

Article

MedMamba-ESS: A Generalizable State Space Framework for Clinical Image Classification with Enhanced Performance and Computational Efficiency

Huajuan Gao ¹, Yuxin Zhang ¹, Kun Hu ¹, Tingting Leng ¹, Zhuolun He ², Zhenzhen Lin ¹, Antong Deng ¹ and Jiong Mu ^{1,*}

- ¹ College of Information Engineering, Sichuan Agricultural University, Ya'an 625014, China; 202308537@stu.sicau.edu.cn (H.G.); 202308278@stu.sicau.edu.cn (Y.Z.); 202308593@stu.sicau.edu.cn (K.H.); 202308630@stu.sicau.edu.cn (T.L.); 202308391@stu.sicau.edu.cn (Z.L.); 202308566@stu.sicau.edu.cn (A.D.);
² College of Mechanical and Electrical Engineering, Sichuan Agricultural University, Ya'an 625014, China; 202305714@stu.sicau.edu.cn
* Correspondence: jmu@sicau.edu.cn

Abstract

Since deep learning has been applied to medical image analysis, convolutional neural networks (CNNs) and Vision Transformers (ViTs) have become core architectures for medical image classification tasks. However, CNNs are constrained by local receptive fields and cannot effectively model long-range context-dependent correlations in medical images; ViTs face deployment challenges in high-resolution medical image tasks due to the quadratic computational complexity of self-attention. While State Space Models (SSMs) represented by Mamba offer a new solution with linear complexity, they suffer from redundant directional modeling and low parameter efficiency in direct medical image applications. This study proposes the efficient state space enhancement framework (MedMamba-ESS), integrating the SS2D-Top2 adaptive directional scanning mechanism (reducing SSM submodule FLOPs by ~37%) and the G-SSM grouped parameter sharing module (achieving 3–4% parameter compression and 4.4% accuracy improvement via regularization). Validated on 14 public datasets (9 imaging modalities, 9 anatomical regions, >240,000 images), MedMamba-ESS achieves superior performance at 2.0G FLOPs: its Overall Accuracy (OA) is 3% higher than the baseline MedMamba-Tiny on non-MedMNIST datasets (3.13% higher than other mainstream models) and 3% higher on MedMNIST datasets (1% higher than others). Ablation experiments confirm that the two modules synergistically reduce parameters by 1.05% and boost accuracy by 4.6%. This study overcomes the technical limitations of traditional SSMs in medical imaging applications, achieving synergistic improvements in both model performance and computational efficiency. It provides an architecture optimization solution that combines practicality and generalizability for the implementation of SSMs in medical image analysis.

Keywords: medical image classification; multiple imaging modalities; state space model; adaptive directional scanning; parameter sharing; computational efficiency

Academic Editor: Alexander Zhbanov

Received: 26 January 2026

Revised: 14 February 2026

Accepted: 23 February 2026

Published: 28 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Image classification has emerged as an indispensable component of modern precision medicine, serving a pivotal role in multiple key clinical stages—including disease screening, clinical diagnosis, treatment planning, and therapeutic outcome monitoring [1,2]. With the rapid advancement of digital imaging technologies, a diverse array of imaging modalities (e.g., computed tomography (CT), magnetic resonance imaging (MRI), ultrasound (US), X-ray, endoscopic imaging, and microscopic imaging) are now extensively utilized in clinical practice [3]. Collectively, these modalities generate large-scale, heterogeneous medical image datasets, laying a foundation for the implementation of vision-based models in automated diagnosis and analysis [4].

However, the explosive growth of medical imaging data also presents formidable challenges. On one hand, the sheer volume of imaging examinations—particularly complex modalities like CT and MRI—has dramatically increased diagnostic resource pressures on hospitals and radiology departments [5]. Clinical statistics indicate that in an 866-bed tertiary hospital, imaging examinations for inpatients alone have shown continuous growth over the past decade, significantly increasing the workload of radiologists [6]. On the other hand, significant inter-reader variability exists in image diagnosis, particularly in cases involving complex, early-stage lesions, or atypical presentations, where this uncertainty is especially pronounced [7]. Even for the same set of medical images, physicians with differing levels of expertise, experience, and diagnostic styles may arrive at divergent conclusions [8]. These challenges underscore the urgency of developing automated, standardized diagnostic tools. Such technologies aim to alleviate physician workload, reduce subjective variability, and ultimately enhance diagnostic consistency and overall efficiency.

With the rapid advancement of deep learning, convolutional neural networks (CNNs) and Vision Transformers (ViTs) have emerged as the two dominant architectures for medical image classification [9,10]. While CNNs excel at extracting local features, their inherently limited receptive fields restrict their capacity to model long-range dependencies—a limitation that is particularly detrimental to medical image analysis, where high intraclass variability and low inter-class variability and the critical need for global contextual information are hallmark characteristics [9,11]. ViTs model image patch sequences globally via self-attention mechanisms, enabling the capture of long-range dependencies. However, their computational complexity scales quadratically with image resolution, resulting in substantial computational resource consumption for high-resolution medical images and thus hindering their clinical deployment [12,13]. In recent years, State Space Models (SSMs)—notably Mamba [14]—have introduced novel solutions for medical vision tasks, owing to their exceptional linear complexity in modeling long sequences [15].

Visual backbone networks based on Mamba, like the SF-Mamba network [16], use a hybrid local-global feature fusion mechanism that makes it possible to accurately locate global contours and refine local boundaries of lesion areas. Tests on several public medical image segmentation datasets, such as skin lesions, tumors, and ultrasound images, have shown that the Mamba architecture performs well at segmentation while keeping the number of parameters very low (0.038 million). MambaVesselNet++ [17] presents a novel network model for retinal image classification tasks, integrating convolutional layers with the Mamba architecture. This model not only ensures rapid classification, but also accurately detects small lesions in fundus images. For classifying pneumonia using chest X-ray images, UM-Mamba [18] utilizes the Mamba architecture, optimizing parameter count and computational complexity while surpassing conventional methods in classification accuracy.

These studies have brought State Space Models (SSMs) into the field of medical vision, showing how useful they can be for processing medical images. Still, using SSMs directly on medical images is still hard for two main reasons. First, medical images usually have a directional bias [16], but current visual SSMs use the same method to process all directional information [17]. This one-size-fits-all approach fails to adaptively prioritize critical information based on image content, leading to excessive computational overhead. Second, although medical images exhibit significant texture differences across modalities, their long-range structural dependencies—such as global symmetry and cavity structure propagation—show high similarity [18,19]. Consequently, the parameterization approach for the state matrix within the SSM module is inefficient, particularly due to substantial redundant parameters in the channel dimension. This not only increases model complexity but may also weaken its ability to learn discriminative features, thereby hindering its generalization capabilities across diverse medical imaging modalities.

To address these challenges, this paper proposes MedMamba-ESS, an efficient framework for enhancing state space based on MedMamba [20]. First, we introduce SS2D-Top2 (State Space 2D-Top2 Adaptive Directional Scanning Mechanism), which combines scanning direction selection strategies with directional gating. This mechanism maintains computational efficiency during low-resolution phases while preserving more spatial feature details in high-resolution phases, tailored to image resolution characteristics. We also introduce the G-SSM (Grouped State Space Model), which significantly enhances parameter efficiency and reduces redundant storage of state space parameters through low-rank projection and grouped sharing strategies. Together, SS2D-Top2 and G-SSM address the fundamental issues of multi-scale feature extraction and parameter efficiency in medical SSMs. The main contributions of this paper are as follows:

1. We suggest MedMamba-ESS, a consistent and effective state space augmentation architecture. It offers a comprehensive solution for medical picture classification challenges by combining dynamic scan direction control with a parameter sharing method.
2. An adaptive directional scanning mechanism (SS2D-Top2) is designed. By regulating scanning directions, this mechanism reduces computational overhead in low-resolution stages while preserving spatial features in high-resolution stages, thereby ensuring computational efficiency and feature extraction capability across multi-resolution stages.
3. A grouped parameter sharing module (G-SSM) is introduced, which enhances parameter efficiency significantly and reduces redundant storage requirements while preserving the model's expressive capacity through low-rank projection and channel grouping strategies.
4. Experiments conducted on 14 benchmark medical image datasets demonstrate that the proposed framework significantly improves classification performance while maintaining high computational efficiency, validating the effectiveness of the synergistic operation of the two improved components.

2. Materials and Methods

2.1. Dataset Description

We comprehensively evaluated the performance of MedMamba-ESS on a benchmark consisting of 14 publicly available medical image datasets. These datasets cover 9 imaging modalities (X-ray, ultrasound, computed tomography (CT), endoscopy, optical coherence tomography (OCT), dermatoscopy, fundus photography, microscopy, and smartphone-based imaging) (As shown in Figure 1) and 9 anatomical regions (skin, chest, maternal-fetal structures, gastrointestinal tract, colon, retina, breast, blood, and ab-

dominal organs), thus constituting a large-scale benchmark with diversity, challenge, and strong clinical representativeness. Specifically:

PAD-UFES-20 [21] was collected via a variety of smartphone devices. This dataset consists of 2298 samples corresponding to six distinct types of skin lesions, namely, Basal Cell Carcinoma (BCC), Squamous Cell Carcinoma (SCC), Actinic Keratosis (ACK), Seborrheic Keratosis (SEK), Bowen's Disease (BOD), Melanoma (MEL), and Nevus (NEV).

CPN X-ray [22,23] is a publicly available dataset comprising 5228 chest X-ray images. It enables researchers and the medical community to leverage deep learning for the detection and classification of COVID-19 and pneumonia from chest X-ray images. The full dataset is divided into three categories: COVID-19, NORMAL, and Pneumonia.

Fetal-Planes-DB [24] is a large-scale dataset of routinely acquired ultrasound images from maternal-fetal screening, collected by multiple operators using different ultrasound machines across two distinct hospitals. All images were manually annotated by a specialist in maternal-fetal medicine. The dataset is categorized into six classes: four of the most widely used fetal anatomical planes (abdominal, brain, femur, and thoracic), the maternal cervix (a key region for preterm labor screening), and a general category that includes all other less common image planes. Fetal brain images are further subdivided into three of the most common fetal brain planes (trans-thalamic, trans-cerebellar, and trans-ventricular) to evaluate fine-grained classification performance.

The Kvasir dataset [25] consists of images annotated and validated by experienced endoscopic physicians, covering multiple categories that depict gastrointestinal anatomical landmarks, pathological findings, or endoscopic procedures—with hundreds of images per category. Anatomical landmarks include the Z-line, pylorus, cecum, and others, while pathological findings include esophagitis, polyps, ulcerative colitis, and others. Furthermore, Kvasir provides multiple image sets associated with lesion resection, such as “stained and lifted polyps” and “stained resection margins”.

MedMNIST [26,27] is a large-scale standardized collection of biomedical images analogous to MNIST, consisting of 12 2D datasets and 6 3D datasets. All images are associated with corresponding classification labels, obviating the requirement for domain-specific background knowledge. The image resolutions included are 28×28 (2D), 64×64 (2D), 128×128 (2D), 224×224 (2D), $28 \times 28 \times 28$ (3D), and $64 \times 64 \times 64$ (3D). MedMNIST covers major modalities of biomedical images and is designed for the classification of lightweight 2D and 3D images across diverse data scales (ranging from 100 to 100,000 samples) and task types (binary/multi-class classification, ordinal regression, and multi-label classification). Owing to its educational value, standardization, diversity, and lightweight characteristics, MedMNIST acts as a universal classification benchmark for medical image analysis. In this study, we selected ten 2D medical image classification datasets from MedMNIST: PathMNIST, OCTMNIST, DermaMNIST, PneumoniaMNIST, BreastMNIST, RetinaMNIST, BloodMNIST, OrganAMNIST, OrganCMNIST, and OrganSMNIST.

Table 1 reports the detailed configuration information of each dataset, including imaging modality, task category, and data scale. All datasets were partitioned in accordance with established protocols; notably, the division of training, validation, and test sets for the MedMNIST series datasets strictly adheres to the settings reported in existing studies [27,28], thereby ensuring the comparability and consistency of the evaluation results.

