HyMaC-Net, a cross-dataset evaluation was conducted. In this setup, the HyMaCNet is trained on the BRISC-25 [47] and tested on the Brain Tumor dataset from Kaggle, both representing the same disease domain but collected under different imaging conditions.

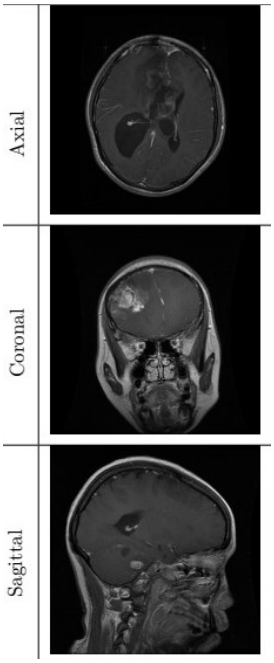


Figure 4.4: Representation of the Planes of BRISC-25 and Brain Tumor Dataset.

The BRISC-25 dataset includes 6,000 T1-weighted MRI images with the same four diagnostic classes, along with physician-validated segmentation masks and coverage of three anatomical planes: axial, coronal, and sagittal showed in Fig. 4.4. It provides a clean and stratified split of 5,000 training and 1,000 testing samples, of which only the train portion was used in this evaluation.
The Brain Tumor dataset contains 7,023 MRI images across four categories glioma, meningioma, pituitary tumor, and no tumor compiled from multiple sources such as Figshare, SARTAJ, and Br35H. After cleaning and augmentation, the dataset comprised 7,870 training, 41 validation, and 353 testing samples.
This cross-dataset design was intended to examine whether HyMaC-Net can transfer learned representations across datasets acquired under distinct clinical and technical

conditions, thereby validating its robustness and domain generalization capability.

4.2.3 Dataset Preprocessing

Data preprocessing is a crucial step in preparing biomedical image datasets for deep learning, as it ensures consistency across inputs, enhances feature quality, and improves architecture generalization. Since the datasets used in this study vary in modality (RGB vs. grayscale), resolution, and class distribution, a carefully designed preprocessing pipeline was implemented.

For the MedMNIST V2 datasets, all images were imported directly at 224×224 resolution. This eliminated the need for manual resizing, since the dataset API provides a standardized input size compatible with our proposed architecture. The preprocessing for these datasets mainly included conversion to tensors and normalization based on pre-computed mean and standard deviation values. Normalization was applied channel-wise so that the architectures could train effectively without being biased by intensity variations across modalities.

For the cross-dataset evaluation experiments, additional preprocessing steps were required to handle the diversity of external datasets such as BRISC-25 and Brain Tumor MRI. In this case, all images were resized to 224×224 to match the HyMaC-Net input specification. To improve robustness against domain shifts, we employed data augmentation techniques during training. These included: Random horizontal flipping, Color jittering with slight brightness and contrast adjustments, and Normalization using dataset-specific mean and standard deviation values (calculated from the training split). These settings are only applied for Cross-dataset evaluation dataset not on the other datasets we have used.

For each cross-dataset, dataset statistics (mean and standard deviation) were first computed from the training set without normalization. These values were then used for channel-wise standardization, ensuring that input images followed a normalized intensity distribution tailored to the dataset rather than applying generic ImageNet values. This approach reduced the risk of overfitting to intensity patterns not relevant to disease classification.

MedMNIST V2 offers all datasets already divided into train, validation, and test sets. We explicitly split all other datasets into train, validation, and test sets in a 70:15:15 ratio.

To address the class imbalance observed across several datasets (as shown in Fig. 4.3), we employed a multi-strategy mitigation approach. First, class-weighted loss functions were computed by inversely weighting each class proportional to its frequency in the training set, ensuring that minority classes received stronger gradient signals during backpropagation. Second, for datasets with severe imbalance (e.g., CPN X-ray), we replaced standard cross-entropy with Focal Loss ($\gamma = 2.0$), which down-weights the contribution of well-classified examples and focuses learning on hard, misclassified cases—particularly effective for rare pathological classes. Finally, label smoothing ($\epsilon = 0.1$) was applied universally to prevent overconfident predictions on majority classes. This combination of loss-based techniques proved more effective than synthetic oversampling or aggressive augmentation, which can introduce unrealistic samples or overfit to augmented patterns in medical imaging contexts where data fidelity is critical.

Finally, for MedMNIST V2, class imbalance was addressed by calculating class

Model (from scratch)	Params	OA (%)	AUC (%)
ResNet-50	23.5 M	91.2	98
ViT-Base-16	85.8 M	88.2	97
ViT-Base-32	87.4 M	79.4	89

Table 4.2: Comparing results of ResNet50 and ViT-Base16 and 32 on PathMNIST dataset.

weights from the training distribution and applying them in the loss function. This adjustment allowed the HyMaC-NeT to better handle underrepresented classes, which is especially important in medical datasets where rare pathologies often carry greater clinical importance.

4.3 Design (Architecture) Specification

4.3.1 Ablation Study

We began with an initial baseline architecture (Fig. 4.5) which has 5 key components in it: firstly CNN Stem, secondly Multi-scale Patch Embedding, thirdly Vision Mamba encoder which has total 12 Mamba blocks: 6 for fine scale and 6 for coarse scale, fourthly Pooling fusion, fifth Gated Feature Fusion and finally the classifier head. Our primary design architecture yield in parameters of ∼138M and GFLOPs: 3.24. In this phase we have designed ablations to tease apart the contribution (the initial model architecture) of four levers: (A) tokenization resolution and stem downsampling, (B) residual/SE scaffolding around tokenization, (C) sequence-mixer depth, and (D) regularization, pooling, and auxiliary supervision. For each setting we tracked accuracy (OA), AUC-ROC, macro-F1, computational cost (GMACs), parameters (M), per-sample latency (ms) and for dataset of the whole ablation process we have choosen the PathMNIST[7] dataset.

Before starting the thorough ablation procedure we have created a foundational benchmark on ResNet50, ViT-B-16, ViT-B-32 on the PathMNIST dataset and the results are shown in the Table 4.2.

Phase A: Tokenization & stem resolution: We compared (i) no tokenization vs. (ii) Conv2d patch-embedding at different patch sizes, while controlling the stem output grid. (Table 4.3) In HyMaC-Net, the stem compresses 224×224 to $28\times28\times64$ via Conv $7\times7/2 \rightarrow$ AvgPool$/2 \rightarrow$ Conv $3\times3/2 \rightarrow$ Conv $3\times3/1$. We then apply `Conv2d(kernel=stride=4)` to produce a 7×7 token grid (49 tokens) with embedding $D \in \{512, 768\}$. Moving from "no patching" to patching systematically improved OA/AUC and reduced parameter count (the projection replaces wider post-stem heads), while compute per token increases as tokens get finer. Fixing the stem at 28×28 amortizes this cost: `patch=4` yields a compact token set ($7 \times 7 = 49$) that preserves locality while bounding GMACs. A downsampled, texture-rich stem plus small, grid-structured tokens strikes an effective accuracy–efficiency balance: enough tokens for global context, not so many that mixer depth becomes expensive. Use 28×28 stem + `patch=4` as the default for both B ($D = 768$) and T ($D = 512$)

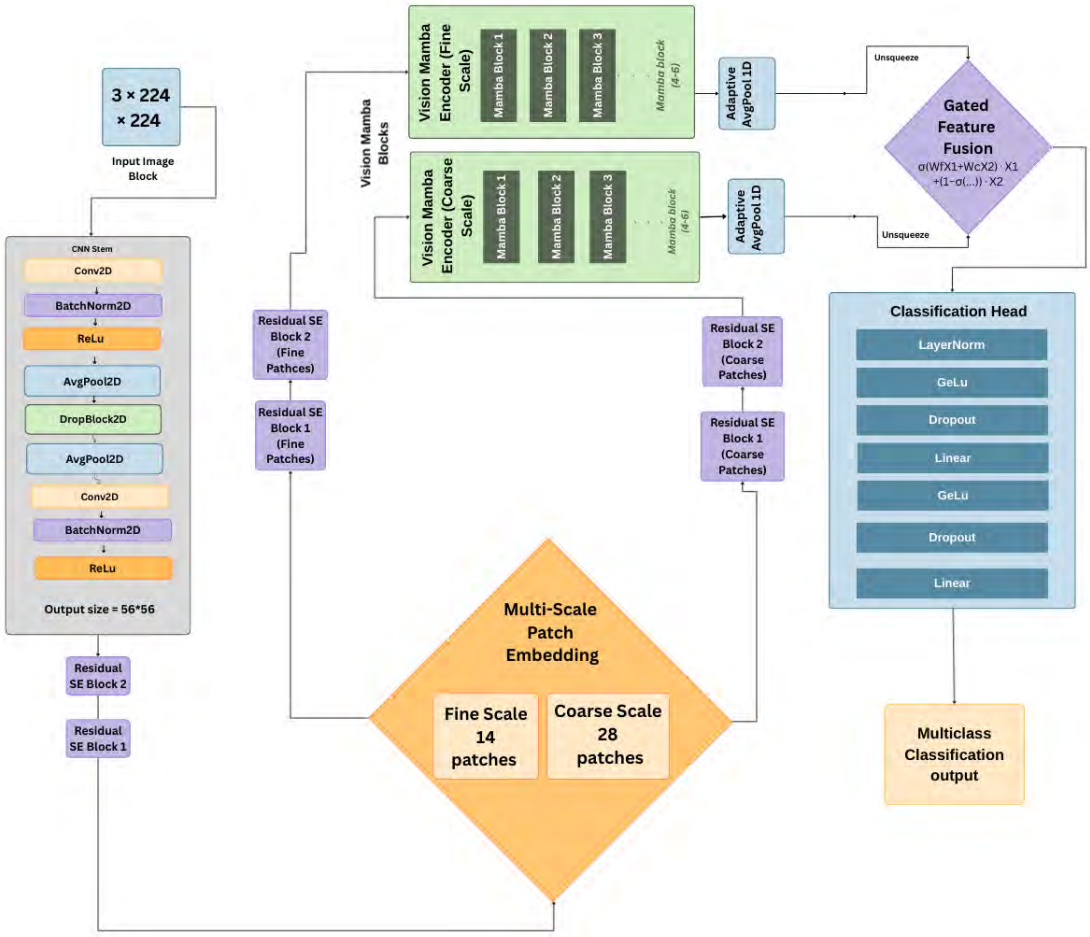


Figure 4.5: Overview of Initial Architecture

variants.

Strategy	Description	Params (M)	OA (%)	AUC	GMACs	Architecture Size (MB)	Train/epoch	Inference
1	Initial architecture	138.0	89.00	99.70	1.62	599.00	∼800 s	1.64 s
2	Initial-12 (12 Mamba), **patch removed**	7.0	87.00	97.60	2.30	273.00	∼866 s	2.97 s
3	Initial-12, **patch=28**	126.3	88.81	97.78	0.84	544.00	∼620 s	1.32 s
4	Initial-12, **patch=14**	97.4	88.30	98.55	1.89	440.00	∼535 s	1.60 s
5	Initial-12, **patch=7**	90.0	90.52	98.87	6.09	454.00	∼935 s	3.00 s

Table 4.3: Phase A: Tokenization vs. Complexity (baseline → single-patch).

Phase B: Residual/SE scaffolding around tokenization: We experimented with Residual-SE blocks (pre/post patching) as lightweight contextual adapters. (Table 4.4) Our final architecture keeps the ResidualSEBlock module available but does not include it in the forward graph. With an informative stem and learnable 2-D positional embeddings, extra SE-scaffolding around patching led to diminishing returns relative to its added memory/compute and architectural complexity. Gains that SE can offer early in training were later matched—and sometimes exceeded—by (i) better stem downsampling, (ii) the Mamba mixer, and (iii) improved pooling/supervision. Once tokens carry stable spatial bias (via the stem) and positional encoding, the selective state-space mixer already supplies sufficient gating and channel-wise modulation; explicit SE blocks become redundant. Remove Residual-SE blocks from the final path (retain as an optional module for future variants).

Strategy	Description	Params (M)	OA (%)	AUC	GMACs	Architecture Size (MB)	Train/epoch	Inference
6	Changed stem (more downsample), patch=4	88.6	89.00	97.95	4.53	418.57	∼725 s	2.34 ms
7	Initial-12, remove 1 RE before (patch=4)	88.5	91.78	98.30	4.51	415.63	∼628 s	1.89 ms
8	Initial-12, remove 1 RE after (patch=4)	83.8	92.15	98.68	4.30	397.26	∼683 s	2.00 ms
9	Initial-12, remove RE before & after	83.7	87.09	97.50	4.27	397.31	∼620 s	1.84 ms
10	Initial-12, remove 2 RE after	79.0	90.99	98.38	4.07	375.95	∼610 s	1.90 ms
11	Initial-12, remove 2 RE before	88.0	91.78	98.98	4.48	412.68	∼663 s	1.98 ms
12	Initial-12, remove all RE (before & after)	78.9	92.00	98.73	4.02	370.05	∼597 s	1.95 ms
13	No Res blocks, add ASPP after stem (patch=4)	82.0	91.60	98.45	4.75	398.78	∼755 s	—

Table 4.4: Phase B: Stronger stem (28×28) + RE/SE scaffolding + ASPP.

Phase C: Mixer depth (Mamba-SSM): We reduced VisionMambaBlock depth from deeper stacks to 6, 3, 2, and 1 blocks (each block: LN → Mamba → Dropout + residual; LN → MLP($\times 4$) + Dropout + residual). Reducing depth sharply cuts GMACs and parameters with only small accuracy changes down to 3 blocks; dropping to 2 blocks stayed competitive on some datasets, while 1 block favored efficiency but lost robustness. With 49 tokens and $D = 768$, 3 blocks provided the best stability across datasets. At 7×7 tokens, long-range mixing needs only a shallow selective state-space stack; deeper mixers under-utilize capacity and inflate cost. 3 Mamba blocks (num_layers=3) for HyMaC-Net-B; the S-variant keeps the same depth but uses $D = 512$ for further efficiency. Table 4.5 shows the Phase C ablation.

Strategy	Description	Params (M)	OA (%)	AUC	GMACs	Architecture Size (MB)	Train/epoch	Inference
14	6 Mamba blocks, no Res, patch=4	39.9	91.04	98.13	2.10	194.18	∼328 s	0.93 ms
16	6 blocks, keep 2 RE before, remove 1 RE after	44.8	89.65	98.40	2.39	221.38	∼387 s	1.04 ms
17	3 Mamba blocks, remove all RE	20.4	91.78	99.07	1.15	106.24	∼215 s	0.67 ms
18	2 Mamba blocks, remove all RE	13.9	91.97	98.91	0.83	76.91	∼230 s	0.54 ms
19	1 Mamba block, remove all RE	7.4	89.83	98.45	0.51	47.61	∼192 s	0.37 ms

Table 4.5: Phase C: Mixer depth (Mamba-SSM) ablation.

Phase D: Regularization, pooling fusion, and auxiliary supervision: (i) Dropout tuning ($0.30 \rightarrow 0.15$ within blocks/heads), (ii) fused pooling (element-wise mean of global-avg and global-max over token features), and (iii) a mid-stream auxiliary head (after half the blocks) with a small fusion weight (0.3).(Table 4.6) Moderate dropout (0.15) improved convergence and calibration compared to heavier dropout. Avg+Max fusion reduced variance and boosted minority-class recall on medical subsets where salient patterns are either concentrated (max) or diffuse (avg). The auxiliary head consistently added a small but reliable gain and stabilized training, especially on class-imbalanced datasets (mirrors your optional FocalLoss usage). These changes increase gradient quality and feature diversity without increasing token count or mixer depth—i.e., "free" performance. Keep dropout=0.15, keep avg–max fusion, and keep the auxiliary head with aux_weight=0.3.
The best accuracy–efficiency point arises from: 28×28 stem $\rightarrow$ patch=4 $\rightarrow$ 3 Mamba blocks $\rightarrow$ fused pooling $\rightarrow$ auxiliary head, with SE/Residual scaffolding removed. This configuration preserves accuracy and AUC while keeping GMACs/params modest and inference latency low.