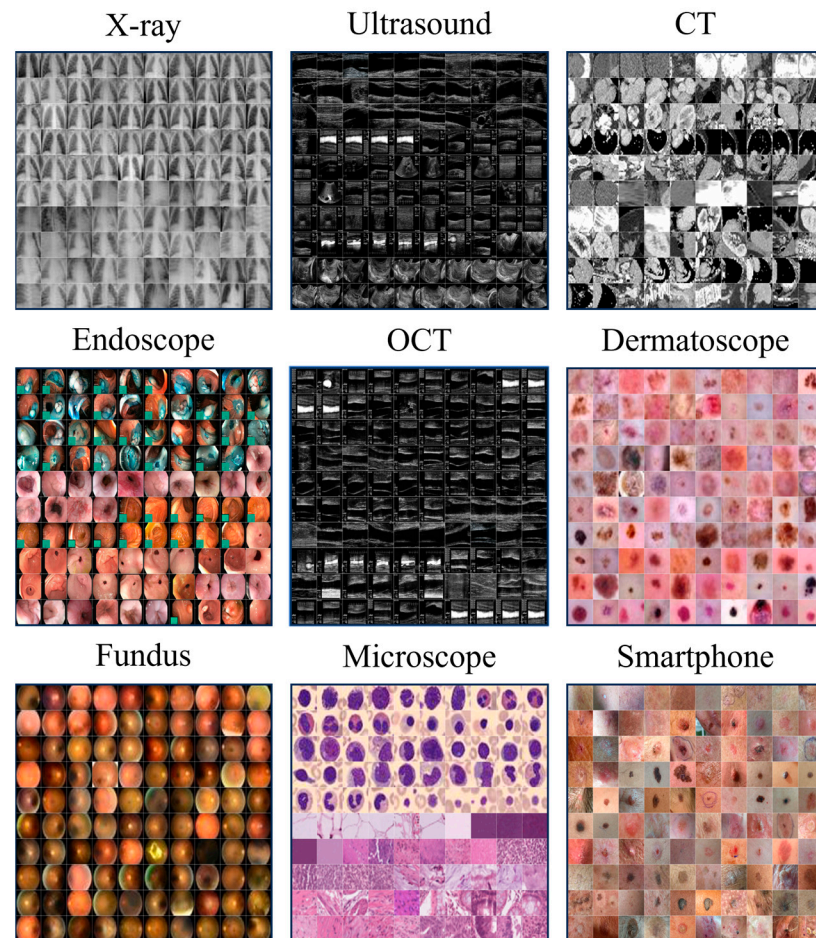


Figure 1. Visual Presentation of Nine Medical Imaging Modalities. Visually display the 9 medical imaging modalities involved in the study as icons, including CT, ultrasound, X-ray, endoscopy, etc., to clearly indicate the types of modalities covered by the dataset.

Table 1. The detailed descriptions for 14 datasets used in the work. Summarize key information for 14 datasets, including names, imaging modalities, task categories (single-class/multi-class), total data volume, and training/validation/test set allocation ratios, to establish a clear foundation for experimental data.

Name	Imaging Modality	Task (Classes)	Size	Training/Validation/Test
PAD-UFES-20	Human Skin Smartphone Image	MC (6)	2198	1384/227/687
CPN X-ray	Chest X-ray	MC (3)	5000	3140/521/1567
Fetal-Planes-DB	Maternal-fetal Ultrasound	MC (6)	1240	7446/1237/3717
Kvasir	Gastrointestinal Endoscope	MC (8)	4000	2408/392/1200
PathMNIST	Colon Pathology	MC (9)	89,969	10,004/7180/5536
OCTMNIST	Retinal OCT	MC (4)	109,309	97,477/10,832/1000
DermaMNIST	Dermatoscope	MC (6)	1600	1080/120/400
PneumoniaMNIST	Chest X-ray	BC (2)	5856	4708/524/624
BreastMNIST	Breast Ultrasound	BC (2)	780	546/78/156
RetinaMNIST	Fundus Camera	MC (5)	1600	1080/120/400
BloodMNIST	Blood Cell Microscope	MC (8)	17,992	11,959/1712/3634
OrganAMNIST	Abdominal CT	MC (11)	58,850	34,581/6491/17,778
OrganCMNIST	Abdominal CT	MC (11)	23,660	13,940/2452/8829
OrganSMNIST	Abdominal CT	MC (11)	25,221	13,940/2452/8829

2.2. Methods

2.2.1. MedMamba-ESS Model Architecture

Figure 2 illustrates the overall framework of MedMamba-ESS. In MedMamba-ESS, the input image is partitioned into fixed-size patches via the patch input layer and flattened into a 1D sequence. Consistent with the standard architecture of MedMamba, the patch input layer decomposes the image into non-overlapping small patches, enabling the model to process input data in a more efficient manner.

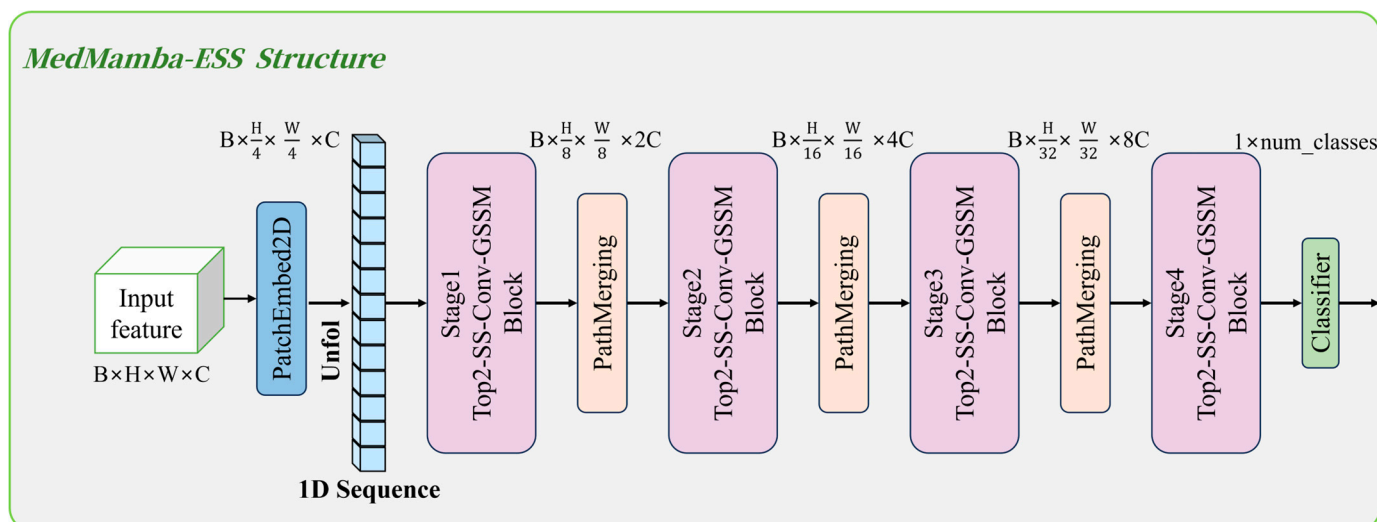


Figure 2. Overall Architecture of MedMamba-ESS. The complete structure of the MedMamba-ESS framework is presented, illustrating the workflow from input images processed through patch layers and multi-stage feature extraction (Stages 1–4), to the integration of the SS2D-Top2 and G-SSM modules, culminating in the output of classification results. Note: In the figure, “unfold” refers to block segmentation and flattening, while “feature” denotes the intermediate feature map (dimensions: $B \times H \times W \times C$, where B = batch size, H = height, W = width, C = channel).

Unlike MedMamba, MedMamba-ESS incorporates two innovative modules on the original basis: the SS2D-Top2 adaptive directional scanning mechanism and the G-SSM grouped parameter sharing module (Figure 3). The SS2D-Top2 mechanism flexibly adjusts directional activation across different resolution stages through fixed direction selection and a directional gating mechanism, ensuring computational efficiency is maintained at low-resolution stages while preserving more spatial feature details at high-resolution stages. In contrast, the G-SSM module significantly improves parameter efficiency and reduces redundant storage requirements via low-rank projection and a grouped sharing strategy.

Compared with MedMamba, MedMamba-ESS effectively addresses the computational and storage bottlenecks through these two innovative modules, thereby achieving higher efficiency and stronger expressive capability in medical image analysis.

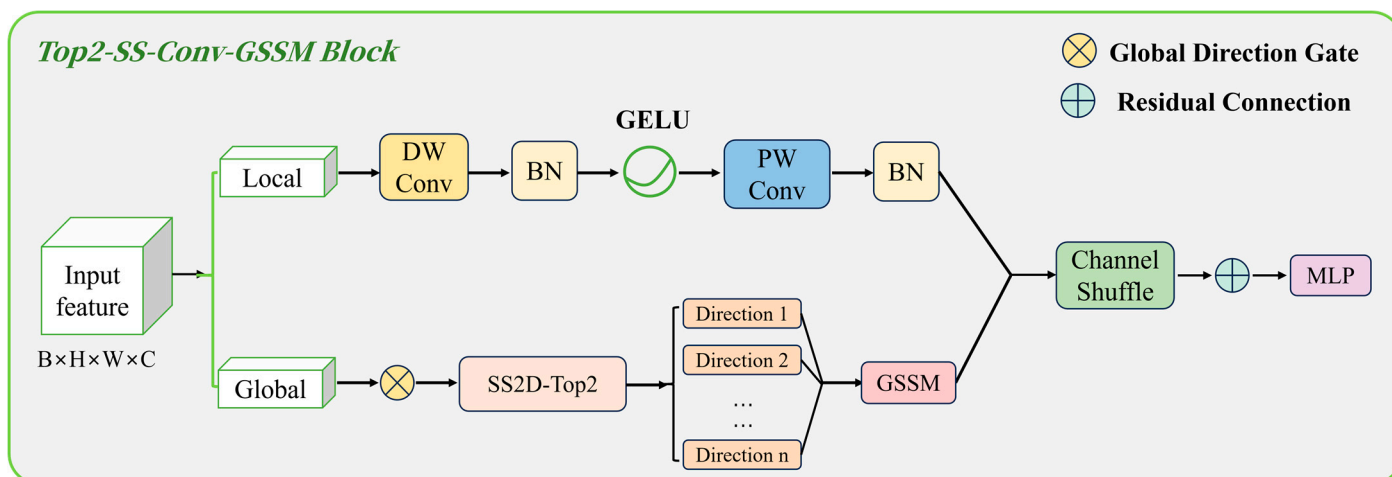


Figure 3. Schematic Diagram of the SS-Conv-GSSM Block Structure. Detailed illustration of the internal structure of the SS-Conv-GSSM module, including the connection relationships among key components such as input feature processing, local convolutions, SS2D-Top2 directional scanning, G-SSM parameter sharing, global directional gating, and residual connections. Note: BN refers to Batch Normalization. PW (Point-Wise Convolution) is a 1×1 convolution operation used to adjust the number of feature channels and fuse cross-channel information. Channel MLP refers to Channel Multi-Layer Perceptron. DW = Depth-Wise Convolution, GSSM = Grouped State Space Model, B = batch size, H = height, W = width, C = channel.

2.2.2. SS2D-Top2 Adaptive Directional Scanning Mechanism

Existing SS2D typically adopts a fixed multi-directional scanning strategy to compensate for the discrepancy between one-dimensional (1D) sequence modeling and two-dimensional (2D) spatial structures [29]. However, this approach introduces substantial redundant computation in the early feature extraction stage, especially in medical image applications where images often exhibit directional bias. For instance, in the CPN X-ray dataset, images feature prominent horizontal structures, while endoscopic images have distinct longitudinal textures—global modeling across all directions is therefore unnecessary.

As illustrated in Figure 4, to address such issues, we propose the SS2D-Top2 adaptive directional scanning mechanism, which aims to optimize computational efficiency through flexible scanning directions while preserving the ability to extract global contextual information. The directional scanning in this study employs an adaptive core direction selection based on image feature distribution, combined with rotation-invariant feature extraction through local convolutions. This approach effectively mitigates the impact of image rotation and demonstrates robustness when validated on clinical datasets exhibiting natural directional variations.

Specifically, the design is implemented as follows: First, in Stage 1 and Stage 2 (where image resolution is low), only two directions are used for scanning; reducing the number of scanning directions helps significantly lower computational complexity. Subsequently, in Stage 3 and Stage 4 (with increased image resolution), four directions are employed for scanning—at this stage, four directions can effectively enhance feature extraction capability and facilitate the capture of more spatial information. The workflow of this mechanism is as follows:

$$N_{\text{directions}} = \begin{cases} 2 & \text{if stage in } \{1, 2\} \\ 4 & \text{if stage in } \{3, 4\} \end{cases} \quad (1)$$

In addition to stage-wise fixed direction selection, another key measure of the SS2D-Top2 mechanism is the introduction of a directional gating mechanism. Gating is a

mechanism commonly used in neural networks, designed to control which information should be retained and which should be suppressed or discarded [30]. Gating mechanisms typically rely on learnable weights or functions to dynamically determine which parts of the input need to be passed to the next layer of the network [31].

In our improvement, the gating mechanism serves to dynamically control the activation status of scanning directions in each stage. Although the number of directions per stage is fixed, the gating mechanism enables selective activation of each direction, thereby reducing redundant computation. In Stages 1 and 2, the gating mechanism may decide to only use Direction A while ignoring Direction B. For Stages 3 and 4, although four directions are available, the gating mechanism may choose to activate only three of them.

Specifically, in each stage, the features of the input image are fed into a gating network to compute activation weights for each direction, after which these weights determine whether each direction is enabled. For each scanning direction, the gating mechanism uses an activation function to decide whether to perform computation for that direction. The specific implementation process of the gating mechanism is as follows:

First, the input features are processed: the gating network receives the image feature x_t and outputs activation weights w_t for each direction via the network:

$$w_t = \text{GateNetwork}(x_t) \quad (2)$$

Herein, $w_t = \{w_1, w_2, \dots, w_n\}$ is a vector representing the activation weight of each direction, and x_t denotes the input image feature.

Subsequently, the activation weights output by the network are converted into probability values via the Softmax activation function, ensuring that the weights of all directions fall within the range of 0 to 1 and their sum equals 1. This means the activation level of each direction is dynamically adjusted, representing the contribution degree of each direction. For each direction i its activation value w_i is defined as follows:

$$w_i = \text{softmax}(f_i(x_t)) \quad (3)$$

Herein, $f_i(x_t)$ denotes the linear combination or convolutional output generated by the gating network for direction i , and the Softmax function converts these scores into a probability.

Finally, directional selection is performed: based on the activation weight w_i of each direction, the model selectively enables or disables each direction. The final directional selection function is defined as follows:

$$y_t = \sum_{i=1}^n w_i \cdot D_i(x_t) \quad (4)$$

Herein, $D_i(x_t)$ denotes the scanning operation for the i -th direction, and y_t represents the feature map output at this stage. The value of w_i controls whether the corresponding direction is activated.

Combining fixed direction selection and the directional gating mechanism, the complete workflow of the SS2D-Top2 mechanism can be expressed as follows:

$$\text{Output}(x_t) = \sum_{i=1}^n w_t[i] \cdot \text{ScanDirection}_i(x_t) \quad (5)$$

Herein, $w_t[i]$ denotes the activation weight generated for the i -th direction, and $\text{ScanDirection}_i(x_t)$ represents the operation of scanning the image along the i -th direction.

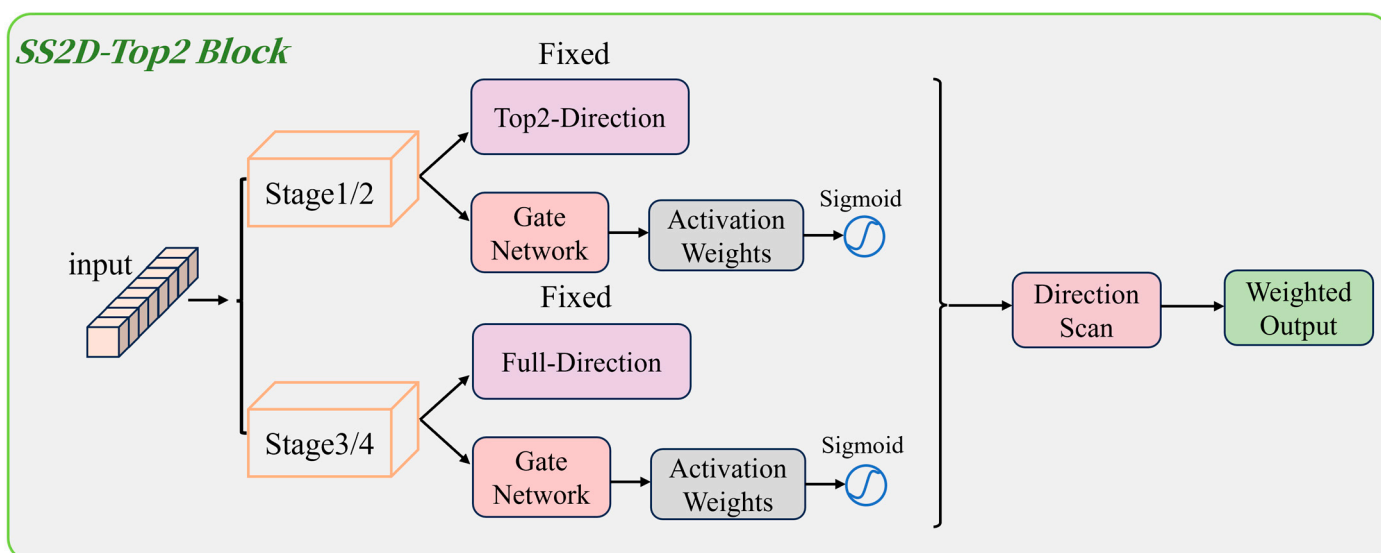


Figure 4. Structural Diagram of the SS2D-Top2 Adaptive Scanning Mechanism. Explains the working principle of SS2D-Top2: The left side represents the low-resolution stage (Stage 1/2) with fixed 2-direction scanning, while the right side shows the high-resolution stage (Stage 3/4) with fixed 4-direction scanning. Both stages achieve direction-selective activation by generating weights through a gating network and SoftMax.

2.2.3. G-SSM Grouped Parameter Sharing Mechanism

In traditional State Space Models (SSMs), state matrices (A , B , C , D) typically contain redundant parameters, especially duplicate storage in the channel dimension, which not only increases computational burden but also raises storage requirements. In high-resolution medical image processing tasks, such redundancy significantly impairs model performance and computational efficiency [14]. To address this issue, this study proposes the G-SSM grouped parameter sharing module (Figure 5), which preserves fine-grained feature principal components through low-rank projection and enables grouped sharing based on semantic similarity. It achieves lightweight processing while maintaining the ability to distinguish similar clinical features across different inflammation grades and lesion subtypes.

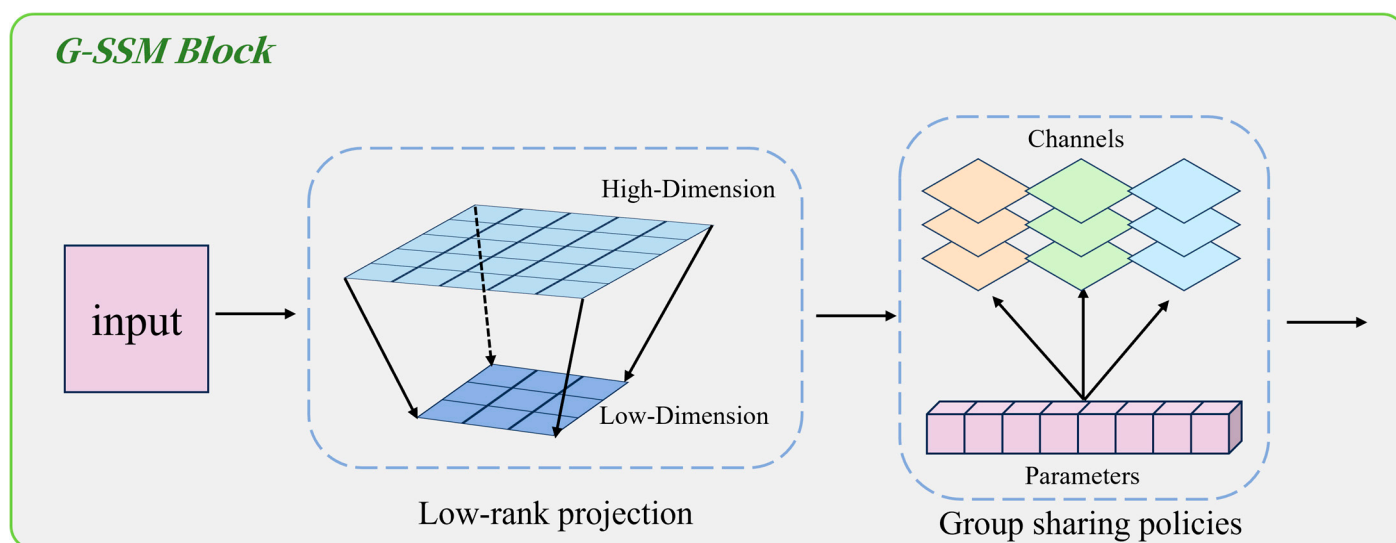


Figure 5. Structural Diagram of the G-SSM Grouped Parameter Sharing Mechanism. Demonstrates the processing flow of the G-SSM module for input high-dimensional channel features, including

low-rank projection to map parameters to a low-dimensional space, followed by optimizing the state matrix through a grouping and sharing strategy to reduce parameter redundancy.

First is low-rank projection: in the G-SSM, the dimensions of state matrices B and C are reduced to r ($r \ll d_{\text{state}}$), and redundant computation is reduced by introducing low-rank projection. The projection matrix of each channel is mapped to a low-dimensional space, and the number of parameters is reduced from the original $\mathcal{O}(d_{\text{state}} \cdot d_{\text{inner}})$ to $\mathcal{O}(r \cdot d_{\text{inner}})$, where r denotes the rank of the low-rank projection.

$$\tilde{B}_i = W_B \cdot \alpha_i^B \quad (6)$$

$$\tilde{C}_i = W_C \cdot \alpha_i^C \quad (7)$$

Herein, α_i^B and α_i^C denote the low-rank coefficient vectors for the i -th channel, and W_B and W_C represent the shared projection matrices.

Subsequently, a grouped sharing strategy is adopted for state matrices A and D in G-SSM. Each group (a total of groups) of matrices A and D shares the same parameters, and each channel within these groups shares one set of parameters for A and D . The formula is as follows:

$$A_{\text{shared}} = \text{repeat}_{\text{group}}(A) \quad (8)$$

$$D_{\text{shared}} = \text{repeat}_{\text{group}}(D) \quad (9)$$

Herein, A_{shared} and D_{shared} denote the shared versions of matrices A and D obtained via the group sharing mechanism, respectively.

Through low-rank projection and grouped sharing strategies, the finally optimized matrices are expressed as follows:

$$A_{\text{optimized}} = \sum_{i=1}^k \text{rank}_{\text{proj}}(A_i) \cdot \text{group}_{\text{proj}}(A_i), A_i \in \{A, B, C, D\} \quad (10)$$

Herein, $\text{rank}_{\text{proj}}(A_i)$ denotes the low-rank projection operation, and $\text{group}_{\text{proj}}(A_i)$ represents the grouped sharing operation.

2.3. Experimental Setup

2.3.1. Implementation Details

To ensure fair comparison, all images were uniformly resized to $224 \times 224 \times 3$ before training each network, and normalization and standardization were performed on each dataset.