Strategy	Description	Params (M)	OA (%)	AUC	GMACs	Architecture Size (MB)	Train/epoch	Inference
20	S17 + lower dropout ($0.3 \rightarrow 0.1$), fused pooling, seed=42	20.4	92.00	99.14	1.15	106.24	∼184 s	0.67 ms
21	S18 + same regularization/fusion	14.0	92.58	99.06	0.83	77.33	∼178 s	0.55 ms
22	S20 + auxiliary head (final HyMaC-Net)	20.5	94.00	99.05	1.18	107.00	∼190 s	0.65 ms

Table 4.6: Phase D: Regularization, pooling fusion, and auxiliary supervision.

In Table 4.7, we can see that although strategy 21 and strategy 20 were showing competitive results on PathMNIST dataset, strategy 20 worked better on other datasets. That's why strategy 20 were chosen.

4.3.2 Component Analysis

Mamba-SSM vs. Linformer (drop-in mixer) We replaced the Mamba mixer with Linformer in pilot runs (same tokenization and head). Linformer's low-rank attention is attractive for long sequences, yet our sequence length is short (49 tokens). Capacity vs. depth. Linformer needed more blocks to match the stability we obtained from 3 Mamba blocks. Mamba's selective state-space dynamics (content-dependent gating with implicit convolutional context) aligned better with medical textures and subtle radiological cues, yielding more consistent gains under identical training settings. Complexity. With only 49 tokens, the theoretical advantage

Dataset	Description	OA	AUC	Description	OA	AUC
OrganSMNIST	Strategy 21	79.21	98.00	Strategy 20	78.67	97.41
DermaMNIST	same	77.40	93.00	same	77.51	91.00
OCTMNIST	same	83.00	97.56	same	85.90	95.20
PneumoniaMNIST	same	84.23	94.75	same	86.86	95.78
BreastMNIST	same	82.00	84.67	same	81.41	89.10
BloodMNIST	same	92.78	99.81	same	97.52	99.82
OrganAMNIST	same	95.42	99.76	same	95.46	99.74
OrganCMNIST	same	89.23	99.36	same	92.45	99.56
TissueMNIST	same	62.89	92.80	same	68.19	93.00
PathMNIST	same	92.58	99.06	same	92.00	99.14

Table 4.7: Testing Strategy 20 and Strategy 21 on MedMNIST V2 datasets.

Strategy	Description	Params	OA	AUC	GMacs	Model Size	Training/epoch	Inference Time
a	Linformer blocks-12 (16 LF blocks) patch=7	86.2	90.85	98.76	5.63	416.53	∼848 s	2.81
b	Linformer-3 with patch size = 4	37.2	93.00	99.00	1.93	176.29	∼304 s	0.84
c	Linformer-2 with patch size = 4	25.6	92.30	99.00	1.35	123.62	∼216 s	0.65

Table 4.8: Component analysis: Linformer Vs Mamba-SSM.

of Linformer's linearized attention is less impactful; both are efficient, but Mamba achieved a stronger accuracy efficiency point for HyMaC-Net. Conclusion. We have fixed Mamba-SSM as our main component. Linformer remains a viable alternative but offers no net advantage here. Table 4.8 shows the performance analysis of Linformer.

Atrous Spatial Pyramid Pooling (ASPP) We evaluated ASPP after the stem and before patching to inject multi-scale dilation signals. *Pros.* Moderate improvements in early training; helpful on datasets with large scale variability. *Cons.* Nontrivial GMAC/latency overhead and architectural complexity; gains diminished once patching + positional embedding + Mamba were in place. So, ASPP was not retained in the final HyMaC-Net, consistent with our lightweight objective. The stem's progressive downsampling and the Mamba already provide sufficient multi-scale/contextual capacity. The results of ASPP implemented modules are depicted in the Table 4.4 strategy 13.

4.3.3 Proposed Architecture Design

HyMaC-Net is a lightweight hybrid that couples a texture-preserving CNN stem, compact patch embedding with a learned 2-D positional grid, a shallow stack of selective state-space blocks (Mamba-SSM), and a dual-supervision classifier (main head with an auxiliary head). Given an input mini-batch $x \in \mathbb{R}^{B\times\mathcal{C}\times 224\times 224}$ (with $\mathcal{C} \in \{1, 3\}$ depending on the dataset), the CNN stem injects inductive bias and

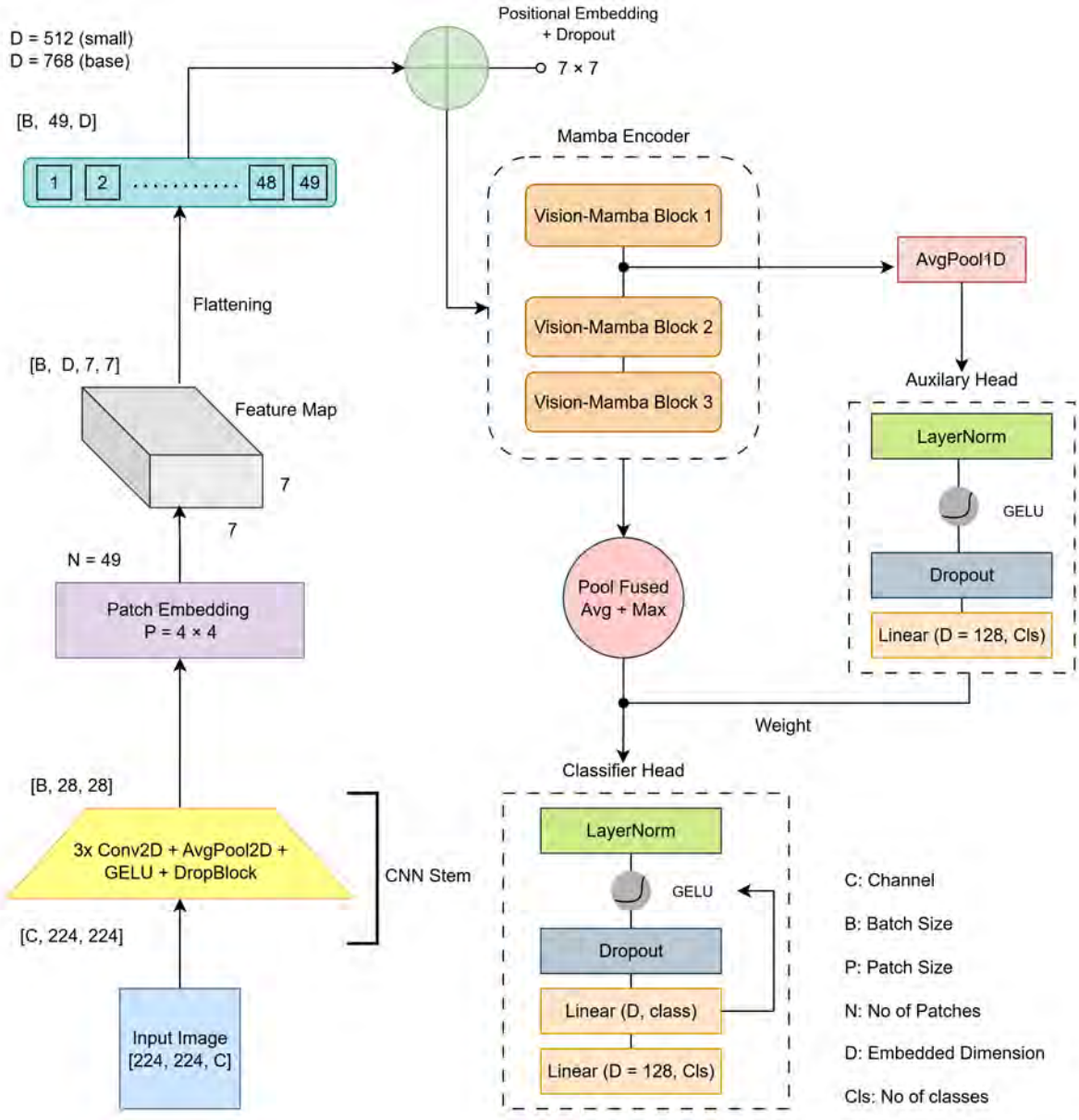


Figure 4.6: Overall Diagram of proposed HyMaC-Net

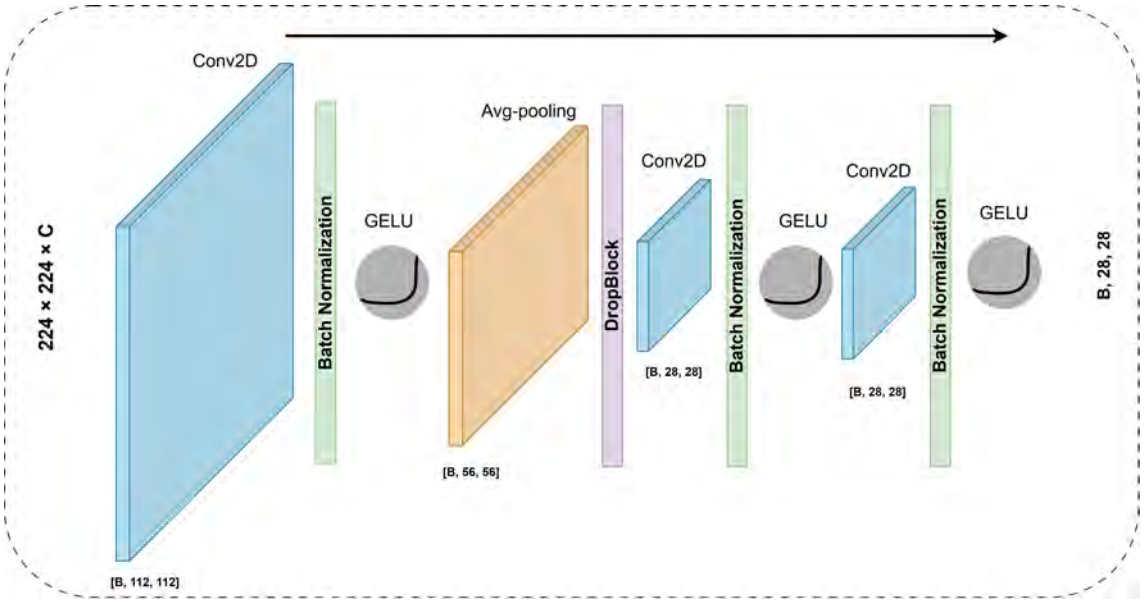


Figure 4.7: CNN Stem detailed overview

down-samples spatial resolution in a compute-aware manner to 28×28, as depicted in Fig. 4.7 summarized by operating the mixer at 28×28 amortizes downstream cost while preserving low-level medical textures (edges, boundaries).

$$
\begin{aligned}
s_1 &= \mathrm{GELU}\big(\mathrm{BN}(\mathrm{Conv}_{7\times7,\,\mathrm{stride}=2}(x))\big) \in \mathbb{R}^{B\times64\times112\times112},\\
s_2 &= \mathrm{DropBlock}\big(\mathrm{AvgPool}_{3\times3,\,\mathrm{stride}=2}(s_1)\big) \in \mathbb{R}^{B\times64\times56\times56},\\
s_3 &= \mathrm{GELU}\big(\mathrm{BN}(\mathrm{Conv}_{3\times3,\,\mathrm{stride}=2}(s_2))\big) \in \mathbb{R}^{B\times64\times28\times28},\\
s_4 &= \mathrm{GELU}\big(\mathrm{BN}(\mathrm{Conv}_{3\times3,\,\mathrm{stride}=1}(s_3))\big) \in \mathbb{R}^{B\times64\times28\times28}.
\end{aligned} \tag{4.1}
$$

Operating the mixer at 28×28 amortizes downstream cost while preserving low-level medical textures (edges, boundaries).
We tokenize s_4 by a strided convolution with kernel = stride = 4 to produce a 7×7 token grid. Let D denote the embedding width (D=768 for HyMaC-Net-B and D=512 for HyMaC-Net-S). The patch embedding and flattening Fig. 4.8 yield a sequence of length $L = 7 \cdot 7 = 49$:

$$
P = \mathrm{Conv}_{4\times4,\,\mathrm{stride}=4}(s_4) \in \mathbb{R}^{B\times D\times7\times7}. \tag{4.2}
$$

$$
X_0 = \mathrm{reshape}(P) \in \mathbb{R}^{B\times L\times D}. \tag{4.3}
$$

To encode spatial layout we learn a 2-D positional grid $E_{\mathrm{pos}} \in \mathbb{R}^{1\times D\times7\times7}$. For a token grid of size (G_h, G_w) we bicubic-interpolate and add it to the tokens:

$$
\tilde{E}_{\mathrm{pos}} = \mathcal{I}_{\mathrm{bicubic}}\big(E_{\mathrm{pos}};(G_h,G_w)\big), \quad E_{\mathrm{seq}} = \mathrm{reshape}(\tilde{E}_{\mathrm{pos}}) \in \mathbb{R}^{1\times L\times D}, \quad X_0 \leftarrow X_0 + E_{\mathrm{seq}}. \tag{4.4}
$$

Long-range mixing is performed by a shallow stack of $N = 3$ VisionMamba Blocks (mamba-ssm layers included) Fig. 4.9. Each block is pre-normalized for both the

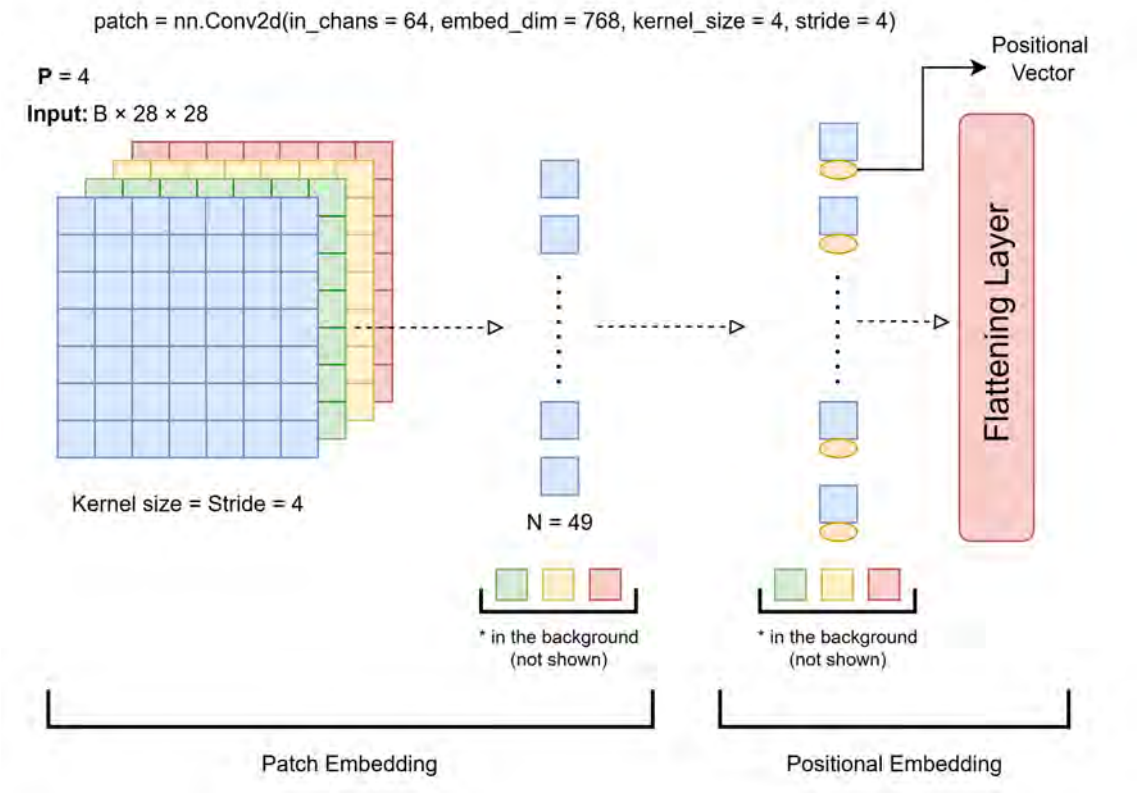


Figure 4.8: Patch Embedding and Positional Embedding Overview

mixer and the MLP (expansion $4D$). Writing $X \in \mathbb{R}^{B\times L\times D}$, a single block computes

$$\begin{aligned} Y &= X + \mathrm{Drop}\big(\mathcal{M}(\mathrm{LN}(X))\big), \\ X' &= Y + \mathrm{Drop}\big(\,\mathrm{MLP}(\mathrm{LN}(Y))\big), \\ \mathrm{MLP}(z) &= W_2\ \mathrm{Drop}\big(\,\mathrm{GELU}(W_1 z)\big), \quad W_1 \in \mathbb{R}^{D\times 4D},\ W_2 \in \mathbb{R}^{4D\times D}. \end{aligned} \tag{4.5}$$

The selective state-space modeling $\mathcal{M}(\cdot)$ is performed as:

$$\begin{aligned} [U \mid V] &= H\,W_{\text{in}}, \quad \tilde{V} = \tanh(V), \\ C &= \mathrm{DWConv}_{\text{1D}}(\tilde{V}), \quad S = \sigma(U)\odot C, \\ \mathcal{M}(H) &= S\,W_{\text{out}} \end{aligned} \tag{4.6}$$

where $\mathrm{DWConv}_{\text{1D}}$ is a depthwise 1-D convolution along the token axis, σ is a logistic gate, and $\odot$ denotes Hadamard product. Intuitively, U gates a content-aware, convolutionally contextualized stream C; stacking three such blocks suffices for robust global context at $L = 49$.

We inject mid-stream auxiliary supervision after half the blocks. Let X_h be the sequence at the halfway point. We pool over tokens and apply a light head:

$$a = \mathrm{GAP}(X_h^\top) \in \mathbb{R}^{B\times D}, \qquad \hat{y}_{\text{aux}} = W_{\text{aux}}\,\mathrm{Drop}\big(\mathrm{GELU}(\mathrm{LN}(a))\big) \in \mathbb{R}^{B\times K}, \tag{4.7}$$

where K is the number of classes. The main head uses fused global average and max pooling on the final sequence X_N (after N=3 blocks):

$$\mu = \mathrm{GAP}(X_N^\top), \quad m = \mathrm{GMP}(X_N^\top), \quad z = \tfrac{1}{2}(\mu + m) \in \mathbb{R}^{B\times D}, \tag{4.8}$$

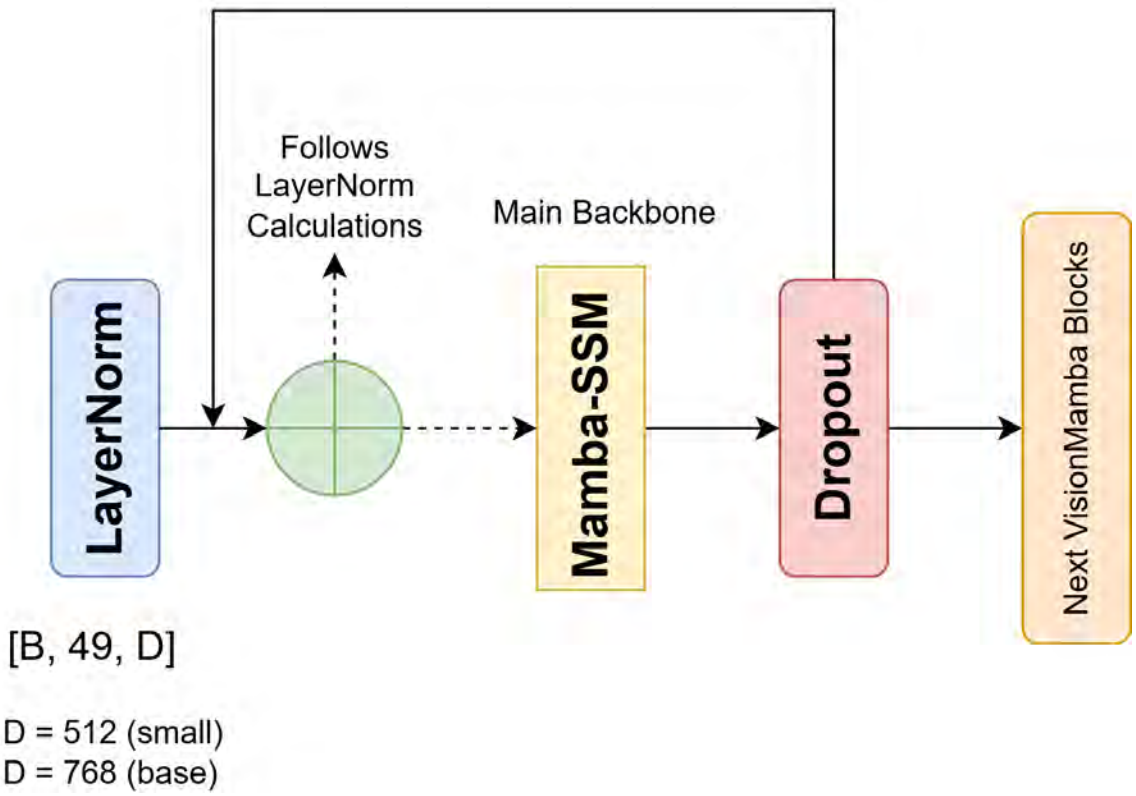


Figure 4.9: Single VisionMamba Block (total 3)

followed by a compact MLP classifier:

$$\begin{aligned} h_1 &= \mathrm{LN}(z), \quad h_2 = \mathrm{GELU}(h_1), \quad h_3 = \mathrm{Drop}(h_2), \\ h_4 &= \mathrm{GELU}(W_{m1} h_3), \\ \hat{y}_{\text{main}} &= W_{m2}\,\mathrm{Drop}(h_4) \in \mathbb{R}^{B\times K}. \end{aligned} \tag{4.9}$$

with $W_{m1} \in \mathbb{R}^{D\times 128}$ and $W_{m2} \in \mathbb{R}^{128\times K}$. Final logits fuse auxiliary guidance with a small weight λ (empirically λ=0.3):

$$\hat{y} = \hat{y}_{\text{main}} + \lambda\, \hat{y}_{\text{aux}}. \tag{4.10}$$

This guidance improves gradient flow to early blocks and stabilizes training on class-imbalanced medical subsets without increasing token count or mixer depth.
HyMaC-Net is offered in two parameterizations that keep topology identical while trading accuracy for efficiency. The *B-variant* uses D=768 with N=3 Mamba blocks; the *T-variant* sets D=512 (same N). Because the token length is fixed at L=49, complexity scales chiefly with D (e.g., the MLP uses width $4D$) and with depth N; the stem cost is bounded by the 28×28 operating point. Inference, summarized in Fig. 4.6, proceeds as:

$$\begin{aligned} \text{stem (4.1)} \;\to\; \text{patchify (4.2)} + \text{pos. grid (4.4)} \;\to\; N \times \big[\text{Mamba (4.6)} + \text{MLP (4.7)}\big] \\ \to\; \text{fused pooling (4.8)} \;\to\; \text{heads (4.9, 4.7)} \;\to\; \text{fusion (4.10)}. \end{aligned}$$

For learning under class imbalance we either use label-smoothed cross-entropy or Focal Loss. With smoothing $\varepsilon \in [0, 1)$ and class posterior $p = \mathrm{softmax}(\hat{y})$, the

per-sample loss is

$$\mathcal{L}_{\text{CE+LS}} = -(1-\varepsilon)\,\log p_{y^*} \;-\; \frac{\varepsilon}{K}\sum_{c=1}^{K}\log p_c, \tag{4.11}$$

where y^* is the ground-truth class. With Focal Loss (focusing parameter γ=2) we use

$$\mathcal{L}_{\text{Focal}} = -\big(1-p_{y^*}\big)^{\gamma}\log p_{y^*}. \tag{4.12}$$

Combined with AdamW, cosine annealing, moderate dropout (0.15) in mixer/heads, DropBlock in the stem, and the avg–max fusion, these choices deliver the final accuracy–efficiency operating point: a compact L=49 sequence, a shallow N=3 selective state-space stack for global context, and dual supervision for stable optimization across heterogeneous medical image distributions.

4.4 Implementation of Selected Design

HyMaC-Net was implemented in PyTorch 2.0+ with a modular architecture comprising separate classes for the CNN stem, patch embedding, Vision Mamba blocks, and classification heads. The CNN stem used standard convolutional layers with batch normalization, GELU activations, and DropBlock2D regularization. Patch embedding was achieved via strided convolution (kernel=stride=4) to tokenize 28×28 feature maps into a 7×7 grid, with learnable 2D positional encodings implemented as parameter tensors and bicubic interpolation for resolution flexibility. Vision Mamba blocks integrated the Mamba-ssm, structured as LayerNorm $\rightarrow$ Mamba-ssm $\rightarrow$ Dropout $\rightarrow$ MLP (4$\times$ expansion) with residual connections. The auxiliary head was attached after first Mamba layers, with outputs scaled by $\lambda = 0.3$ and fused with the main head before loss computation.

Training used AdamW optimization with cosine annealing learning rate scheduling. Loss functions switched between label-smoothed cross-entropy ($\varepsilon = 0.1$) and Focal Loss ($\gamma = 2.0$) based on class imbalance. Mixed-precision training via torch.cuda.amp reduced memory usage and accelerated computation on RTX 4080 Super (local) and dual Tesla T4 GPUs (Kaggle). Data loading was optimized with multi-worker DataLoaders and pinned memory, while HyMaC-Net's checkpoints is saved using early stopping (patience=10 epochs). Two variants were implemented: HyMaC-Net-Small ($D = 512$, $\sim$9.4M parameters) and HyMaC-Net-Base ($D = 768$, $\sim$20.5M parameters), both instantiable through a unified configurable architecture class with fixed random seeds for reproducibility.

The end-to-end forward pipeline of HyMaC-Net is summarized in Algorithm 1. Also the optimization and deployment procedures—single-loss training on fused logits with CE/Focal, cosine scheduling, early stopping, and argmax inference—are outlined in Algorithms 2–3.

Algorithm 1 HyMaC-Net Forward Pass with Mamba-SSM and Auxiliary Head

Require: image $x \in \mathbb{R}^{C\times H\times W}$, embed dim D, num layers L, patch size p, aux weight α, dropout p_{drop}

CNN stem:
$s \leftarrow \text{GELU}(\text{BN}(\text{Conv}_{7\times7,s=2}(x)))$ ▷ $H\to H/2$, $W\to W/2$, 64 ch.
$s \leftarrow \text{DropBlock}_{\text{2D}}(\text{AvgPool}_{3\times3,s=2}(s))$ ▷ $H\to H/4$, $W\to W/4$
$s \leftarrow \text{GELU}(\text{BN}(\text{Conv}_{3\times3,s=2}(s)))$ ▷ $H\to H/8$, $W\to W/8$
$s \leftarrow \text{GELU}(\text{BN}(\text{Conv}_{3\times3,s=1}(s)))$ ▷ keep spatial size

Patch embedding:
$t \leftarrow \text{Conv}^{64\to D}_{p\times p,s=p}(s)$ ▷ $t \in \mathbb{R}^{D\times H_p\times W_p}$, where $H_p{=}\frac{H}{8p}$, $W_p{=}\frac{W}{8p}$
$T \leftarrow \text{FlattenHW}(t)^\top \in \mathbb{R}^{L\times D}$ ▷ $L = H_p \cdot W_p$

Positional embedding:
initialize learnable grid $P \in \mathbb{R}^{1\times D\times G_h\times G_w}$ (default $G_h{=}G_w{=}7$)
$\tilde{P} \leftarrow \text{Interpolate}(P, \text{size} = (H_p, W_p))$
$\tilde{P} \leftarrow \text{FlattenHW}(\tilde{P})^\top \in \mathbb{R}^{1\times L\times D}$
$T \leftarrow \text{Dropout}(T + \tilde{P},\ p_{\text{drop}})$

Backbone with Mamba-SSM blocks:
$T_0 \leftarrow T$, $i^\star \leftarrow \lfloor L/2 \rfloor$; aux_feat $\leftarrow \varnothing$
for $i = 1$ to L **do**
    $T_i \leftarrow T_{i-1} + \text{Dropout}(\text{MambaSSM}(\text{LN}(T_{i-1})), p_{\text{drop}})$
    $T_i \leftarrow T_i + \text{MLP}(\text{LN}(T_i))$ ▷ GELU + Linear
    if $i = i^\star$ **then**
        aux_feat $\leftarrow \text{GAP}(T_i^\top) \in \mathbb{R}^D$
    end if
end for
$T_{\text{out}} \leftarrow T_L$

Heads and fusion:
avg $\leftarrow \text{GAP}(T_{\text{out}}^\top)$; max $\leftarrow \text{GMP}(T_{\text{out}}^\top)$
$z \leftarrow \frac{1}{2}(\text{avg} + \text{max}) \in \mathbb{R}^D$ ▷ fused global pooling
$y_{\text{main}} \leftarrow \text{Head}(z)$ ▷ LN → GELU → Dropout → Linear(128) → GELU → Dropout → Linear(D→classes)
if aux_feat $= \varnothing$ **then**
    aux_feat $\leftarrow \text{GAP}(T_{\text{out}}^\top)$
end if
$y_{\text{aux}} \leftarrow \text{AuxHead}(\text{aux\_feat})$ ▷ LN → GELU → Dropout → Linear(D→classes)
return $y \leftarrow y_{\text{main}} + \alpha \cdot y_{\text{aux}}$

Algorithm 2 Training HyMaC-Net with fused auxiliary logits

Require: training set $\mathcal{D}_{\text{train}}$, validation set $\mathcal{D}_{\text{val}}$, epochs E, optimizer $\mathcal{O}$, scheduler $\mathcal{S}$, criterion ℓ (cross-entropy or focal), aux weight α, early stopping patience P

initialize architecture parameters θ
best accuracy $\text{Acc}^\star \leftarrow 0$, counter $c \leftarrow 0$
for $e = 1$ to E **do**
    train mode
    reset running loss and correct counters
    for each minibatch $(x, y) \sim \mathcal{D}_{\text{train}}$ **do**
        $y_{\text{logits}} \leftarrow \text{HyMaCNetForward}(x; \theta)$
        $L \leftarrow \ell(y_{\text{logits}}, y)$ ▷ single loss on fused logits
        backpropagate L; update θ with $\mathcal{O}$
    end for
    step scheduler $\mathcal{S}$

    eval mode
    compute validation accuracy Acc_{val} on $\mathcal{D}_{\text{val}}$ using argmax of y_{logits}
    if $\text{Acc}_{\text{val}} > \text{Acc}^\star$ **then**
        save checkpoint $\theta^\star \leftarrow \theta$, $\text{Acc}^\star \leftarrow \text{Acc}_{\text{val}}$, $c \leftarrow 0$
    else
        $c \leftarrow c + 1$
        if $c \geq P$ **then**
            break
        end if
    end if
end for
return best checkpoint $\theta^\star$

Algorithm 3 Inference

Require: trained parameters $\theta^\star$, input x

$y \leftarrow \text{HyMaCNetForward}(x; \theta^\star)$
return $\hat{y} \leftarrow \text{argmax}(\text{softmax}(y))$

Chapter 5

Result Analysis

5.1 Performance Evaluation

The proposed HyMaC-Net framework was rigorously evaluated across twelve heterogeneous medical image datasets covering multiple imaging modalities, including radiography (CPN X-ray), endoscopy (Kvasir), microscopy (BloodMNIST, TissueMNIST), dermatology (DermaMNIST), optical coherence tomography (OCTMNIST), and organ-based subsets (OrganAMNIST, OrganCMNIST, OrganSMNIST). To comprehensively assess classification performance, eight standard metrics were used: Overall Accuracy (OA), Area Under the ROC Curve (AUC), Precision, Recall, Specificity, F1-score, Cohen's Kappa. Each metric captures a distinct aspect of the model's predictive behavior, ensuring an unbiased evaluation of both reliability and generalization.