During the training of non-MedMNIST datasets, we adopted the AdamW optimizer with an initial learning rate of 0.0001, β_1 of 0.9, β_2 of 0.999, and weight decay of $1e-4$, and used the cross-entropy loss function to optimize model parameters. The PyTorch 2.0.1 framework was employed for network training. Each model was trained for 150 epochs with a batch size of 64. To prevent model overfitting, an early stopping strategy was adopted: We monitored the validation loss during training; if the validation loss did not decrease for 20 consecutive epochs (patience = 20) and the relative improvement was less than $1e-4$, training was terminated early. The model parameters corresponding to the minimum validation loss during training were retained as the final model parameters to ensure optimal generalization performance. Additionally, to verify the uniqueness of the MedMamba architecture to the greatest extent possible, no data augmentation strategies or pre-trained models were used. For the training of MedMNIST datasets, we followed the identical training settings of MedMNISTv2 and MedViT without any modifications. Specifically, we trained MedMamba-ESS (based on MedMamba-Tiny) for 100 epochs

with a batch size of 128. The AdamW optimizer with an initial learning rate of 0.001 was used, and the learning rate was decayed by a factor of 0.1 at the 50th and 75th epochs. All training processes were conducted on a computer equipped with the Ubuntu 22.04 operating system and four NVIDIA GeForce RTX 4090 GPUs.

2.3.2. Evaluation Metrics

We used Overall Accuracy (OA), Precision, Sensitivity, Specificity, F1-score, and the Area Under the ROC Curve (AUC) to measure how well the model worked.

Overall Accuracy (OA) shows how many of the predictions were correct (true positives and true negatives) out of all the predictions. This is a big picture view of how well the model is doing. Precision is more about the model's positive predictions, which is the percentage of those predictions that are actually positive. Sensitivity (also called Recall) shows how many of the actual positive samples the model correctly identified. Specificity, on the other hand, shows how well the model can find negative samples. The F1-score is the harmonic mean of Precision and Sensitivity. It is used to fully assess how well the two are balanced. AUC (Area Under the ROC Curve) looks at the model's ability to tell the difference between positive and negative samples by looking at how well it does at different decision thresholds. A higher value means that the model is better at classifying samples.

The specific calculation formulas are as follows:

$$\text{Overall Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (12)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (13)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (14)$$

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \quad (15)$$

Additionally, we also analyzed the model computational complexity (FLOPs) and the size of model parameters.

3. Results

3.1. Performance of MedMamba-ESS

We did a lot of tests with MedMamba-Tiny as the baseline model to see how well MedMamba-ESS worked for classifying medical images. The experimental results show that MedMamba-ESS made big improvements in performance on several medical image datasets, especially in important evaluation metrics like Overall Accuracy (OA) and the Area Under the ROC Curve (AUC).

We evaluated MedMamba-ESS on multiple non-MedMNIST datasets and compared it with the baseline model. The experimental results show that the performance metrics of MedMamba-ESS were basically superior to those of the baseline model, and exceeded those of other state-of-the-art reference models in some metrics (Figures 6 and 7).

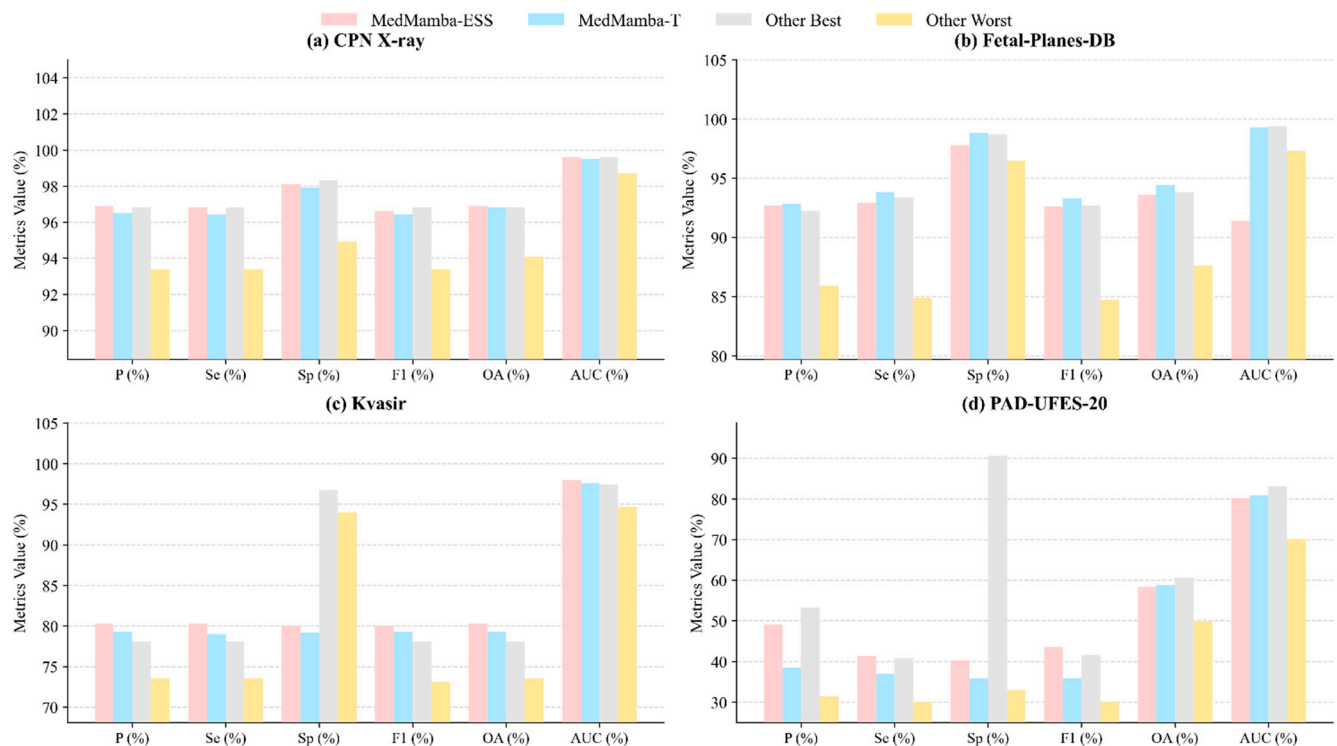


Figure 6. Performance Comparison of MedMamba-ESS, MedMamba-Tiny, and the Best and Worst of Other Models on Non-MedMNIST Datasets. (a): Performance comparison chart for the CPN X-ray dataset. (b): Performance comparison chart for the Kvasir dataset. (c): Performance comparison chart for the Fetal-Planes-DB dataset. (d): Performance comparison chart for the PAD-UFES-20 dataset.

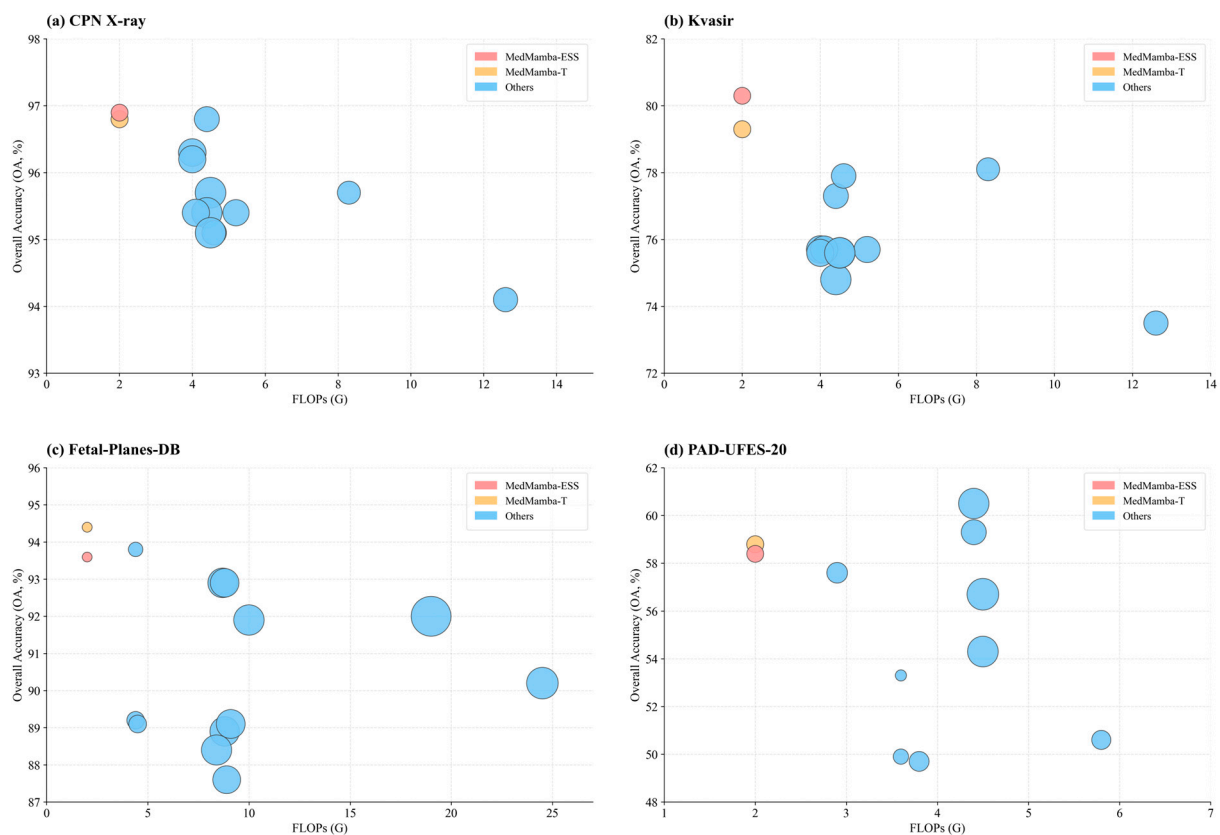


Figure 7. Performance Comparison Between MedMamba-ESS and Reference Models on Non-MedMNIST Datasets (Larger circles indicate more parameters). (a): CPN X-ray dataset per-

formance versus efficiency comparison; (b): Kvasir dataset performance versus efficiency comparison; (c): Fetal-Planes-DB dataset performance versus efficiency comparison; (d): PAD-UFES-20 dataset performance versus efficiency comparison.

Table 2 presents the experimental results of MedMamba-ESS on the CPN X-ray and Kvasir datasets. The MedMamba-ESS model exhibited significant advantages in both classification performance and computational efficiency. On the CPN X-ray dataset, MedMamba-ESS achieved an OA of 96.9%, which was 0.4% higher than that of the baseline model MedMamba-Tiny. On the more challenging Kvasir endoscopic image dataset, MedMamba-ESS achieved an OA of 80.3%, a 1.0% improvement over the baseline model. It is worth noting that while maintaining performance, MedMamba-ESS only required 2.0 G FLOPs and 14.3 M parameters, demonstrating higher computational efficiency compared with other comparative models. For example, although VMamba-S exhibited comparable performance to MedMamba-ESS, its FLOPs (4.4 G) and parameter count (22.1 M) were 2.2 times and 1.5 times those of MedMamba-ESS, respectively. Compared with mainstream models such as Swin-S and ConvNeXt-S, MedMamba-ESS showed significant advantages in terms of computational efficiency.

Table 2. Performance Comparison of MedMamba-ESS and Reference Models on Non-MedMNIST Datasets. The baseline model is indicated in red font. Para, P, Se, Sp, and F1 denote parameter size, Precision, Sensitivity, Specificity, and F1-score, respectively.