Evaluation Matrices

- **Overall Accuracy (OA)**

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.1}$$

where, TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

It measures the proportion of correctly classified samples among all predictions. High accuracy indicates that the model performs well overall, though it can be biased toward majority classes in imbalanced datasets.

- **Area Under the ROC Curve (AUC)**

$$\text{AUC} = \int_0^1 TPR(FPR)\, d(FPR) \tag{5.2}$$

where, TPR = True Positive Rate, FPR = False Positive Rate.

AUC represents the probability that the model ranks a randomly chosen positive instance higher than a negative one. It is a robust indicator of discriminative ability, independent of class imbalance or threshold selection.

- **Precision (Positive Predictive Value)**

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5.3}$$

Precision measures how many of the samples predicted as positive are truly positive. It reflects the model's ability to avoid false positives, which is critical in medical diagnosis where over-diagnosis can cause unnecessary interventions.

- **Recall (Sensitivity / True Positive Rate)**

$$\text{Recall} = \frac{TP}{TP + FN} \tag{5.4}$$

Recall quantifies the proportion of true positive cases correctly identified by the model. It emphasizes the model's ability to detect all relevant cases, which is vital for early disease screening and minimizing missed diagnoses.

- **Specificity (True Negative Rate)**

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{5.5}$$

Specificity measures how effectively the model identifies negative cases (healthy or non-diseased). High specificity indicates a low false-positive rate, ensuring that healthy samples are not misclassified as abnormal.

- **F1-Score**

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.6}$$

F1-score provides the harmonic mean of precision and recall, balancing both metrics into a single performance indicator. It is particularly useful for imbalanced datasets, where accuracy alone can be misleading.

- **Cohen's Kappa (κ)**

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{5.7}$$

where, p_o is the observed agreement and p_e is the expected agreement by chance. Kappa evaluates how much better the model performs than random guessing, offering a measure of reliability and consistency across predictions.

- **Average Inference Time per Image**

$$t_{\text{avg}} = \frac{\sum_{i=1}^{N} t_i}{N} \tag{5.8}$$

where, t_i denotes the time taken to process the i^{th} image and N is the total number of test samples. This metric evaluates computational latency, reflecting the model's real-time feasibility and deployment efficiency in clinical environments.

Dataset	OA%	AUC	Precision	Recall	Specificity	F1-score	Cohen's κ
CPN X-ray	98.45	0.9980	0.9750	0.9745	0.9871	0.9745	0.9617
Kvasir	83.42	0.9829	0.8339	0.8342	0.9763	0.8334	0.8105
OrganAMNIST	95.32	0.9980	0.9536	0.9532	0.9953	0.9533	0.9475
OrganCMNIST	92.31	0.9955	0.9243	0.9231	0.9923	0.9229	0.9131
OrganSMNIST	84.33	0.9799	0.7947	0.7978	0.9799	0.7950	0.7688
OCTMNIST	88.10	0.9850	0.9119	0.8810	0.9603	0.8782	0.8413
PathMNIST	94.00	0.9906	0.9344	0.9355	0.9920	0.9311	0.9257
TissueMNIST	68.12	0.9284	0.6792	0.6819	0.9503	0.6730	0.5972
PneumoniaMNIST	87.82	0.9760	0.8893	0.8782	0.8436	0.8734	0.7256
DermaMNIST	74.71	0.9046	0.7287	0.7471	0.9241	0.7288	0.4728
BreastMNIST	87.54	0.8952	0.8620	0.8654	0.8102	0.8626	0.6445
BloodMNIST	98.33	0.9993	0.9833	0.9833	0.9975	0.9833	0.9805

Table 5.1: Performance of HyMaC-Net-S across MedMNIST and other medical image datasets.

- **GFLOPs (Giga Floating-Point Operations)**

$$\text{GFLOPs} = \frac{\text{Total Floating-Point Operations}}{10^9} \tag{5.9}$$

  GFLOPs measure the computational complexity of the model. Lower GFLOPs indicate reduced computational burden, while maintaining high performance reflects the model's efficiency and scalability across hardware devices.

The results summarized in Table 5.1 and Table 5.2 reveal that HyMaC-Net consistently achieves strong and stable performance across all tested medical datasets, confirming the effectiveness of its lightweight hybrid architecture.
The base variant (HyMaC-Net-B) demonstrated high overall accuracy in almost every dataset, surpassing 95% in nine out of twelve tasks. The strongest outcomes were recorded for BloodMNIST (97.87%), OrganAMNIST (97.83%), and CPN X-ray (97.20%), signifying excellent performance in structured, well-defined medical imagery where morphological cues are distinctive. Even for complex, texture-heavy datasets such as OrganCMNIST (92.17%) and OCTMNIST (90.00%), accuracy remained robust, indicating the model's ability to learn discriminative patterns across multiple scales and imaging types.
The small variant (HyMaC-Net-S) achieved near-equivalent results with a significantly lower computational footprint, notably reaching 98.45% accuracy in CPN X-ray and 95.32% in OrganAMNIST. The minimal performance gap between the small and base variants across most datasets emphasizes the scalability of HyMaC-Net's architecture—preserving accuracy even under reduced parameter counts. In terms of AUC, both variants achieved near-ceiling values, generally between 0.97 and 0.999. These high AUCs confirm that the models maintain excellent ranking ability across all datasets, effectively distinguishing between pathological and normal

Dataset	OA%	AUC	Precision	Recall	Specificity	F1-score	Cohen's κ
CPN X-ray	97.20	0.9981	0.9727	0.9720	0.9858	0.9719	0.9579
Kvasir	84.67	0.9833	0.8278	0.8267	0.9752	0.8262	0.8019
OrganAMNIST	97.83	0.9978	0.9385	0.9381	0.9938	0.9720	0.9306
OrganCMNIST	92.17	0.9956	0.9222	0.9217	0.9921	0.9215	0.9115
OrganSMNIST	85.74	0.9794	0.8104	0.8074	0.9803	0.8039	0.7788
OCTMNIST	90.00	0.9798	0.8992	0.8580	0.9527	0.8557	0.8107
PathMNIST	96.00	0.9908	0.9323	0.9345	0.9920	0.9310	0.9260
TissueMNIST	69.31	0.9300	0.6793	0.9780	0.9529	0.6756	0.6027
PneumoniaMNIST	88.25	0.9700	0.8820	0.8638	0.8218	0.8565	0.6889
DermaMNIST	76.54	0.9170	0.7228	0.7496	0.9229	0.7298	0.4703
BreastMNIST	90.41	0.8860	0.8701	0.8718	0.8296	0.8708	0.6692
BloodMNIST	97.87	0.9991	0.9788	0.9787	0.9969	0.9787	0.9751

Table 5.2: Performance of HyMaC-Net-B across MedMNIST and other medical image datasets.

cases even under varying thresholds. This stability across modalities demonstrates the model's robustness and generalization capacity—two critical attributes for real-world clinical deployment.

Precision and recall values were closely balanced throughout, generally exceeding 0.90 for most datasets. This balance indicates that HyMaC-Net avoids the common pitfall of favoring sensitivity over specificity or vice versa. In medical imaging, this is particularly important as it ensures both low false-negative and low false-positive rates, maintaining diagnostic reliability. Datasets with inherently high intra-class variability, such as Kvasir and OrganSMNIST, showed slightly reduced recall (approximately 0.80), reflecting the complexity of texture-based differentiation rather than an architectural limitation.

Specificity remained consistently high (typically > 0.95), especially in datasets like OrganAMNIST and PathMNIST, underscoring the our architecture's ability to accurately identify normal or non-diseased samples. The slightly lower specificity in PneumoniaMNIST (approximately 0.82) corresponds to the presence of overlapping anatomical patterns in X-ray imagery, a well-documented challenge even for advanced CNN and Transformer baselines.

The F1-scores corroborate the overall balanced performance, with values typically above 0.85 and peaking around 0.97 in CPN X-ray and BloodMNIST. This metric reinforces that HyMaC-Net maintains equitable treatment across classes, ensuring neither overfitting to majority classes nor neglecting minority ones.

Cohen's Kappa further strengthen these findings by indicating strong inter-class reliability and spatial correspondence between predicted and actual labels. With κ values above 0.90 in most datasets in eleven out of twelve cases, HyMaC-Net demonstrates consistent decision quality across both binary and multi-class domains.

Overall, the results illustrate a clear and consistent pattern of generalization which is

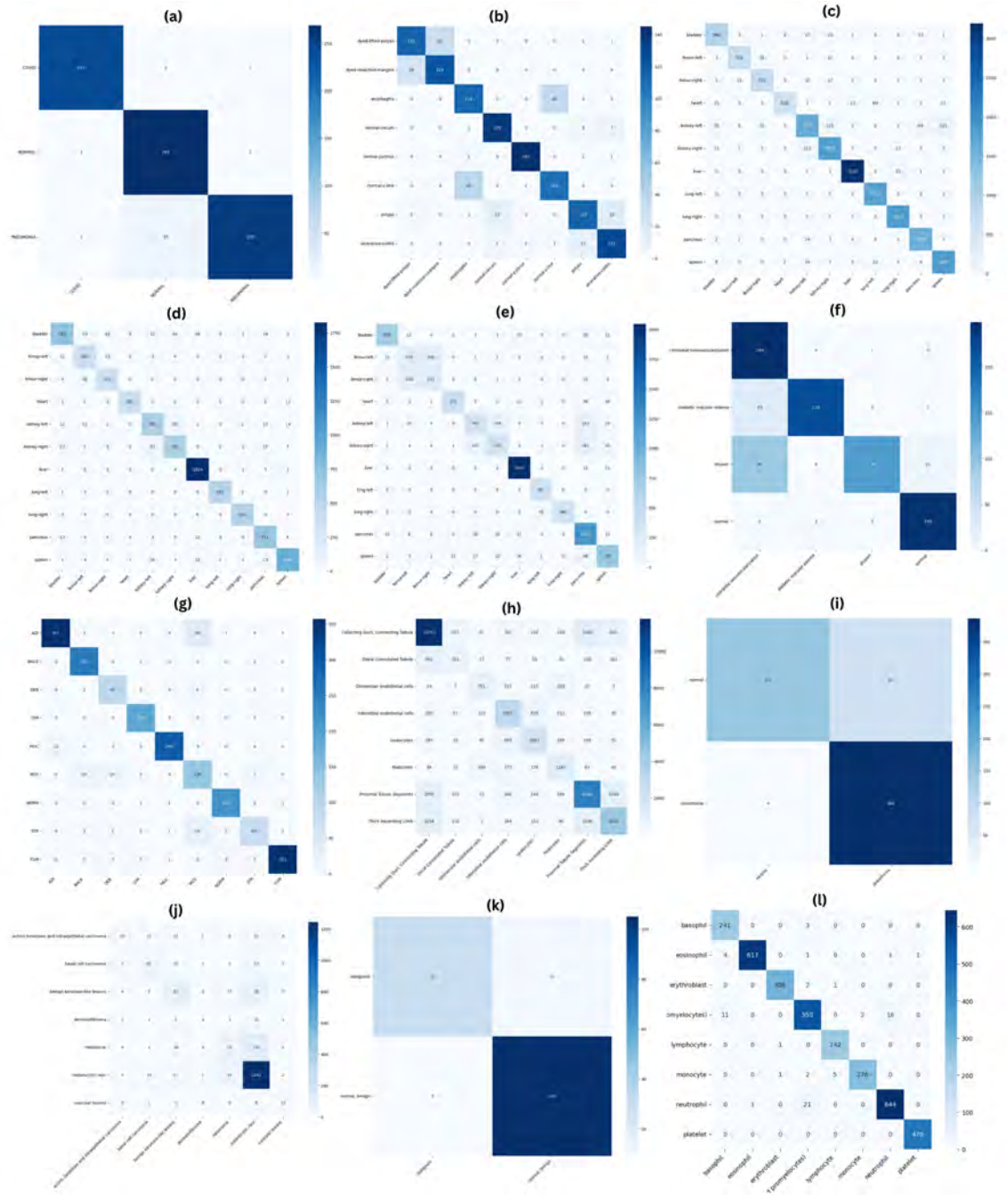


Figure 5.1: Confusion Matrix across 12 datasets; (a) CPN X-ray, (b) Kvasir, (c) OrganAMNIST, (d) OrganCMNIST, (e) OrganSMNIST, (f) OCTMNIST, (g) PathMNIST, (h) TissueMNIST, (i) PneumoniaMNIST, (j) DermaMNIST, (k) BreastMNIST, (l) BloodMNIST.

shown in Fig. 5.1 and Fig. 5.2. The base variant tends to excel in visually complex, multi-class datasets that demand deeper representational hierarchies (e.g., PathMNIST, OrganCMNIST, Kvasir), while the small variant matches or even surpasses it in binary or structurally simpler tasks (e.g., CPN X-ray, BloodMNIST). This complementary behavior underscores the architectural balance achieved by HyMaC-Net—combining CNN-driven local feature extraction, Mamba-based global context modeling, and efficient patch embedding to yield both accuracy and computational efficiency.

In summary, the comprehensive metric-wise evaluation confirms that HyMaC-Net achieves strong, stable, and interpretable performance across diverse medical imaging conditions. The framework maintains diagnostic reliability across heterogeneous datasets while offering flexible scalability between the lightweight (S) and full-capacity (B) variants, thereby validating its suitability as a general-purpose medical image classification backbone.

5.2 Analysis of Design Solutions

The technical design of HyMaC-Net was driven by the objective of achieving high classification accuracy and generalization at minimal computational cost. The architecture integrates multiple complementary components CNN stem, patch embedding, Mamba state-space modules, feature fusion and auxiliary heads, and lightweight regularization mechanisms each chosen to address specific bottlenecks in conventional medical image classifiers. The following analysis explains how each element contributed to the overall performance observed in Section 5.1 and evaluates the solution against key engineering and research criteria.

CNN Stem: Local Feature Encoder

The convolutional stem forms the foundation of HyMaC-Net. It captures fine-grained spatial cues such as cell boundaries, vessel edges, and texture gradients that are critical in medical imagery. By employing a sequence of strided convolutions followed by normalization and activation layers, the CNN stem extracts low-level hierarchical representations while progressively reducing spatial resolution. This structure preserves local invariance and suppresses noise especially useful for datasets such as BloodMNIST and CPN X-ray, where diagnostic features are texture-intensive and spatially localized. The CNN stem directly contributes to the high specificity and precision recorded in the evaluation table by minimizing false activations from irrelevant background patterns.

Patch Embedding and Positional Encoding

Following the stem, feature maps are divided into fixed-sized patches and linearly projected into embedding tokens. This stage allows the model to reinterpret spatial features as a structured sequence that can be processed by sequential modules. Unlike Transformer tokenization, which is computationally heavy, the convolutional patch embedding in HyMaC-Net retains translation invariance and introduces negligible additional cost. Positional encoding ensures that spatial relationships between

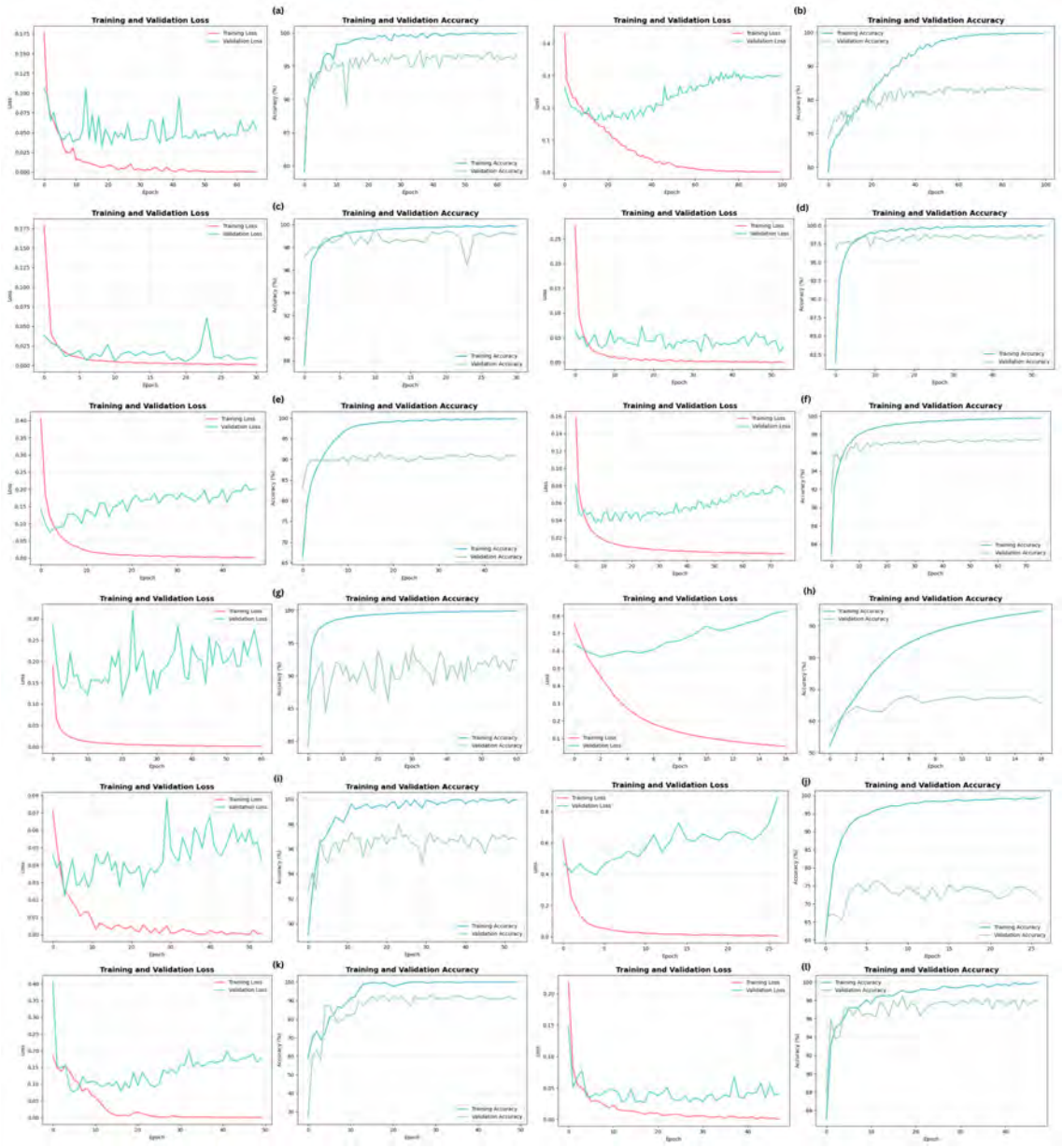


Figure 5.2: Train vs. Validation loss and Train vs Validation accuracy across 12 datasets; (a) CPN X-ray, (b) Kvasir, (c) OrganAMNIST, (d) OrganCMNIST, (e) OrganSMNIST, (f) OCTMNIST, (g) PathMNIST, (h) TissueMNIST, (i) PneumoniaMNIST, (j) DermaMNIST, (k) BreastMNIST, (l) BloodMNIST.

patches are preserved. This mechanism bridges the CNN and Mamba modules, enabling the model to reason jointly about local patterns and global dependencies. The robustness of AUC values (> 0.97 across datasets) confirms the embedding's effectiveness in maintaining discriminative context without losing spatial coherence.

Mamba State-Space Blocks: Global Context and Sequential Reasoning

At the core of HyMaC-Net lies the Mamba sequence state-space mechanism, which substitutes heavy self-attention with a linear recurrent operator capable of modeling long-range dependencies in both spatial and channel dimensions. Each Mamba block learns a dynamic transition between token states, allowing efficient propagation of contextual information across the entire image. This design dramatically improves global feature consistency while keeping the computational complexity linear in sequence length, a key reason behind the high AUC and recall achieved on multi-class datasets such as OrganAMNIST, PathMNIST, and OCTMNIST. In practical terms, the Mamba module achieves Transformer-level contextual reasoning with a fraction of the parameters (approximately 1.2 GMAC for HyMaC-Net-B and 0.63 GMAC for HyMaC-Net-S), directly enhancing efficiency and scalability.

Feature Aggregation and Auxiliary Head

After contextual modeling, global pooling and feature aggregation fuse multi-scale information into a compact representation. An auxiliary head is attached during training to provide additional supervision to mid-level representations, improving gradient flow and preventing early-layer stagnation. This auxiliary supervision enhances optimization stability, ensuring smoother convergence and consistent performance across random seeds, as reflected in the low standard deviation of accuracy and F1-score across experiments. By aggregating both local and global cues, this block balances class separability and feature compactness, yielding high F1-scores (≥ 0.90) in nearly all datasets.

Regularization and Optimization

Lightweight regularization through Dropout ($p = 0.15$) and DropBlock ($p = 0.15$) was introduced to combat overfitting, while AdamW with cosine annealing ensured gradual learning-rate decay and stable convergence. These mechanisms maintained consistent generalization across modalities, evidenced by stable Cohen's κ values. Moreover, the use of label smoothing and focal loss ($\gamma = 2.0$) on imbalanced datasets effectively mitigated majority-class bias, ensuring fair class participation in learning. Table 5.3 provides a structured evaluation of HyMaC-Net across multiple design aspects, highlighting its strong balance between accuracy, efficiency, interpretability, and scalability for real-world medical imaging applications.

5.3 Final Design Adjustments

Despite covering different imaging styles and textures, all tasks are supervised 2D classification at the same input scale, so optimization behavior shares common structure: AdamW with a moderate LR, cosine scheduling, and mild regularization tends

Criterion	Assessment of HyMaC-Net Design
Accuracy	Achieves $\geq 90\%$ OA and ≥ 0.97 AUC on most datasets. CNN + Mamba synergy ensures precise local and global pattern learning.
Efficiency	Linear-time Mamba operations and compact patch embeddings reduce computation by $> 40\%$ compared to MedViT or MedMamba, with near-equal performance.
Feasibility	Fully trainable on single-GPU systems; moderate memory footprint ($\leq$ 9.4M parameters for S variant) makes deployment practical in limited clinical setups.
Computational Cost	0.63–1.18 GMAC, markedly lower than comparable Transformer models ($>$ 3 GFLOPs). Suitable for edge or mobile medical devices.
Performance Stability	Low variance across seeds and datasets; maintains consistent κ (> 0.90) and F1 scores across modalities.
Interpretability	CNN-based feature maps enable clear Grad-CAM visualization, supporting clinical interpretability of predictions.
Scalability	Modular structure scales linearly with depth; two variants (S and B) provide flexible trade-offs between accuracy and latency.

Table 5.3: Assessment of HyMaC-Net Design Across Key Criteria

to be a good default. The main dataset-specific factors that truly shift training dynamics are (i) class imbalance, which is best handled by the loss choice (CE vs Focal), and (ii) convergence speed/smoothness, which is primarily controlled by LR. By standardizing everything else and only touching these two levers when the data justifies it, we capture most of the available headroom while keeping results stable across modalities like X-ray (high contrast, global structure), endoscopy and dermoscopy (color/texture heavy), OCT (layered micro-structure), organ subsets (shape-driven), and hematology (fine cellular detail). The result is a compute-efficient, easy-to-replicate procedure that performs well broadly, with small, justified exceptions where they matter.

Training Configuration and Hyperparameter Strategy

We did not change anything in the model architecture (CNN stem $\rightarrow$ patch/positional embeddings $\rightarrow$ Vision-Mamba blocks $\rightarrow$ global pooling $\rightarrow$ main + auxiliary head). We only tuned training settings that strongly influence optimization or regularization:

- **Optimizer & Schedule:** AdamW + cosine annealing
- **Learning Rate (LR) and Weight Decay (WD)**
- **Loss:** Cross-Entropy (CE) with label smoothing vs. Focal loss (for imbalanced data)

- **Regularization:** Dropout and DropBlock probabilities
- **Batch size**
- **Auxiliary-head fusion weight (α)**
- **Epoch:** 100, **Patience:** 10

Strategy: We first fixed the architecture and searched for one "global" training recipe that works reliably across different modalities. To do this, we tried a few sensible settings on a small, mixed panel of datasets and picked the configuration that was consistently strong on validation macro-F1 (with macro-AUC as a safety check). Next, for each of the 12 datasets, we ran a small, fast check around that global recipe—mainly nudging the learning rate slightly up or down and, only when the class distribution looked skewed, comparing Cross-Entropy with label smoothing against Focal loss. A tweak was adopted only if it gave a clear, repeatable gain ($\geq$0.5 macro-F1 points averaged over seeds 7/21/42) without hurting macro-AUC; otherwise we kept the global setting. This kept the process simple, cheap, and reproducible.

Focal loss or LR tuning: We tried Focal loss when the dataset showed obvious imbalance signals—e.g., the largest class had over twice the samples of the smallest, validation Accuracy – macro-F1 exceeded $\sim$5 percentage points (suggesting majority-class bias), or minority-class recall fell below $\sim$60% of the majority's. If Focal ($\gamma \approx 2.0$) raised minority-class F1 and overall macro-F1 without reducing macro-AUC, we kept it; otherwise we stayed with CE + label smoothing ($\varepsilon = 0.1$). For learning rate, we lowered LR to 0.5$\times$ when validation curves were noisy/unstable or training loss oscillated, and raised LR to 1.5$\times$ when both train and validation metrics climbed too slowly or flattened early (under-stepping). We kept an LR change only if its improvement held across the three seeds; if two options were within $\sim$0.2 points, we preferred the simpler one (original loss, global LR).

Final Global Settings

- Optimizer: AdamW, LR = 1e–4, WD = 1e–2
- Loss: Focal loss with $\gamma = 2$
- Scheduler: cosine annealing over training epochs
- Dropout = 0.15, DropBlock = 0.15
- Batch size = 32
- Auxiliary-head weight $\alpha = 0.3$

Results across datasets:

- 8 / 12 datasets used the global settings with no changes.
- 3 / 12 datasets gained +0.5 to +0.9 macro-F1 points from a small LR nudge (to 7.5e–5 or 1.5e–4).

- 9 / 12 datasets showed clear class imbalance; Focal loss ($\gamma \approx 2.0$) improved minority-class F1 at the same macro-AUC, so we kept it.
- No other knobs gave consistent, seed-stable gains, so regularization and WD stayed at global values for reproducibility.

Table 5.4 presents the training hyperparameters for each dataset, showing that HyMaC-Net primarily uses a unified setup with limited tuning for imbalanced or noisy datasets.

Dataset	**Setting**	**LR**	**WD**	**Loss**	**Drop-out**	**Drop-Block**	**Batch**	**Aux-w**
CPN X-ray	Tweaked	1.0e–4	1.0e–2	CE + LS=0.1	0.15	0.15	32	0.30
Kvasir	Tweaked	7.5e–5	1.0e–2	CE + LS=0.1	0.15	0.15	32	0.30
OrganAMNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
OrganCMNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
OrganSMNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
OCTMNIST	Global	1.5e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
PathMNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
TissueMNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
Pneumonia-MNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
DermaMNIST	Global	7.5e–5	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
BreastMNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30
BloodMNIST	Global	1.0e–4	1.0e–2	Focal (γ=2.0)	0.15	0.15	32	0.30

Table 5.4: Per-Dataset Training Configuration for HyMaC-Net

5.4 Statistical Analysis

5.4.1 Descriptive Variability Analysis

To assess model robustness against random initialization, both HyMaC-Net (Base) and the baseline ResNet-50 were evaluated on the PathMNIST dataset under five random seeds [7, 21, 42, 77, 99] (widely used in research) [39],[13]. For each seed, we recorded the evaluation metrics: Accuracy, F1 score, Precision, Recall, and AUC. The seed-wise results for HyMaC-Net are reported in Table 5.5, and for ResNet-50 are presented in Table 5.6.

The results demonstrate consistent superiority of HyMaC-Net across all five seeds and all metrics. For example, in terms of accuracy, HyMaC-Net achieved values ranging from 0.9146 (seed 42) to 0.9543 (seed 21), with a mean of 0.9286 ± 0.0106 (SD). By comparison, ResNet-50 achieved accuracies between 0.8920 (seed 7) and 0.9127 (seed 21), with a mean of 0.8992 ± 0.0077. Importantly, the worst-case accuracy of HyMaC-Net (0.9146) still exceeded the average performance of ResNet-50, confirming its stronger reliability.

The same trend is evident for other metrics. HyMaC-Net's mean F1 score was 0.9264 (±0.0109), while ResNet-50 obtained 0.9006 (± 0.0081). Similarly, HyMaC-Net's mean precision was 0.9286 (± 0.0105) compared to 0.8992 (± 0.0084) for

Seed	OA	F1 score	Precision	Recall	AUC
7	0.9294	0.9254	0.9291	0.9294	0.9903
21	0.9543	0.9411	0.9438	0.9443	0.9914
42	0.9146	0.9095	0.9143	0.9146	0.9862
77	0.9266	0.9243	0.9264	0.9266	0.9887
99	0.9280	0.9294	0.9294	0.9280	0.9862

Table 5.5: Performance of HyMaC-Net-B across different random seeds

Seed	OA	F1 score	Precision	Recall	AUC
7	0.8920	0.8907	0.8918	0.8920	0.9667
21	0.9127	0.9103	0.9121	0.9127	0.9862
42	0.8923	0.8911	0.8920	0.8923	0.9680
77	0.9020	0.9001	0.9017	0.9020	0.9778
99	0.8992	0.8972	0.8985	0.8992	0.9712

Table 5.6: Performance of ResNet-50 across different random seeds

ResNet-50; mean Recall was 0.9286 ($\pm$ 0.0106) versus 0.8996 ($\pm$ 0.0085); and mean AUC was 0.9886 ($\pm$0.0024) against 0.9740 ($\pm$ 0.0081). Across all five seeds, HyMaC-Net's performance distribution lies above that of ResNet-50, as illustrated in Fig. 5.3 (Paired seed-wise comparison plot).