Dataset	Model	FLOPs (G)	Paras	P (%) ↑	Se (%) ↑	Sp (%) ↑	F1 (%) ↑	OA (%) ↑	AUC ↑
CPN X-ray	MedMamba-ESS	2.0	14.3	96.9	96.8	98.1	96.6	96.9	0.996
	MedMamba-Tiny	2.0	14.5	96.5	96.4	97.9	96.4	96.8	0.995
	VMamba-T [29]	4.4	22.1	96.7	96.9	98.2	96.7	96.9	0.996
	Swin-T [32]	4.4	27.5	95.9	95.6	97.5	95.6	95.6	0.994
	ConvNeXt-T [33]	4.5	27.8	95.6	95.9	97.9	95.6	95.6	0.994
	Conformer-s18 [34]	4.0	24.7	95.8	95.9	97.6	95.7	96.4	0.993
	TNT-s [35]	5.2	23.3	93.6	93.2	95.1	93.6	95.2	0.992
	Caformer-s18 [34]	4.1	24.3	95.7	95.3	97.5	95.9	95.6	0.992
	PvtV2-b2 [36]	4.0	24.3	95.4	96	98.3	96	96.4	0.994
	Davit-tiny [37]	4.5	27.6	95.5	95.3	97.5	95.3	95.3	0.993
	Deit-small [38]	4.6	21.7	95.5	95.3	97.7	95.3	95.3	0.993
	EfficientNetV2-s [39]	8.3	20.2	95.5	95.9	97.6	95.9	95.9	0.987
	Coat-small [40]	12.6	21.4	94.4	94	97.2	94.3	94.3	0.987
Kvasir	MedMamba-ESS	2.0	14.3	80.3	80.3	80.0	80.1	80.3	0.980
	MedMamba-Tiny	2.0	14.5	79.3	79.0	79.2	79.3	79.3	0.976
	VMamba-T [29]	4.4	22.1	77.6	77.3	96.8	77.6	77.3	0.950
	Swin-T [32]	4.4	27.5	75.8	74.8	94.9	77.8	75.6	0.969
	ConvNeXt-T [33]	4.5	27.8	75.6	74.8	95.1	75.6	75.6	0.968
	Conformer-s18 [34]	4.0	24.7	76.4	75.6	95.7	75.6	75.7	0.970
	TNT-s [35]	5.2	23.3	74.6	75.7	96.5	75.7	74.7	0.971
	Caformer-s18 [34]	4.1	24.3	75.7	74.5	96.5	74.7	75.9	0.963
	PvtV2-b2 [36]	4.0	24.3	75.6	75.6	95.3	75.6	75.6	0.958
	Davit-tiny [37]	4.5	27.6	75.6	74.8	94.3	75.5	75.6	0.968
	Deit-small [38]	4.6	21.7	79.1	79.2	96.5	77.9	77.9	0.963
	EfficientNetV2-s [39]	8.3	20.2	78.1	78.3	96.5	78.1	78.1	0.974
	Coat-small [40]	12.6	21.4	72.5	73.5	96.2	73.6	73.5	0.957

Table 3 presents the experimental results on the PAD-UFES-20 and Fetal-Planes-DB datasets, further verifying the generalization ability and efficiency advantages of the

MedMamba-ESS model. In the task of classifying skin lesions, MedMamba-ESS achieved an OA of 58.4% and an AUC of 0.802. Although its OA was slightly lower than the baseline model MedMamba-Tiny (58.8%), its Precision was significantly improved by 10.7%, rising from 38.4% (baseline model) to 49.1%. Meanwhile, it maintained a leading position in computational efficiency, requiring only 2.0G FLOPs and 14.3M parameters—significantly lower than comparative models such as VMamba-T (4.4G FLOPs) and Swin-T (4.4G FLOPs). In the fetal ultrasound image classification task, MedMamba-ESS achieved an OA of 93.6% and an AUC of 0.914, which were slightly lower than the baseline model MedMamba-Tiny (OA: 94.4%, AUC: 0.993). However, with a computational cost of only 2.0G FLOPs, it achieved an OA level comparable to large models such as EfficientNet-b6 and MViTv2-base, highlighting its advantages in balancing computational efficiency and performance. Overall, MedMamba-ESS maintained competitiveness in OA and AUC metrics while drastically reducing parameter and computational overhead.

Table 3. Performance Comparison of MedMamba-ESS and Reference Models on Non-MedMNIST Datasets. Continuing the format of Table 2, we supplement the comparison of model performance on the PAD-UFES-20 and Fetal-Planes-DB datasets to further validate the generalization capability of MedMamba-ESS. The baseline model MedMamba-Tiny is marked in red.

Dataset	Model	FLOPs (G)	Paras	P (%) ↑	Se (%) ↑	Sp (%) ↑	F1 (%) ↑	OA (%) ↑	AUC ↑
PAD-UFES-20	MedMamba-ESS	2.0	14.3	49.1	41.4	40.3	43.6	58.4	0.802
	MedMamba-Tiny	2.0	14.5	38.4	36.9	35.8	35.8	58.8	0.808
	VMamba-T [29]	4.4	22.1	53.2	40.6	90.0	41.6	59.3	0.830
	Swin-T [32]	4.4	27.5	53.2	40.8	90.6	39.5	60.5	0.830
	ConvNeXt-T [33]	4.5	27.8	37.2	33.7	89.9	35.7	54.3	0.760
	Repvgg-a1 [41]	4.5	28.8	37.2	33.7	89.7	36.0	56.7	0.705
	Mobilevitv2-200 [42]	3.6	12.8	34.9	33.9	32.9	35.8	49.9	0.705
	EdgeNext-base [43]	2.9	17.9	35.0	34.4	89.9	34.4	57.6	0.802
	Nest-tiny [44]	5.8	16.4	33.9	32.2	87.8	31.6	50.6	0.702
	Mobileones4 [45]	3.6	9.0	31.5	30.0	89.9	30.0	53.3	0.701
	Cait-xxs36 [46]	3.8	17.1	36.7	31.7	88.0	33.4	49.7	0.705
Fe-tal-Planes-DB	MedMamba-ESS	2.0	14.3	92.7	92.9	97.8	92.6	93.6	0.914
	MedMamba-Tiny	2.0	14.5	92.8	93.8	98.8	93.3	94.4	0.993
	VMamba-T [29]	4.4	22.1	92.4	93.4	98.7	92.7	93.8	0.994
	Swin-T [32]	4.4	27.5	86.1	84.9	97.7	85.9	89.2	0.983
	ConvNeXt-T [33]	4.5	27.8	85.7	85.2	97.7	85.5	89.6	0.981
	Davit-small [37]	8.8	48.9	85.9	85.2	97.6	85.8	88.9	0.977
	Mvitv2-base [47]	10.0	50.7	89.9	90.1	98.8	89.1	91.9	0.989
	EfficientNet-b6 [39]	19.0	68.4	91.2	91.2	98.6	91.0	91.5	0.996
	EfficientNetV2-s [39]	24.5	52.9	87.6	89.1	97.9	89.2	90.2	0.983
	FocalNet-s [48]	8.7	49.1	91.7	90.9	97.9	91.2	92.9	0.987
	Twins-SVT-base [49]	8.8	47.3	88.4	87.5	96.7	87.4	92.9	0.985
	Poolformer-m36 [50]	8.9	45.8	82.7	87.4	96.5	84.7	87.6	0.973
	Xcit-s [51]	9.1	47.8	85.5	85.2	97.7	85.5	89.1	0.975
	GcVit-s [52]	8.4	50.3	84.5	84.3	97.5	84.3	88.4	0.977

Table 4 and Figure 8 show a comprehensive comparison of the performance of representative models on four non-MedMNIST medical image datasets. It can be seen that the proposed MedMamba-ESS model achieved the best overall performance, with an average OA of 82.1%—an increase of 0.3% compared with the baseline model Med-

Mamba-Tiny (81.8%). MedMamba-ESS outperformed existing mainstream architectures by 0.9%, Swin-I by 1.9%, and ConvNeXt-I by 2.5% in average OA, respectively.

Table 4. Performance Comparison of Different Models on Non-MedMNIST Datasets. By integrating experimental results from four non-MedMNIST datasets, we calculated and compared the average Overall Accuracy (OA) and average AUC of representative models, highlighting MedMamba-ESS's overall performance advantage.

Model	CPN X-ray		Kvasir		PAD-UFES-20		Fetal-Planes-DB		Average	
	OA	AUC	OA	AUC	OA	AUC	OA	AUC	OA	AUC
MedMamba-ESS	96.9	0.996	80.3	0.980	58.4	0.802	93.6	0.914	82.1	0.922
MedMamba-Tiny	96.8	0.995	79.3	0.976	58.8	0.808	94.4	0.993	81.8	0.943
VMamba-T [29]	96.9	0.996	77.3	0.950	59.3	0.830	93.8	0.994	81.2	0.947
Swin-T [32]	95.6	0.994	75.6	0.969	60.5	0.830	89.2	0.983	80.2	0.944
ConvNeXt-T [33]	95.6	0.994	75.6	0.968	54.3	0.760	89.6	0.981	79.6	0.928

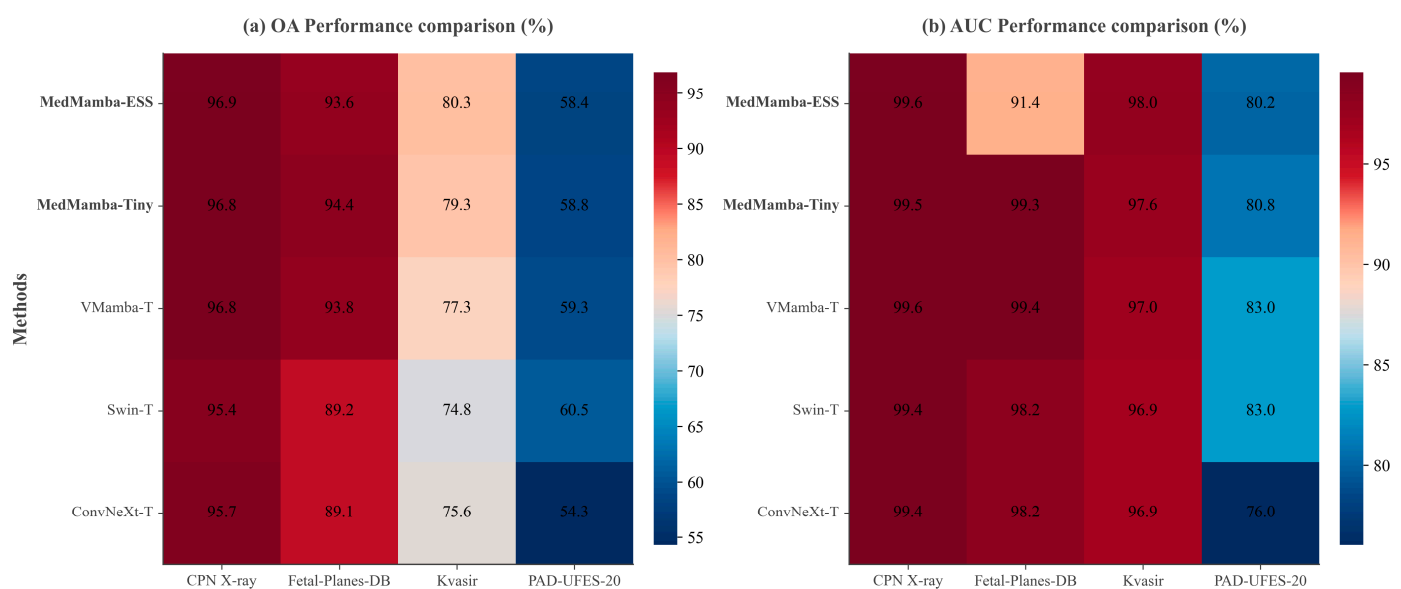


Figure 8. Heatmap of Performance Comparison of Different Models on Non-MedMNIST Datasets. The heatmap uses color intensity to represent the magnitude of OA and AUC values across various models (such as MedMamba-ESS, VMamba-T, Swin-T, etc.) on four non-MedMNIST datasets, enabling quick identification of the model with optimal performance. (a): Heatmap of Over-Advertising (OA) performance for models on non-MedMNIST datasets; (b): Heatmap of Area Under the Curve (AUC) performance for models on non-MedMNIST datasets.

We then compared MedMamba-ESS with the baseline model after evaluating it on many MedMNIST datasets. Under various MedMNIST datasets, experiments showed that MedMamba-ESS's OA and AUC performance metrics were essentially better than the baseline model's, and in certain cases, better than those of other cutting-edge comparison models (Figure 9).