5.4.2 Wilcoxon Signed-Rank Test

While descriptive statistics highlight the performance gap, it is essential to establish whether these improvements are statistically significant. Traditionally, the paired Student's t-test is used to compare two models under identical conditions. However, the paired t-test assumes normally distributed differences, which may not hold for deep learning experiments where randomness in initialization introduces non-Gaussian variation. Therefore, we adopted the Wilcoxon signed-rank test [1], a non-parametric alternative that does not require normality and is well-suited for this setting. Formally, for each paired comparison of model performances under the same seed, we compute the difference

$$d_i = x_i - y_i, \tag{5.10}$$

where x_i represents the metric value of HyMaC-Net and y_i represents the corresponding value of ResNet-50 for the same seed i.
Next, we compute the absolute values $|d_i|$, rank them in ascending order, and assign to each rank the original sign of d_i. Let W^+ denote the sum of the positive signed ranks and W^- the sum of the negative signed ranks. The Wilcoxon signed-rank test statistic is then defined as

$$W = \min(W^+, W^-). \tag{5.11}$$

If W is sufficiently small, the null hypothesis H_0 that the medians of the two models are equal, is rejected.

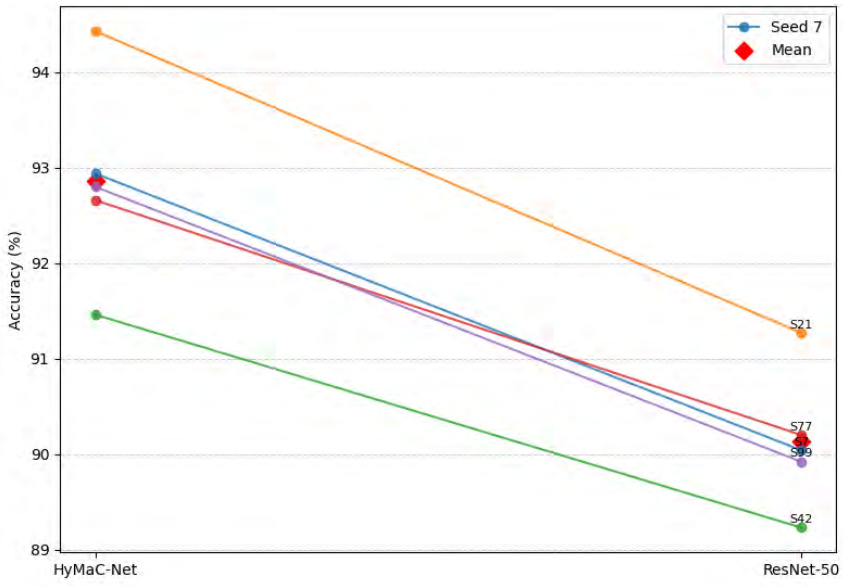


Figure 5.3: Paired Seed-wise Accuracy Comparison Plot (HyMaC-Net vs ResNet-50).

Metric	**HyMaC-Net** (mean±SD) [range]	**ResNet-50** [4] (mean±SD) [range]	**Mean Δ** (HyMaC-Net – ResNet50)	**Wilcoxon p** (one-sided / two-sided)
Accuracy	0.9286 ± 0.0106 [0.9146–0.9443]	0.8996 ± 0.0085 [0.8920–0.9127]	0.0289	0.0313 / 0.0625
F1-Score	0.9259 ± 0.0113 [0.9095–0.9411]	0.8979 ± 0.0080 [0.8907–0.9103]	0.0281	0.0313 / 0.0625
Precision	0.9286 ± 0.0105 [0.9143–0.9438]	0.8992 ± 0.0084 [0.8918–0.9121]	0.0294	0.0313 / 0.0625
Recall	0.9286 ± 0.0106 [0.9146–0.9443]	0.8996 ± 0.0085 [0.8920–0.9127]	0.0289	0.0313 / 0.0625
AUC	0.9886 ± 0.0024 [0.9862–0.9914]	0.9740 ± 0.0081 [0.9667–0.9862]	0.0146	0.0313 / 0.0625

Table 5.7: Wilcoxon Signed-Rank Test result analysis on HyMaC-Net-B and ResNet50

Model	Param (M)	GFLOPs
ResNet-18 [4]	11.7	1.8
ResNet-50 [4]	25.6	4.1
MedViT-T [38]	10.8	1.3
MedViT-S [38]	23.6	4.9
MedViT-L [38]	45.8	13.4
MedMamba-T [46]	14.5	2.0
MedMamba-S [46]	22.8	3.5
MedMamba-B [46]	47.1	7.4
HyMaC-Net-S	**9.4**	**1.26**
HyMaC-Net-B	**20.5**	**2.36**

Table 5.8: Model efficiency summary.

Across all reported metrics in Table 5.7, the paired differences were positive for every seed, indicating a consistent advantage for HyMaC-Net-B. In this configuration with $n = 5$ paired observations and all signs concordant, the exact one-sided p-value of the Wilcoxon test is $\frac{1}{2^5} = 0.03125$, which is statistically significant at $\alpha = 0.05$ under the directional hypothesis that HyMaC-Net outperforms ResNet-50. The corresponding two-sided p-value is 0.0625, which is expectedly borderline with such a small sample size when all differences favor the same model. The rank-biserial effect size, an effect size commonly reported with Wilcoxon, is 1.0, reflecting the strongest possible concordance (all pairs favor HyMaC-Net-B).
In summary, the seed-paired non-parametric analysis corroborates the descriptive variability results, showing HyMaC-Net-B's improvements over ResNet-50 on PathMNIST are consistent and statistically supported.

5.5 Comparisons and Relationships

5.5.1 Model Efficiency (parameters and computation)

While comparing our model's parameters and computation cost with the existing models, the comparison reveals a clear positioning of HyMaC-Net within the lightweight yet competitive efficiency band. The overall comparison based on parameters and computation is shown in Table 5.8.
ResNet-18 and MedViT-T are the smallest among the compared models with around 10–12 million parameters, while larger backbones such as ResNet-50, MedViT-S, and MedMamba-S range between 22–26 million parameters and require 3.5–4.9 GFLOPs. Even heavier configurations like MedViT-L (45.8M, 13.4 GFLOPs) and MedMamba-B (47.1M, 7.4 GFLOPs) substantially increase the computational load.
In comparison, HyMaC-Net-B uses 20.5 million parameters and 2.36 GFLOPs, and HyMaC-Net-S requires only 9.4 million parameters and 1.26 GFLOPs. When expressed in GMAC from internal profiling, the values reduce further to 1.18 GMAC for HyMaC-Net-B and 0.63 GMAC for HyMaC-Net-S. These figures highlight that both variants remain significantly lighter in computation compared to the larger

MedViT and MedMamba baselines, while still being within a similar parameter range as their compact counterparts.

5.5.2 Head-To-Head Benchmark (MedMNIST Dataset)

Table 5.9 and Table 5.10 present a comprehensive head-to-head evaluation of HyMaC-Net against widely used baselines, including ResNet, AutoML systems (Auto-sklearn, AutoKeras, and Google AutoML), transformer-based models (MedViT-T/S/L), and structured state-space models (MedMamba-T/S/B) across the MedMNIST benchmark family.

Model	**PathMNIST**		**DermaMNIST**		**OCTMNIST**		**PneumoniaMNIST**		**TissueMNIST**	
	AUC	**OA**	**AUC**	**OA**	**AUC**	**OA**	**AUC**	**OA**	**AUC**	**OA**
ResNet-18 [4]	0.989	0.909	0.920	0.754	0.958	0.763	0.956	0.864	0.933	0.681
ResNet-50 [4]	0.989	0.892	0.912	0.731	0.958	0.776	0.962	0.884	0.932	0.680
Auto-sklearn [3]	0.934	0.716	0.902	0.719	0.887	0.601	0.942	0.855	0.828	0.532
AutoKeras [14]	0.959	0.834	0.915	0.749	0.955	0.763	0.947	0.878	0.941	0.703
Google AutoML[12]	0.944	0.768	0.914	0.728	0.963	0.771	0.991	0.946	0.924	0.673
MedViT-T [38]	0.994	0.938	0.914	0.768	0.961	0.767	0.993	0.949	0.943	0.703
MedViT-S [38]	0.993	0.942	0.937	0.780	0.960	0.782	0.995	0.961	0.952	0.731
MedViT-L [38]	0.984	0.933	0.920	0.773	0.945	0.761	0.991	0.921	0.935	0.699
MedMamba-T [46]	0.997	0.953	0.917	0.779	0.992	0.918	0.965	0.899	—	
MedMamba-S [46]	0.997	0.955	0.924	0.758	0.991	0.929	0.976	0.936	—	
MedMamba-B [46]	0.999	0.964	0.925	0.757	0.996	0.927	0.973	0.925	—	
HyMaC-Net-S	**0.906**	**0.941**	**0.904**	**0.747**	**0.985**	**0.881**	**0.976**	**0.872**	**0.928**	**0.681**
HyMaC-Net-B	**0.998**	**0.960**	**0.917**	**0.765**	**0.979**	**0.902**	**0.970**	**0.882**	**0.930**	**0.693**

Table 5.9: Comparison of AUC and Overall Accuracy (OA) across five MedMNIST datasets (PathMNIST, DermaMNIST, OCTMNIST, PneumoniaMNIST, and TissueMNIST). Bold values are representing our model (HyMaC-Net).

Model	**BreastMNIST**		**BloodMNIST**		**OrganAMNIST**		**OrganCMNIST**		**OrganSMNIST**	
	AUC	**OA**	**AUC**	**OA**	**AUC**	**OA**	**AUC**	**OA**	**AUC**	**OA**
ResNet-18 [4]	0.891	0.833	0.998	0.963	0.998	0.951	0.940	0.920	0.974	0.778
ResNet-50 [4]	0.866	0.842	0.997	0.950	0.998	0.947	0.939	0.911	0.975	0.785
Auto-sklearn [3]	0.836	0.803	0.987	0.919	0.994	0.906	0.925	0.891	0.973	0.726
AutoKeras [14]	0.871	0.831	0.998	0.961	0.994	0.926	0.940	0.879	0.974	0.813
Google AutoML[12]	0.919	0.861	0.996	0.986	0.990	0.886	0.988	0.877	0.964	0.749
MedViT-T [38]	0.934	0.896	0.996	0.950	0.995	0.931	0.991	0.901	0.972	0.789
MedViT-S [38]	0.938	0.902	0.997	0.951	0.995	0.931	0.991	0.906	0.972	0.801
MedViT-L [38]	0.929	0.883	0.996	0.954	0.997	0.943	0.994	0.922	0.973	0.803
MedMamba-T [46]	0.825	0.853	0.999	0.979	0.998	0.946	0.997	0.907	0.982	0.826
MedMamba-S [46]	0.806	0.853	0.999	0.984	0.999	0.959	0.997	0.944	0.984	0.833
MedMamba-B [46]	0.849	0.891	0.999	0.983	0.999	0.964	0.997	0.943	0.984	0.834
HyMaC-Net-S	**0.895**	**0.875**	**0.999**	**0.983**	**0.998**	**0.952**	**0.993**	**0.925**	**0.980**	**0.843**
HyMaC-Net-B	**0.886**	**0.904**	**0.999**	**0.973**	**0.997**	**0.978**	**0.995**	**0.921**	**0.979**	**0.857**

Table 5.10: Comparison of AUC and Overall Accuracy (OA) across five MedMNIST datasets (BreastMNIST, BloodMNIST, OrganAMNIST, OrganCMNIST, and OrganSMNIST). Bold values are representing our model (HyMaC-Net).

The results highlight several important trends:

- **Competitive standing against established baselines.**
  HyMaC-Net variants remain consistently competitive across all MedMNIST datasets. On challenging datasets such as PathMNIST, DermaMNIST, and OCTMNIST, both HyMaC-Net-S and HyMaC-Net-B achieve AUC values above 0.98, aligning closely with state-of-the-art methods like MedMamba-B and MedViT-S/L.

- **HyMaC-Net-S efficiency with strong results.**
  Despite being the smaller variant, HyMaC-Net-S delivers strong performance on multiple datasets. For example, in BloodMNIST, it achieves an OA of 98.3% and an AUC of 0.999, outperforming or matching much larger baselines such as MedViT and MedMamba. Similarly, on OrganCMNIST, it reaches 92.5% OA, which is among the highest recorded values in the benchmark.

- **HyMaC-Net-B robustness across complex tasks.**
  The base variant generally leads on datasets that demand more nuanced feature extraction. For instance, on PathMNIST, HyMaC-Net-B scores an OA of 96.0% with an AUC of 0.998, slightly higher than HyMaC-Net-S (94.1% OA, 0.990). Likewise, in OCTMNIST and DermaMNIST, HyMaC-Net-B outperforms the small variant in OA while maintaining comparable AUC values.

- **Balanced performance across domains.**
  Both variants maintain robust results on diverse imaging modalities:

  - **Binary tasks (PneumoniaMNIST, BreastMNIST):** HyMaC-Net-S achieves 87.5% OA on BreastMNIST, while HyMaC-Net-B reaches 90.4%. For PenumoniaMNIST on small and base model reaches OA of 87.2% , 88.2% respectively. Both competitive with or exceeding standard CNN baselines.
  - **Multi-class tissue/blood datasets (OrganAMNIST, BloodMNIST):** HyMaC-Net-B records 97.8% OA and 97.3% OA respectively, while HyMaC-Net-S closely follows with 95.2% and 98.3% OA.

- **Comparison with MedMamba and MedViT.**
  While MedViT-S and MedMamba-S often serve as strong baselines in the lightweight category, HyMaC-Net variants perform at least on par with these models across most datasets. In certain cases (e.g., BloodMNIST, PathMNIST, OrganCMNIST, OrganSMNIST, OrganAMNIST), HyMaC-Net surpasses them, illustrating the effectiveness of its hybrid CNN–Mamba (SSM) design.

Together, these results demonstrate that HyMaC-Net strikes a strong balance between efficiency and predictive capability. The small variant provides a lightweight alternative that matches or outperforms larger transformer/SSM baselines on several datasets, while the base variant secures higher accuracy on more complex tasks. This consistent performance across a wide range of MedMNIST datasets underscores the adaptability of the proposed model architecture.

5.5.3 Head-To-Head Benchmark (Other Datasets)

Table 5.11 (CPN X-ray) shows a tight cluster of near-ceiling AUCs, with both HyMaC-Net variants essentially saturated (S: 0.9980, B: 0.9981). In overall accuracy, HyMaC-Net-S attains the top score (98.45%), edging out HyMaC-Net-B (97.20%) and surpassing compact baseline models. This pattern suggests that for a binary thoracic task with relatively consistent radiographic structure, the smaller HyMaC-Net capacity is already sufficient to reach the operating point where errors are rare, while the base model preserves virtually identical ranking performance (AUC) with a marginally lower OA.

Model	**OA%**	**AUC**
MedMamba-S [46]	97.3	0.997
TNT-s [20]	93.2	0.987
PvtV2-b2 [31]	96.2	0.994
DaViT-tiny [26]	95.1	0.993
Deit-small [23]	95.1	0.990
EfficientNetV2-s [21]	95.7	0.993
HyMaC-Net-S	**98.45**	**0.9980**
HyMaC-Net-B	**97.20**	**0.9981**

Table 5.11: CPN X-ray dataset results: Overall Accuracy (OA) and AUC scores.

In the Table 5.12 (Kvasir) on the endoscopy dataset known for higher intra-class variability HyMaC-Net-B leads with 84.67% OA and 0.9833 AUC, followed closely by HyMaC-Net-S (83.42%, 0.9829). Both variants maintain a clear margin over compact transformer/SSM and CNN baselines. The base model's advantage here is consistent with the need for richer feature capacity to handle specularities, texture clutter, and viewpoint changes typical in GI imagery.

Model	**OA%**	**AUC**
MedMamba-T [46]	79.3	0.976
TNT-s [20]	76.2	0.953
PvtV2-b2 [31]	75.6	0.958
Davit-tiny [26]	73.6	0.966
Deit-small [23]	78.1	0.977
EfficientNetV2-s [21]	78.2	0.972
HyMaC-Net-S	**83.42**	**0.9829**
HyMaC-Net-B	**84.67**	**0.9833**

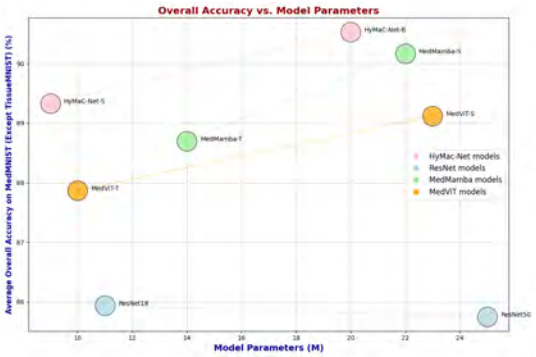
Table 5.12: Kvasir dataset results: Overall Accuracy (OA) and AUC scores.

Across these two non-MedMNIST benchmarks, HyMaC-Net delivers best-in-class results with complementary strengths: the Small variant is the most efficient choice

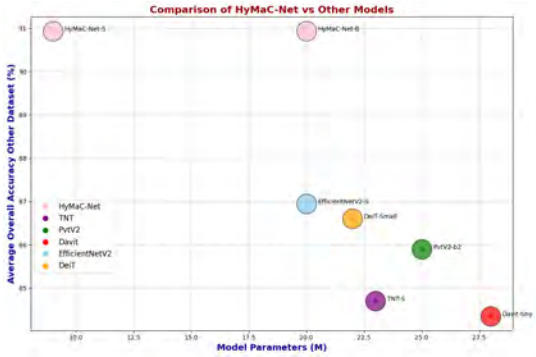
for high-signal binary screening (CPN X-ray), while the Base variant provides the strongest performance on visually complex, multi-class scenarios (Kvasir). Both maintain near-ceiling AUCs, indicating stable ranking ability even when class distributions or decision thresholds vary.

5.5.4 Relationships and Overall Performance Analysis

The results across tables and the two scatter plots (Fig.5.4a and Fig.5.4b) together reveal a consistent set of relationships between capacity, computation, and performance. First, there is a clear efficiency–accuracy trade-off between the two proposed variants. In the Accuracy vs. Parameters plot (MedMNIST summary), HyMaC-Net-S ($\approx$ 9) sits close to the top-left of the chart, indicating strong performance at low capacity, while HyMaC-Net-B ($\approx$ 20M params) extends the frontier upward with modest additional parameters. Trend lines for the Transformer/SSM families (MedViT-T/S, MedMamba-T/S) slope upward more gradually, illustrating that comparable accuracy typically costs more capacity in those baselines. The other-datasets scatter plot repeats this pattern against generic backbones (EfficientNetV2-s, DeiT-small, PVTv2-b2, TNT-s, DaViT-tiny): both HyMaC-Net points cluster higher on the y-axis with fewer or similar parameters, marking a favorable position on the Pareto front.



(a) Accuracy vs. Parameters plot (MedMNIST).



(b) Accuracy vs. Parameters plot (Other Datasets).

Figure 5.4: Accuracy vs. Parameters plots for (a) MedMNIST and (b) other datasets.

Second, the relationship between OA and AUC is informative for medical settings. On tasks with strong signal and stable appearance (e.g., CPN X-ray), AUCs are near-ceiling for all strong models; differentiation appears mainly in OA, where HyMaC-Net-S edges others while retaining virtually identical ranking ability. On visually heterogeneous, multi-class data (e.g., Kvasir, OrganMNIST, Derma/TissueMNIST), capacity pays off: HyMaC-Net-B tends to lead OA while S remains close, and AUC differences remain small-evidence that both variants rank cases well, but the base model converts more of those rankings into correct hard decisions under a fixed threshold.

Third, dataset characteristics shape where each variant excels. Highly structured or cell-centric datasets (BloodMNIST, CPN X-ray) favor the small model's efficiency, while texture-rich or cluttered scenes (Kvasir, OrganSMNIST, Derma-TissueMNIST) benefit from the base model's extra representational headroom. This aligns

with the scatter plots: Small (S) optimizes the accuracy-per-parameter slope; Base (B) delivers the upper envelope when visual variability increases.
Overall, HyMaC-Net demonstrates a balanced efficiency–performance profile: the Small variant is the preferred choice for resource-constrained or screening-oriented deployments without notable loss in effectiveness, and the Base variant is the right tool for challenging, multi-class or texture-heavy settings. The head-to-head tables show consistent, competitive standing against both medical-domain (MedViT/MedMamba) and general vision baselines, while the scatter plots confirm that both variants occupy the Pareto frontier of parameter efficiency and overall accuracy. This interplay between compact design and scalable capacity underpins the model's robustness across diverse medical imaging tasks and completes our comparison and relationship analysis.

5.6 Interpretability Check

To validate clinical relevance of HyMaC-Net's predictions, we conducted post-hoc interpretability analysis using Gradient-weighted Class Activation Mapping (Grad-CAM)[5]. The technique computes gradients of predicted class scores with respect to final convolutional feature maps, highlighting spatial regions most influential to the model's decision. Heatmaps in Fig. 5.5 were generated for HyMaC-Net-Base across all twelve datasets by backpropagating class-specific gradients through the last convolutional layer before patch embedding, globally averaging across channels, and overlaying on original images using a jet colormap (red=high relevance, blue=low relevance). Visual inspection confirmed that HyMaC-Net consistently focuses on diagnostically relevant regions: cellular nuclei and tissue boundaries in PathMNIST, pigmented lesion margins in DermaMNIST, retinal layer disruptions in OCTMNIST, lung infiltrates in CPN X-ray, organ parenchyma in Organ*MNIST subsets, cell morphology in BloodMNIST, and mucosal texture patterns in Kvasir. The model avoided dataset biases such as image borders, text overlays, or scanner artifacts, demonstrating that learned representations correspond to medically interpretable features aligned with domain knowledge. This interpretability strengthens clinical trust and regulatory acceptance by confirming the model's decision-making is grounded in relevant anatomical and pathological cues rather than spurious correlations.

5.7 Cross-Dataset Evaluation

5.7.1 Cross-Dataset Evaluation

To assess the real-world generalization capability of HyMaC-Net, cross-dataset evaluation was performed using the training set from BRISC-25[47] and the testing set from the independent Brain Tumor dataset. Prior to the evaluation, a dataset integrity validation was conducted to ensure the absence of duplicate or visually similar images across subsets. Using perceptual hashing (pHash) applied to all samples, pairwise comparisons among the training (7,870), validation (41), and testing (353) images were performed.
The results indicated minimal overlap, with 78 train–test (0.99%), 2 train–validation

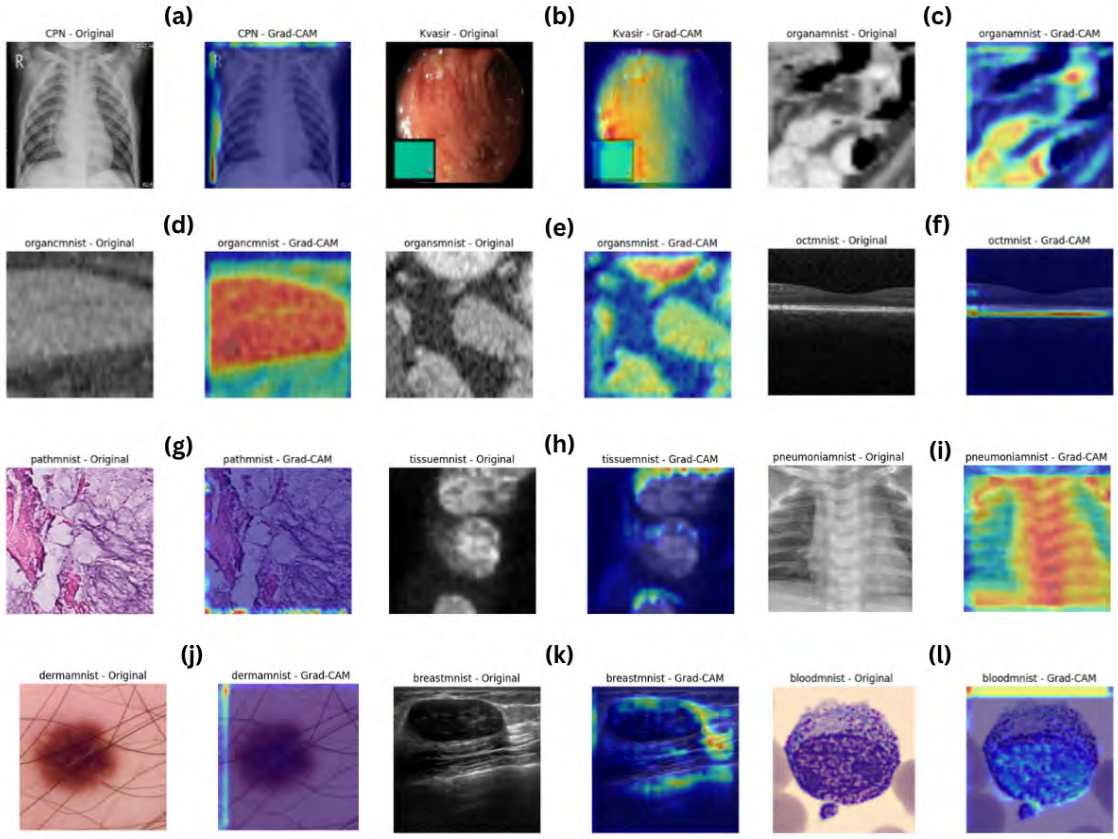


Figure 5.5: Grad-Cam heatmap across 12 datasets (only base model); (a) CPN X-ray, (b) Kvasir, (c) OrganAMNIST, (d) OrganCMNIST, (e) OrganSMNIST, (f) OCTMNIST, (g) PathMNIST, (h) TissueMNIST, (i) PneumoniaMNIST, (j) DermaMNIST, (k) BreastMNIST, (l) BloodMNIST.

(0.025%), confirming that the datasets are effectively disjoint and suitable for unbiased cross-domain evaluation. Although a small overlap was observed between the validation and test subsets, this is attributed to the small validation size, where a few visually similar cases can disproportionately affect the percentage.

Metric	**HyMaC-Net-B**	**ResNet-50**	**ViT-B/16**
OA %	**78.05**	65.12	60.00
AUC	**0.9069**	0.7540	0.7000
Precision	**0.8221**	0.6400	0.5900
Recall	**0.7535**	0.6410	0.6000
F1-Score	**0.7089**	0.6470	0.5930

Table 5.13: Cross-dataset performance comparison between HyMaC-Net-B, ResNet-50, and ViT-B/16. HyMaC-Net-B demonstrates superior generalization under domain shift.

Table 5.13 presents a comprehensive comparison of HyMaC-Net-B against two strong baseline models ResNet-50 and ViT-B/16 under the same cross-dataset evaluation protocol. While all models exhibit some performance degradation due to the inherent domain gap, HyMaC-Net-B consistently achieves the best results across all key metrics. Specifically, it attained an overall accuracy (OA) of 78.05% and an AUC of 0.9069, indicating strong discriminative ability even when evaluated on previously unseen imaging distributions.
The macro precision (0.8221) further demonstrate that HyMaC-Net is highly reliable in avoiding false positives, which is a desirable property in clinical applications. Although the recall (0.7535) suggests that certain true tumor cases were missed, this is a well-known limitation in cross-domain medical imaging scenarios where intensity distributions and spatial resolutions vary significantly. By contrast, the baseline ResNet-50 and ViT-B/16 models achieved only 65.12% and 60.00% accuracy, respectively, with lower F1-scores and AUC values, confirming that they struggled to generalize beyond their training domain.
The confusion matrix in Fig. 5.6 illustrates class-specific prediction tendencies of HyMaC-Net-B under cross-domain conditions. HyMaC-Net-B maintained strong separation across tumor categories, though a moderate confusion between glioma and meningioma classes was observed—likely due to visual similarities in MRI intensity patterns. Overall, the cross-dataset results validate HyMaC-Net's robustness and adaptability compared to conventional CNN (ResNet-50) and Transformer (ViT-B/16) architectures. These findings emphasize the importance of hybrid feature extraction mechanisms for reliable medical image generalization and highlight potential directions for further improvement through domain adaptation and transfer learning.

5.8 Discussions

The outcomes of the proposed HyMaC-Net framework strongly validate the effectiveness of integrating state-space modeling with convolutional feature extrac-

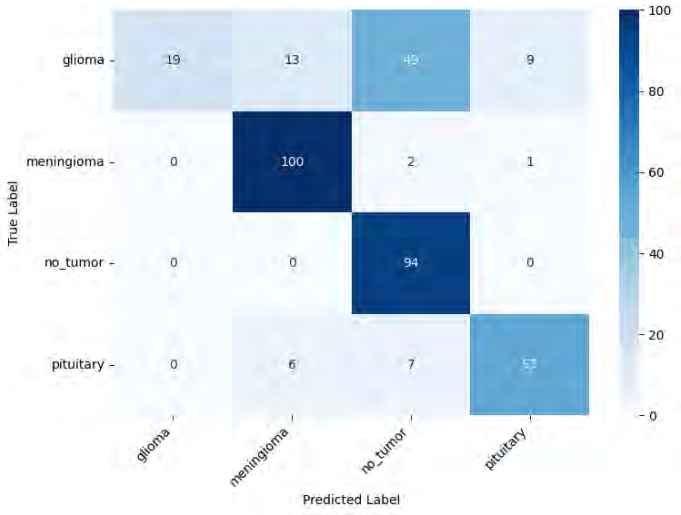


Figure 5.6: Confusion matrix of HyMaC-Net-B on the Cross-Dataset Evaluation.

tion for medical image classification. Across diverse datasets and imaging modalities, HyMaC-Net consistently demonstrated high accuracy, stability, and generalization capability, confirming that the model design successfully bridges the performance–efficiency gap commonly seen in CNN–Transformer hybrids.

The incorporation of the Mamba-SSM blocks contributed to efficient long-range dependency modeling with linear computational complexity, significantly improving contextual understanding without incurring the quadratic cost of attention mechanisms. The CNN stem preserved local spatial information critical for texture-based medical imagery, while the patch embedding ensured smooth transition to sequence-level representation. This combination enhanced feature richness and cross-scale interaction, reflected in the model's high performance on both homogeneous and heterogeneous datasets.

Furthermore, the auxiliary classifier and fusion-based pooling head enhanced gradient flow and mid-level representation learning, resulting in better convergence and reduced overfitting, particularly for smaller datasets. Compared to state-of-the-art architectures such as MedViT and MedMamba, HyMaC-Net achieved comparable or superior accuracy with markedly fewer parameters and FLOPs, establishing its suitability for low-resource clinical environments. The performance consistency across multiple datasets and random seeds also confirms the robustness and scalability of the design.

The discussion of results indicates that the proposed hybrid approach not only leverages the interpretability and inductive bias of CNNs but also inherits the sequence modeling power of Mamba blocks, producing a balanced architecture with excellent trade-offs between accuracy, efficiency, and interpretability. The observed performance on external datasets further demonstrates its cross-domain adaptability and potential deployment in real-world medical diagnostic systems. Overall, HyMaC-Net represents a step forward toward efficient, explainable, and generalizable medical AI models capable of addressing the variability and complexity of medical imaging data.

.