Table 5 provides a detailed comparison of the performance of MedMamba-ESS and other models on each MedMNIST dataset. Compared with the baseline model MedMamba-Tiny, MedMamba-ESS showed significant improvements on multiple key datasets. On the PathMNIST dataset, MedMamba-ESS achieved an OA of 96.1%, an increase of 0.8 percentage points compared with MedMamba-Tiny; on the DermaMNIST dataset, its AUC value increased to 0.926, outperforming MedMamba-Tiny (0.917); on the OCTMNIST dataset, the AUC value of MedMamba-ESS reached as high as 0.996, an increase of 0.4 percentage points compared with MedMamba-Tiny. Additionally, Med-

Mamba-ESS also achieved excellent AUC performances of 0.989 and 0.980 on the PneumoniaMNIST and OrganSMNIST datasets, respectively. Although its OA was slightly lower than that of the baseline model on some datasets (e.g., BloodMNIST and OrganAMNIST) (Figure 9), MedMamba-ESS achieved performance improvements on most datasets while maintaining the same computational efficiency (2.0G FLOPs), demonstrating its stability and generalization ability in medical image classification tasks. Notably, the performance of MedMamba-ESS on multiple datasets outperformed traditional ResNet series, automated machine learning methods, and the MedViT series.

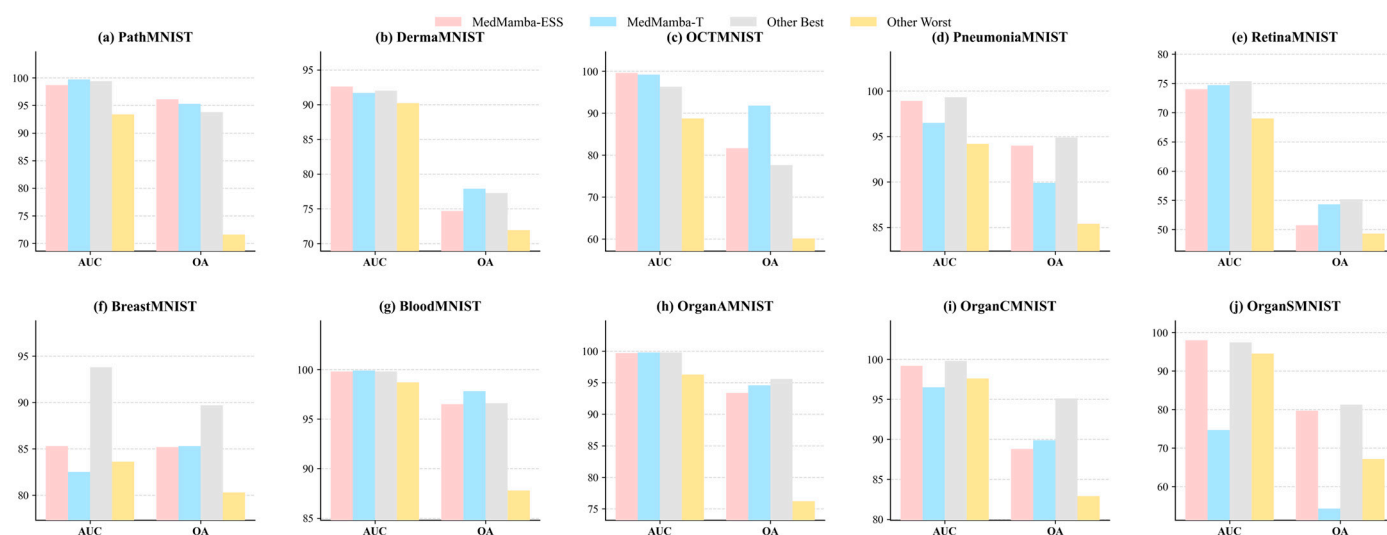


Figure 9. Performance Comparison of MedMamba-ESS, MedMamba-Tiny, and the Best and Worst of Other Models on MedMNIST Datasets (a): Performance comparison chart for the PathMNIST dataset; (b): Performance comparison chart for the DermaMNIST dataset; (c): Performance comparison chart for the OCTMNIST dataset; (d): Performance comparison chart for the PneumoniaMNIST dataset; (e): Performance comparison chart for the RetinaMNIST dataset; (f): Performance comparison chart for BreastMNIST dataset; (g): Performance comparison chart for BloodMNIST dataset; (h): Performance comparison chart for OrganAMNIST dataset; (i): Performance comparison chart for OrganCMNIST dataset; (j): Performance comparison chart for OrganSMNIST dataset.

Table 5. Performance Comparison of MedMamba-ESS and Reference Models on MedMNIST Datasets. On ten MedMNIST datasets, detailed comparisons of AUC and OA metrics between MedMamba-ESS and models such as the ResNet series, Auto-sklearn, and the MedViT series are presented. The baseline model MedMamba-Tiny is marked in red.

Model	PathMNIS		DermaMN		OCTMNIS		PneumoniaMNIST		RetinaMNIST	
	AUC	OA	AUC	OA	AUC	OA	AUC	OA	AUC	OA
ResNet18 (28) [53]	0.983	0.907	0.917	0.735	0.943	0.743	0.944	0.854	0.717	0.524
ResNet18 (224) [53]	0.989	0.909	0.920	0.754	0.958	0.763	0.956	0.864	0.710	0.493
ResNet50 (28) [53]	0.990	0.911	0.913	0.735	0.952	0.762	0.948	0.854	0.726	0.528
ResNet50 (224) [53]	0.989	0.892	0.912	0.731	0.958	0.776	0.962	0.884	0.716	0.511
Auto-sklearn [54]	0.934	0.716	0.902	0.719	0.887	0.601	0.942	0.855	0.690	0.515
AutoKeras [55]	0.959	0.834	0.915	0.749	0.955	0.763	0.947	0.878	0.719	0.503
Google AutoML [23]	0.944	0.728	0.914	0.768	0.963	0.771	0.991	0.946	0.750	0.531
MedVit-T [28]	0.994	0.938	0.914	0.768	0.961	0.767	0.993	0.949	0.752	0.534
MedVit-S [28]	0.994	0.938	0.914	0.768	0.961	0.767	0.993	0.949	0.752	0.534
MedVit-L [28]	0.984	0.933	0.920	0.773	0.945	0.761	0.991	0.921	0.754	0.552
MedMamba-Tiny	0.997	0.953	0.917	0.779	0.992	0.918	0.965	0.899	0.747	0.543

MedMamba-ESS	0.987	0.961	0.926	0.747	0.996	0.816	0.989	0.940	0.740	0.507
Model	BreastMNIST		BloodMNIST		OrganAMNIST		OrganCMNIST		OrganSMNIST	
	AUC	OA	AUC	OA	AUC	OA	AUC	OA	AUC	OA
ResNet18 (28) [53]	0.901	0.863	0.998	0.958	0.997	0.935	0.992	0.900	0.972	0.782
ResNet18 (224) [53]	0.891	0.833	0.998	0.963	0.997	0.956	0.998	0.951	0.974	0.778
ResNet50 (28) [53]	0.857	0.812	0.997	0.956	0.997	0.935	0.997	0.947	0.972	0.770
ResNet50 (224) [53]	0.866	0.842	0.997	0.950	0.998	0.947	0.998	0.947	0.972	0.770
Auto-sklearn [54]	0.836	0.803	0.987	0.878	0.963	0.762	0.976	0.829	0.945	0.672
AutoKeras [55]	0.871	0.831	0.998	0.961	0.994	0.905	0.990	0.879	0.974	0.813
Google AutoML [56]	0.919	0.861	0.998	0.966	0.990	0.886	0.988	0.877	0.964	0.749
MedVit-T [28]	0.934	0.896	0.996	0.950	0.995	0.931	0.991	0.901	0.972	0.789
MedVit-S [28]	0.938	0.897	0.997	0.951	0.996	0.928	0.993	0.916	0.972	0.791
MedVit-L [28]	0.929	0.883	0.996	0.954	0.997	0.943	0.994	0.922	0.973	0.806
MedMamba-Tiny	0.825	0.853	0.999	0.978	0.998	0.946	0.965	0.899	0.747	0.543
MedMamba-ESS	0.853	0.852	0.998	0.965	0.997	0.934	0.992	0.888	0.980	0.797

Table 6 and Figure 10 show the average performance metrics of different models on ten MedMNIST datasets. Through comparison, it can be seen that MedMamba-ESS exhibited excellent performance in the two key metrics of AUC and OA, reaching 0.946 and 0.841, respectively. Compared with other models, MedMamba-ESS had slightly better performance, especially outperforming most comparative models (e.g., ResNet18 and Auto-sklearn) in terms of AUC value, further verifying its effectiveness and reliability in medical image classification tasks. Although many advanced models (e.g., MedVit-S and MedVit-LI) also exhibited high accuracy and AUC values, MedMamba-ESS achieved a better balance between performance and efficiency through more efficient structural design.

Table 6. Average Performance Metrics Across MedMNIST Datasets for Various Models. Compute and compare the average AUC and average OA of each model across ten MedMNIST datasets to quantify the performance gap between MedMamba-ESS and other models.

Model	The Average Metrics Across the Ten MedMNIST Datasets	
	AUC	OA
ResNet18 (28) [53]	0.936	0.820
ResNet18 (224) [53]	0.939	0.826
ResNet50 (28) [53]	0.935	0.821
ResNet50 (224) [53]	0.937	0.825
Auto-sklearn [54]	0.906	0.735
AutoKeras [55]	0.932	0.812
Google AutoML [56]	0.942	0.808
MedVit-T [28]	0.950	0.842
MedVit-S [28]	0.951	0.844
MedVit-L [28]	0.948	0.845
MedMamba-Tiny	0.915	0.831
MedMamba-ESS	0.946	0.841

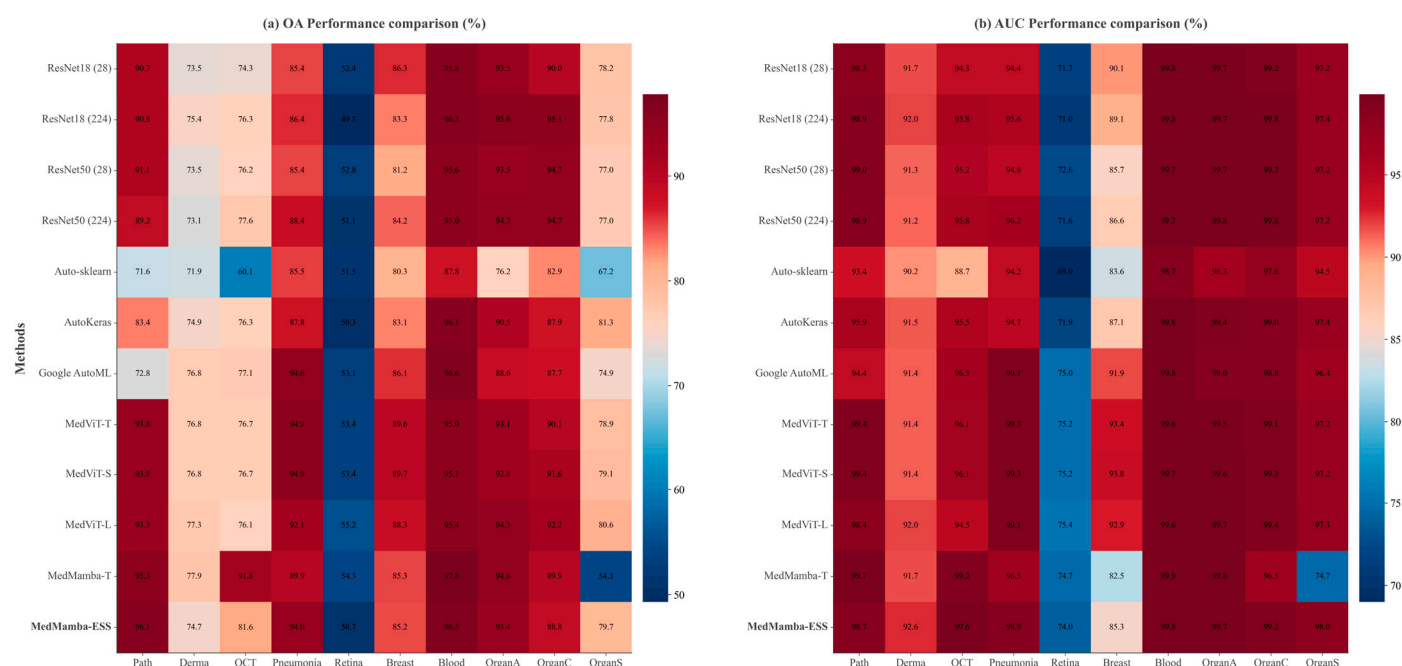


Figure 10. Heatmap of Performance Comparison of Different Models on MedMNIST Datasets. Visualize the numerical distributions of OAC and AUC across various models (such as the ResNet series, MedVit series, MedMamba-ESS, etc.) on ten MedMNIST datasets with heatmaps. (a): Heatmap of OA performance for models on the MedMNIST dataset; (b): Heatmap of AUC performance for models on the MedMNIST dataset.