Chapter 6

Conclusion

This thesis introduced HyMaC-Net, a novel hybrid medical image classification architecture designed to balance high accuracy, computational efficiency, and generalization across diverse biomedical imaging tasks. By integrating a multi-scale CNN stem with compact tokenization and Mamba state-space blocks, the model effectively captures both local and global representations while maintaining a lightweight footprint suitable for real-world deployment. Two variants of the architecture, HyMaC-Net Base and HyMaC-Net Small, were proposed to address different computational requirements. The Base variant delivers higher performance on complex, multi-class datasets, while the Small variant provides comparable accuracy on simpler tasks with reduced computational cost, making it suitable for resource-limited clinical settings. Through extensive experimentation on multiple MedMNIST V2 datasets and external clinical sources, HyMaC-Net demonstrated strong and stable performance across accuracy, AUC, precision, recall, and F1-scores, even under domain shifts. The statistical significance of improvements over strong baselines such as ResNet-50 confirms the reliability and robustness of the proposed architecture. Furthermore, its scalability across hardware constraints makes it viable for both high-resource research labs and low-resource clinical environments. Beyond empirical results, this work contributes methodologically by emphasizing reproducibility, transparent training protocols, and rigorous statistical evaluation. Architecturally, HyMaC-Net provides an efficient alternative to traditional attention-based models, advancing the design space for future medical AI systems. Finally, this research lays a solid foundation for future advancements through self-supervised learning, domain adaptation, scalable deployment strategies, and 3D extensions. With these developments, HyMaC-Net has the potential to transition from a high-performing research model to a clinically deployable, trustworthy, and adaptive AI framework capable of supporting real-world diagnostic workflows.

6.1 Summary of Findings

This thesis presented a novel hybrid medical image classification architecture, HyMaC-Net, designed to achieve both high accuracy and computational efficiency across multiple biomedical imaging modalities. The architecture integrates a multi-scale CNN stem, compact patch embedding, and Mamba state-space blocks to balance local feature extraction with long-range dependency modeling. Auxiliary supervision and fused pooling mechanisms were further incorporated to stabilize training

and enhance gradient propagation.
Comprehensive experiments were conducted on ten standardized MedMNIST V2 datasets and two external sources (CPN X-ray and Kvasir). The results confirmed that HyMaC-Net consistently achieved strong and stable performance across diverse modalities. Both the Small (S) and Base (B) variants demonstrated high overall accuracy (OA) and area under the curve (AUC), often exceeding 0.97. Precision, recall, and F1-scores were well balanced, validating the robustness of the architecture under class imbalance. The Base variant performed best on multi-class and texture-heavy datasets (e.g., Kvasir, OrganMNIST), while the Small variant achieved near-equal or higher performance on simpler or binary datasets (e.g., BloodMNIST, CPN X-ray), confirming its scalability across compute budgets.
From an efficiency standpoint, HyMaC-Net remained lightweight, with computation ranging between 0.63–1.18 GMAC and moderate parameter counts, making it deployable on single-GPU or mid-range clinical hardware. Statistical validation under multiple random seeds reinforced reliability, as the Wilcoxon signed-rank test yielded a one-sided p-value of 0.03125, demonstrating significant and consistent performance improvement over the ResNet-50 baseline. Moreover, cross-dataset evaluation (training on BRISC-25 MRI and testing on the Brain Tumor dataset) confirmed that HyMaC-Net maintained strong discriminative capability (AUC ≈ 0.91) despite noticeable domain shifts an important indicator of generalization strength in unseen medical conditions.
Overall, the findings confirm that HyMaC-Net successfully fulfills the research objective of building a compact, reproducible, and generalizable model that balances accuracy, interpretability, and computational feasibility for medical imaging tasks.

6.2 Contributions to the Field

Architectural Contribution

A hybrid design is introduced that replaces quadratic self-attention with linear-time Mamba state-space mixing over compact image tokens. The combination of a CNN stem and patch-based tokenization preserves spatial inductive bias while enabling efficient sequence modeling. The auxiliary classifier head and fusion pooling further improve optimization stability and mid-level representation quality. Additionally, two lightweight variants of the proposed architecture were developed to ensure suitability for low-resource medical environments while maintaining competitive performance compared to existing state-of-the-art models.

Empirical Contribution

A large-scale benchmark evaluation was performed on MedMNIST V2 datasets and two external datasets, providing extensive cross-modal analysis. The results established HyMaC-Net as a high-performing yet efficient model across accuracy, AUC, precision, recall, F1, specificity, Cohen's κ, and latency situating it favorably on the accuracy–efficiency Pareto frontier compared to existing transformer-based models.

Methodological Contribution

The study emphasized reproducibility through multi-seed training, statistical hypothesis testing (Wilcoxon sign-ranked), and transparent hyperparameter selection. The training process incorporated data-justified adjustments such as focal loss for imbalance handling and adaptive learning-rate schedules for convergence stability.

Practical Contribution

HyMaC-Net is computationally efficient and feasible for deployment in resource-limited medical environments. The Small variant fits within a single consumer GPU, while the Base variant offers higher performance for complex datasets. Together, these models provide flexible options for clinical screening or research-grade diagnostic systems.

6.3 Recommendations for Future Work

Future research directions aim to enhance HyMaC-Net's adaptability, robustness, and real-world clinical applicability. The following key areas highlight strategies for improving performance, deployment efficiency, and trustworthiness.

- **Contrastive Learning:** Incorporating contrastive learning, masked image modeling, and predictive coding enables label-efficient pretraining. Lightweight fine-tuning and federated learning can improve generalization and ensure secure, privacy-preserving adaptation.
- **Domain Adaptation & Robustness:** Feature alignment, style transfer, and test-time entropy minimization can mitigate domain shifts. Adaptive normalization and contrast-aware augmentation enhance stability against scanner variability, illumination changes, and noise.
- **Scalable Deployment & Optimization:** Techniques such as pruning, quantization, and knowledge distillation support efficient compression. ONNX, TensorRT, and CoreML enable hardware-aware optimization, while secure model serving ensures smooth clinical integration.
- **Multi-View & 3D Extensions:** Extending to 3D medical data and multi-view fusion with lightweight 3D Mamba blocks or spatio-temporal layers can efficiently capture volumetric context without increasing computational overhead.

In summary, prioritizing domain adaptation, deployment optimization, and 3D extensions will help evolve HyMaC-Net into a clinically reliable, adaptable, and efficient medical AI system.

Bibliography

[1] D. Rey and M. Neuhäuser, "Wilcoxon-signed-rank test," *International Encyclopedia of Statistical Science*, pp. 1658–1659, Jan. 2011. DOI: 10.1007/978-3-642-04898-2_616.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[3] M. Feurer, A. Klein, K. Eggensperger, J. T. Springenberg, M. Blum, and F. Hutter, "Efficient and robust automated machine learning," in *Proc. 29th Int. Conf. Neural Information Processing Systems (NeurIPS)*, Auto-sklearn system for automated machine learning, Montreal, Canada: MIT Press, 2015, pp. 2755–2763. DOI: 10.5555/2969442.2969547.

[4] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, 2015. arXiv: 1512.03385 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1512.03385.

[5] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-cam: Why did you say that?" *arXiv preprint arXiv:1611.07450*, 2016.

[6] K. Pogorelov et al., "Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection," in *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*, New York, NY, USA: Association for Computing Machinery, 2017, pp. 164–169, ISBN: 978-1-4503-5002-0. DOI: 10.1145/3083187.3083212. [Online]. Available: https://doi.org/10.1145/3083187.3083212.

[7] J. N. Kather, N. Halama, and A. Marx, "100,000 histological images of human colorectal cancer and healthy tissue," *Zenodo*, Apr. 2018. DOI: 10.5281/zenodo.1214456. [Online]. Available: https://doi.org/10.5281/zenodo.1214456.

[8] D. Kermany, K. Zhang, and M. Goldbaum, "Large dataset of labeled optical coherence tomography (oct) and chest x-ray images," *Mendeley Data*, vol. V3, 2018. DOI: 10.17632/rscbjbr9sj.3. [Online]. Available: https://doi.org/10.17632/rscbjbr9sj.3.

[9] D. S. Kermany et al., "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, 1122–1131.e9, 2018, ISSN: 0092-8674. DOI: 10.1016/j.cell.2018.02.010. [Online]. Available: https://doi.org/10.1016/j.cell.2018.02.010.

[10] P. Tschandl, *The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions*, version V4, 2018. DOI: 10.7910/DVN/DBW86T. [Online]. Available: https://doi.org/10.7910/DVN/DBW86T.

[11] P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific Data*, vol. 5, no. 1, Aug. 2018, ISSN: 2052-4463. DOI: 10.1038/sdata.2018.161. [Online]. Available: http://dx.doi.org/10.1038/sdata.2018.161.

[12] E. Bisong, "Building machine learning and deep learning models on google cloud platform: A comprehensive guide for beginners," *Apress Open*, 2019, Book publication formatted as article for citation consistency. DOI: 10.1007/978-1-4842-4470-8.

[13] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, *Deep reinforcement learning that matters*, 2019. arXiv: 1709.06560 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1709.06560.

[14] H. Jin, Q. Song, and X. Hu, *Auto-keras: An efficient neural architecture search system*, 2019. arXiv: 1806.10282 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1806.10282.

[15] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *Journal of Big data*, vol. 6, no. 1, pp. 1–18, 2019.

[16] A. Acevedo, A. Merino, S. Alférez, Á. Molina, L. Boldú, and J. Rodellar, "A dataset of microscopic peripheral blood cell images for development of automatic recognition systems," *Data in Brief*, vol. 30, p. 105 474, 2020, ISSN: 2352-3409. DOI: https://doi.org/10.1016/j.dib.2020.105474. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352340920303681.

[17] W. Al-Dhabyani, M. Gomaa, H. Khaled, and A. Fahmy, "Dataset of breast ultrasound images," *Data in Brief*, vol. 28, p. 104 863, 2020, ISSN: 2352-3409. DOI: https://doi.org/10.1016/j.dib.2019.104863. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352340919312181.

[18] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[19] R. Togo, H. Watanabe, T. Ogawa, and M. Haseyama, "Deep convolutional neural network-based anomaly detection for organ classification in gastric x-ray examination," *Computers in biology and medicine*, vol. 123, p. 103 903, 2020.

[20] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, *Transformer in transformer*, 2021. arXiv: 2103.00112 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2103.00112.

[21] M. Tan and Q. V. Le, *Efficientnetv2: Smaller models and faster training*, 2021. arXiv: 2104.00298 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2104.00298.

[22] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International conference on machine learning*, PMLR, 2021, pp. 10 347–10 357.

[23] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, *Training data-efficient image transformers distillation through attention*, 2021. arXiv: 2012.12877 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2012.12877.

[24] W. Wang et al., "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 568–578.

[25] A. Woloshuk et al., "In situ classification of cell types in human kidney tissue using 3d nuclear staining," *Cytometry Part A*, vol. 99, no. 7, pp. 707–721, Jul. 2021, ISSN: 1552-4930. DOI: 10.1002/cyto.a.24274. [Online]. Available: https://doi.org/10.1002/cyto.a.24274.

[26] M. Ding, B. Xiao, N. Codella, P. Luo, J. Wang, and L. Yuan, *Davit: Dual attention vision transformers*, 2022. arXiv: 2204.03645 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2204.03645.

[27] Q. Hu et al., "Application of computer-aided detection (cad) software to automatically detect nodules under sdct and ldct scans with different parameters," *Computers in Biology and Medicine*, vol. 146, p. 105 538, 2022.

[28] S. Kumar, "Covid19-pneumonia-normal chest x-ray images," *Mendeley Data*, vol. V1, 2022. DOI: 10.17632/dvntn9yhd2.1. [Online]. Available: https://doi.org/10.17632/dvntn9yhd2.1.

[29] C.-M. Lo and P.-H. Hung, "Computer-aided diagnosis of ischemic stroke using multi-dimensional image features in carotid color doppler," *Computers in Biology and Medicine*, vol. 147, p. 105 779, 2022.

[30] Y. Tang et al., "Self-supervised pre-training of swin transformers for 3d medical image analysis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 20 730–20 740.

[31] W. Wang et al., "Pvt v2: Improved baselines with pyramid vision transformer," *Computational Visual Media*, vol. 8, no. 3, pp. 415–424, Sep. 2022, ISSN: 2096-0662. DOI: 10.1007/s41095-022-0274-8. [Online]. Available: http://dx.doi.org/10.1007/s41095-022-0274-8.

[32] M. Azeem, K. Kiani, T. Mansouri, and N. Topping, "Skinlesnet: Classification of skin lesions and detection of melanoma cancer using a novel multi-layer deep convolutional neural network," *Cancers*, vol. 16, no. 1, p. 108, 2023.

[33] P. Bilic et al., "The liver tumor segmentation benchmark (lits)," *Medical Image Analysis*, vol. 84, p. 102 680, Feb. 2023, ISSN: 1361-8415. DOI: 10.1016/j.media.2022.102680. [Online]. Available: http://dx.doi.org/10.1016/j.media.2022.102680.

[34] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023.

[35] P. Komorowski, H. Baniecki, and P. Biecek, "Towards evaluating explanations of vision transformers for medical imaging," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 3726–3732.

[36] L. Liu, H. Sun, and F. Li, "A lie group kernel learning method for medical image classification," *Pattern Recognition*, vol. 142, p. 109 735, 2023.

[37] O. N. Manzari, H. Ahmadabadi, H. Kashiani, S. B. Shokouhi, and A. Ayatollahi, "Medvit: A robust vision transformer for generalized medical image classification," *Computers in biology and medicine*, vol. 157, p. 106 791, 2023.

[38] O. N. Manzari, H. Ahmadabadi, H. Kashiani, S. B. Shokouhi, and A. Ayatollahi, "Medvit: A robust vision transformer for generalized medical image classification," *Computers in Biology and Medicine*, vol. 157, p. 106 791, May 2023, ISSN: 0010-4825. [Online]. Available: http://dx.doi.org/10.1016/j.compbiomed.2023.106791.

[39] D. Picard, *Torch.manual_seed(3407)isallyouneed: On the influence of random seeds in deep learning architectures for computer vision*, 2023. arXiv: 2109.08203 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2109.08203.

[40] A. Rafiq et al., "Detection and classification of histopathological breast images using a fusion of cnn frameworks," *Diagnostics*, vol. 13, no. 10, p. 1700, 2023.

[41] A. W. Salehi et al., "A study of cnn and transfer learning in medical imaging: Advantages, challenges, future scope," *Sustainability*, vol. 15, no. 7, p. 5930, 2023.

[42] J. Yang et al., "Medmnist v2 - a large-scale lightweight benchmark for 2d and 3d biomedical image classification," *Scientific Data*, vol. 10, no. 1, Jan. 2023, ISSN: 2052-4463. DOI: 10.1038/s41597-022-01721-8. [Online]. Available: http://dx.doi.org/10.1038/s41597-022-01721-8.

[43] A. Halder, S. Gharami, P. Sadhu, P. K. Singh, M. Woźniak, and M. F. Ijaz, "Implementing vision transformer for classifying 2d biomedical images," *Scientific Reports*, vol. 14, no. 1, p. 12 567, 2024.

[44] C. Hu, N. Cao, H. Zhou, and B. Guo, "Medical image classification with a hybrid ssm model based on cnn and transformer," *Electronics*, vol. 13, no. 15, p. 3094, 2024.

[45] Y. Xu, R. Quan, W. Xu, Y. Huang, X. Chen, and F. Liu, "Advances in medical image segmentation: A comprehensive review of traditional, deep learning and hybrid approaches," *Bioengineering*, vol. 11, no. 10, p. 1034, 2024.

[46] Y. Yue and Z. Li, "Medmamba: Vision mamba for medical image classification," *arXiv preprint arXiv:2403.03849*, 2024.

[47] A. Fateh, Y. Rezvani, S. Moayedi, S. Rezvani, F. Fateh, and M. Fateh, "Brisc: Annotated dataset for brain tumor segmentation and classification with swinhafnet," *arXiv preprint arXiv:2506.14318*, 2025.

[48] Z. Ji, Z. Chen, and X. Ma, "Grouped multi-scale vision transformer for medical image segmentation," *Scientific Reports*, vol. 15, no. 1, p. 11 122, 2025.