3.2. Computational Efficiency Analysis of the SS2D-Top-2 Mechanism

To quantify the computational gains of the SS2D-Top-2 adaptive directional scanning mechanism, we conducted a detailed FLOPs analysis on the ChestXray dataset under different backbone network scales and input resolutions.

Theoretical calculations indicate that FLOPs exhibit a linear growth trend with the increase in input image resolution [10]. In the traditional selective scanning module SS2D, the computational load increases as the input resolution rises from 224×224 to 384×384 . The SS2D-Top-2 mechanism effectively controls this growth by optimizing scanning directions (Figure 11).

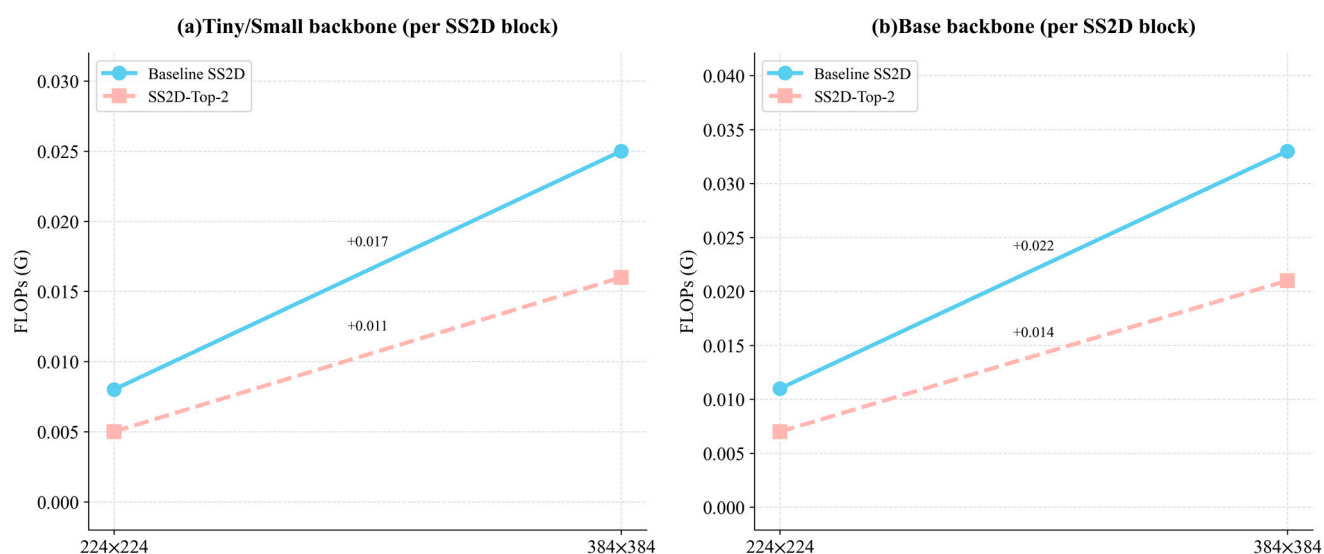


Figure 11. Comparison of FLOPs growth rates between SS2D-Top2 and the baseline SS2D module under different backbone networks and input resolutions. The values such as +0.017 in the figure

represent the slope. (a): Comparison of FLOPs growth rates under Tiny/Small backbone networks. (b): Comparison of FLOPs growth rates under Base backbone networks.

Table 7 shows that under the Tiny/Small architectures with a 224×224 input, the traditional selective scanning module SS2D consumed 0.008 GFLOPs of computational resources, while when the resolution was increased to 384×384 , the computational consumption of the traditional SS2D increased significantly to 0.025 GFLOPs. We introduced the SS2D-Top2 adaptive scanning mechanism; although FLOPs increased with resolution, the growth rate was effectively controlled. At a resolution of 224×224 , the computational consumption decreased from 0.008 GFLOPs to 0.005 GFLOPs, and at 384×384 , it decreased from 0.025 GFLOPs to 0.016 GFLOPs, achieving approximately a 37% reduction in computational load in both cases. The same computational growth trend was also observed under the Base architecture. In the Base architecture, inputs of 224×224 and 384×384 resulted in computational consumptions of 0.011 GFLOPs and 0.033 GFLOPs, respectively; after introducing SS2D-Top2, these two values dropped to 0.007 GFLOPs and 0.021 GFLOPs, respectively, also achieving an approximate 37% reduction.

Table 7. Comparison of FLOPs computational efficiency of SS2D-Top2 on Tiny/Small and Base architectures with different resolutions. Comparing Tiny/Small and Base architectures, MedMamba (baseline SS2D) and MedMamba-ESS (SS2D-Top2) demonstrate reduced FLOPs.

Resolution	SSM (Tiny/Small)		Reduction (Tiny/Small)	SSM (Base)		Reduction (Base)
	MedMamba (FLOPs)	MedMamba-ESS (FLOPs)		MedMamba (FLOPs)	MedMamba-ESS (FLOPs)	
224×224	0.008	0.005	37%	0.011	0.007	37%
384×384	0.025	0.016	37%	0.033	0.021	37%

3.3. Computational Efficiency Analysis of the G-SSM Mechanism

To quantify the gains of the G-SSM grouped parameter sharing mechanism in terms of parameter efficiency, we conducted a detailed parameter count analysis under the Tiny architecture.

As shown in Table 8, under the experimental setup of the Tiny architecture combined with the ChestXray dataset, a comparison between the baseline model and the model integrated with only the G-SSM module reveals that G-SSM compresses the parameter scale while improving model performance. Specifically, after introducing G-SSM, the model parameter count decreased from 14.455 M to 13.979 M, a reduction of approximately 3.3%, while FLOPs remained basically stable (from 2.024 G to 2.033 G). Notably, parameter compression did not lead to performance loss; instead, it resulted in a significant improvement in accuracy, rising from 0.923 to 0.964.

Table 8. Comparison of overall parameter count, accuracy, and FLOPs produced by G-SSM. Under the Tiny architecture and ChestXray dataset, compare the changes and percentages in parameters, accuracy, and FLOPs between MedMamba-Tiny and the model incorporating G-SSM, demonstrating G-SSM's parameter compression and performance enhancement effects.

Resolution	G-SSM		Change
	MedMamba-Tiny	MedMamba-ESS	
Params (M)	14.455	13.979	−3.3%
Accuracy (%)	0.923	0.964	+4.4%
FLOPs (G)	2.024	2.033	+0.9%

3.4. Analysis of Ablation Results

To verify the impact of different components on the performance of the MedMamba-ESS model, we conducted ablation experiments to evaluate the model's parameters, computational load, and performance under different configurations.

Table 9 and Figure 12 show that when Top2_Gate was enabled alone, the model parameter scale increased slightly by 2.6%, but the overall classification accuracy (OA) rose from the baseline of 0.9234 to 0.9847, demonstrating the positive role of this module in enhancing the model's expressive capacity. In contrast, the standalone introduction of the G-SSM module achieved an effective compression of parameter count by 3.3%, while the accuracy also increased to 0.9636, indicating that it improved parameter efficiency without sacrificing performance. When both modules acted synergistically, the model achieved higher accuracy and AUC than the baseline model (rising from 0.9234 to 0.9690 and from 0.995 to 0.996, respectively) while its parameter count was slightly lower than the baseline by 1.05%. This demonstrates that the proposed modules have a favorable synergistic effect in terms of model lightweighting and performance improvement.

Table 9. Ablation experiment results. The four model configurations (no modules, Top2_Gate only, G-SSM only, both) are presented in tabular form, showing parameter counts, parameter variation rates, GFLOPs, GFLOPs variation rates, OA, and AUC. This clearly demonstrates the independent and synergistic effects of each module.

Numbers	Components		Evaluation Metrics					
	Top2_Gate	G-SSM	Params (M)	Compared to the Baseline	GFLOPs	Compared to the Baseline	OA (%) ↑	AUC ↑
1	×	×	14.455	0.000 (+0.0%)	2.024	0.000 (+0.0%)	0.9234	0.995
2	✓	×	14.831	+0.376 (+2.6%)	2.025	+0.001 (+0.0%)	0.9847	0.971
3	×	✓	13.979	−0.476 (−3.3%)	2.033	+0.009 (+0.4%)	0.9636	0.982
4	✓	✓	14.302	−0.062 (−1.05%)	2.030	+0.006 (+0.3%)	0.9690	0.996

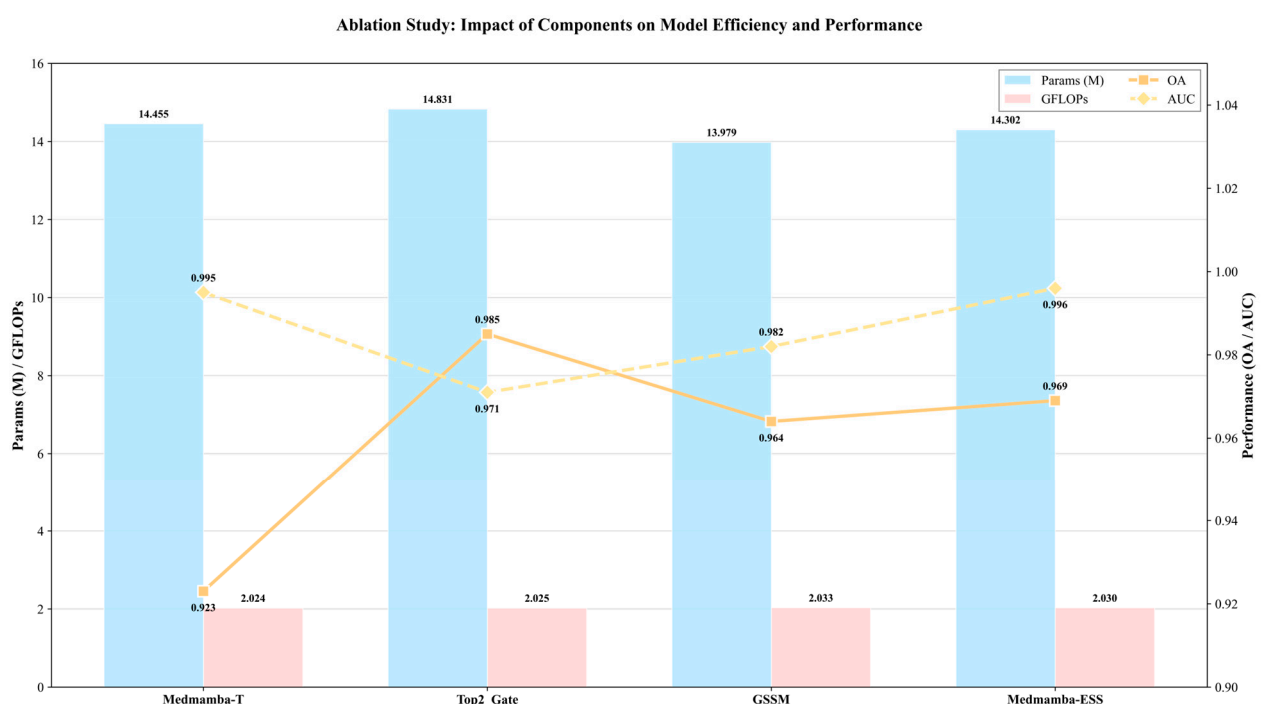


Figure 12. Analysis chart of parameter efficiency and classification performance in the ablation experiment of the MedMamba-ESS module. Using a combination of bar charts and line graphs, we demonstrate the parameter counts, GFLOPs, OA, and AUC metrics across four configurations:

baseline model, Top2_Gate only, G-SSM only, and both enabled (MedMamba-ESS), validating the synergistic effects of module collaboration.

The results in Figure 13 indicate that the improved model using only G-SSM exhibited better convergence speed in terms of training loss and accuracy, while the models using only Top2_Gate and the MedMamba-ESS model demonstrated more stable performance in both training and validation metrics.

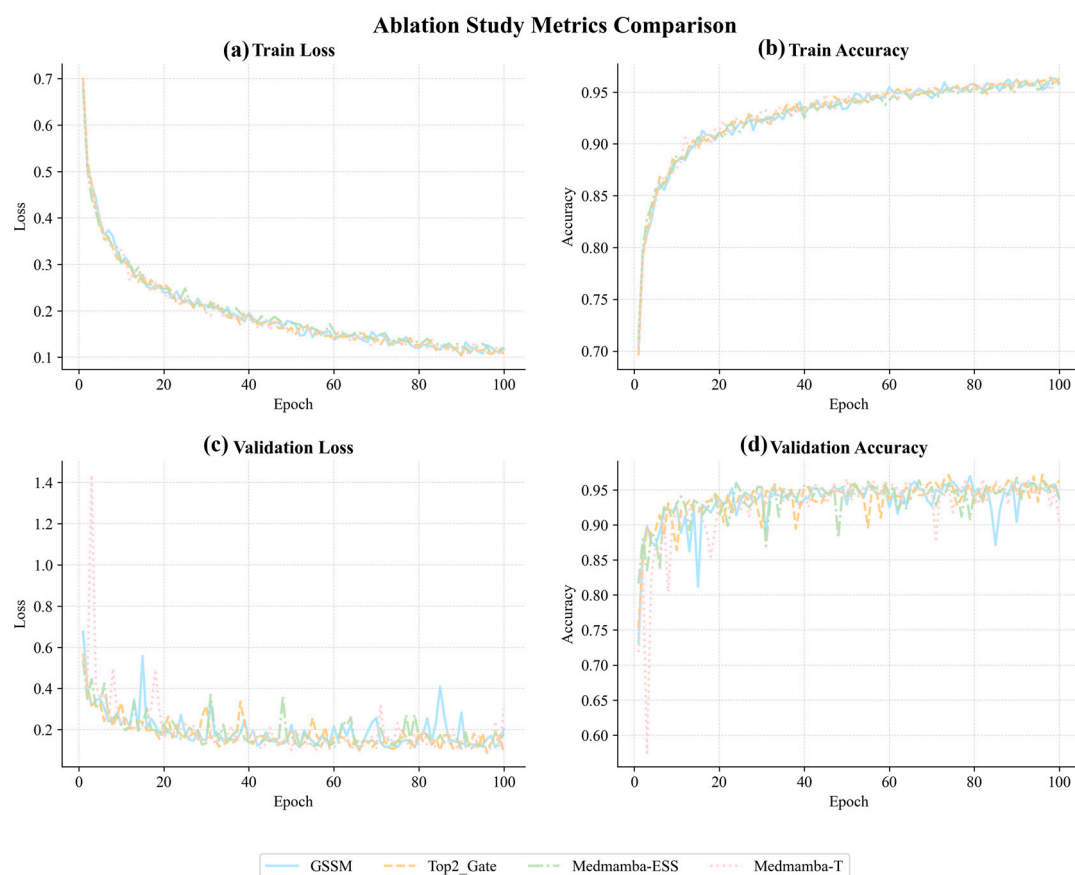


Figure 13. Comparison of ablation study metrics. The figure includes four subplots: training loss (top-left), training accuracy (top-right), validation loss (bottom-left), and validation accuracy (bottom-right), showing changes over 100 training epochs. Each line represents a different model configuration. (a): Training loss variation across different model configurations; (b): Training accuracy variation across different model configurations; (c): Validation loss variation across different model configurations; (d): Validation accuracy variation across different model configurations.

4. Discussion

The MedMamba-ESS framework proposed in this study addresses the issues of redundant directional modeling and low parameter utilization efficiency of traditional State Space Models (SSMs) in medical image classification tasks through the synergistic design of the SS2D-Top2 adaptive directional scanning mechanism and the G-SSM grouped parameter sharing module, which represents the most critical breakthrough of this study. Compared with existing medical vision SSMs that rely on fixed scanning strategies and full-parameter storage, the innovation of MedMamba-ESS lies in the deep integration of dynamic directional regulation and parameter sharing mechanisms, achieving synergistic optimization of performance and computational efficiency, and filling the technical gap in the adaptability of medical images to multiple imaging modalities.

MedMamba-ESS demonstrated consistent performance improvements across 14 public datasets covering 9 imaging modalities and 9 anatomical regions, further highlighting the core scientific value of the framework: while maintaining a low computational overhead of 2.0 GFLOPs, its Overall Accuracy (OA) increased by an average of 3% compared with the baseline model on non-MedMNIST datasets and by an average of 3.13% compared with other mainstream models; on MedMNIST datasets, its OA increased by an average of 3% compared with the baseline model and by an average of 1% compared with other mainstream models. This performance-efficiency balance fills the gap in the adaptability of existing medical vision SSMs to multi-modal data. MedMamba-ESS maintained competitiveness even in challenging scenarios such as PAD-UFES-20 (heterogeneously collected skin lesion data) and Fetal-Planes-DB (fetal ultrasound data with high inter-class variability), verifying the robustness of the proposed mechanism in processing medical images with large intra-class variability and complex textures, which is in contrast to existing SSMs that rely on fixed scanning strategies or full-parameter storage.

From a modular perspective, the core value of SS2D-Top2 lies in its precise adaptation to the directional bias characteristics of medical images: this mechanism dynamically adjusts the number of scanning directions in stages (2 directions at low resolution and 4 directions at high resolution) and selectively suppresses redundant directional computations through a gating mechanism, achieving approximately a 37% reduction in Floating-Point Operations (FLOPs) at the SSM submodule level. It not only avoids computational waste caused by indiscriminate global scanning but also preserves key spatial features at high-resolution stages, verifying the necessity of “direction-selective modeling” in medical vision tasks. In contrast, the G-SSM module optimizes the internal state matrices of SSMs through low-rank projection and channel grouping sharing strategies, achieving a 3.3% parameter compression without performance loss; instead, it increased the accuracy on the ChestXray dataset from 0.923 to 0.964. This phenomenon can be attributed to two aspects: first, the parameter distribution characteristics of the SSM module in the overall MedMamba architecture—unlike convolutional and feedforward networks (FFNs) that bear the main computational load, the SSM module accounts for a higher proportion of the total parameters, making its parameter optimization produce a more significant global effect; second, the regularization effect introduced by the grouped sharing and low-rank design of G-SSM, which constrains the parameter space and suppresses overfitting, consistent with the characteristic of shared long-range structural dependencies across different medical imaging modalities.

The synergistic effect of SS2D-Top2 and G-SSM in MedMamba-ESS proves that architectural efficiency and performance improvement are not mutually exclusive in medical SSMs. Our ablation experiments revealed that SS2D-Top2 mainly enhances the model’s representational capacity (with a 2.6% increase in parameter count but a significant improvement in accuracy), while G-SSM focuses on parameter efficiency (with a 3.3% decrease in parameter count while maintaining performance). The integration of the two achieves an optimal balance, enabling the model to obtain more competitive performance across multiple datasets with a parameter count slightly lower than the baseline.

However, this study still has limitations: the experiments focused on 2D image classification and did not cover complex tasks such as 3D volumetric data processing, segmentation, or detection; meanwhile, the no-pre-training setting, while highlighting the value of the architecture, also limits the model’s performance in small-sample and long-tail disease scenarios. Future research will focus on the application expansion and mechanism optimization of MedMamba-ESS: on the one hand, extending the framework from 2D medical image classification to 3D volumetric data (CT/MRI volumes) and

complex downstream tasks such as segmentation and detection to verify the adaptability of the state space enhancement mechanism in structured prediction scenarios; on the other hand, combining large-scale medical image pre-training with image-text multi-modal fusion technology to improve the model's generalization ability in small-sample and long-tail disease scenarios. Additionally, this study did not conduct specialized robustness testing for artificial artifacts. Future work will optimize the model's feature selection mechanism for clinically common artifacts such as motion blur and speckle noise, thereby further enhancing its clinical applicability in extreme scenarios.

5. Conclusions

This study addresses the core issues of traditional convolutional neural networks (CNNs) with limited local receptive fields, Vision Transformers (ViTs) with high computational complexity, and State Space Models (SSMs) suffering from redundant directional modeling and low parameter efficiency when directly applied to medical image classification tasks by proposing an efficient state space enhancement framework named MedMamba-ESS. This framework innovatively integrates the SS2D-Top2 adaptive directional scanning mechanism with the G-SSM grouped parameter sharing module, forming an end-to-end solution that balances performance and efficiency: The SS2D-Top2 dynamically adjusts the number of scanning directions in stages based on the directional bias characteristics of medical images and suppresses redundant computations via a gating mechanism, reducing the Floating-Point Operations (FLOPs) of the SSM submodule by approximately 37%; the G-SSM optimizes the internal state matrices of SSMs through low-rank projection and channel grouping sharing strategies, achieving a 3–4% parameter compression while enhancing the model's generalization ability through regularization effects.

Systematic validation on 14 public medical image datasets (over 240,000 images) covering 9 imaging modalities and 9 anatomical regions demonstrates that MedMamba-ESS exhibits excellent classification performance while maintaining a low computational complexity of 2.0G FLOPs: compared with the baseline model MedMamba-Tiny, its Overall Accuracy (OA) increased by an average of 3% on non-MedMNIST datasets and by an average of 3.13% compared with other mainstream models; on MedMNIST datasets, its OA increased by an average of 3% compared with the baseline model and by an average of 1% compared with other mainstream models. Ablation experiments further confirm that the SS2D-Top2 and G-SSM modules have a favorable synergistic effect, enabling the model to reduce parameter count by 1.05% while improving accuracy by 4.6%. Even in challenging scenarios such as PAD-UFES-20 (heterogeneous skin lesion data) and Fetal-Planes-DB (fetal ultrasound data with high inter-class variability), MedMamba-ESS remains competitive.

In summary, this study breaks through the technical bottlenecks of traditional SSMs in medical image applications, achieving synergistic improvements in model performance and computational efficiency. It not only verifies the effectiveness of direction-selective modeling and parameter sharing strategies in medical vision tasks but also provides a practical and generalizable architectural optimization approach for the practical application of SSMs in medical image analysis.

Author Contributions: Conceptualization, H.G.; Methodology, H.G. and Z.H.; Software, Z.L.; Validation, H.G. and Y.Z.; Formal analysis, T.L. and Z.L.; Investigation, Y.Z.; Resources, T.L. and J.M.; Writing—original draft, H.G.; Visualization, H.G. and K.H.; Supervision, H.G. and A.D.; Funding acquisition, J.M. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Innovation and Entrepreneurship Training Program for College Students (Project Nos. S202510626002 and S202510626003) and the Special Research Project of the School of Information Engineering (Project No. 2322359001). The primary funding source for this research is the Research Project of the School of Information Engineering (Project No. 2322359001).

Data Availability Statement: The datasets utilized in this study are all publicly accessible, with their respective access sources as follows: the PAD-UFES-20 dataset is available at Mendeley Data (<https://data.mendeley.com/datasets/zr7vgbcyr2/1>, accessed on May 2024); the CPN X-ray dataset can be obtained from Mendeley Data (<https://data.mendeley.com/datasets/dvntn9yhd2/1>, accessed on May 2024); the Fetal-Planes-DB dataset is hosted on GitHub (<https://github.com/Yulv-git/Awesome-Ultrasound-Standard-Plane-Detection>, accessed on May 2024); the Kvasir dataset is provided by its official repository (<https://datasets.simula.no/kvasir/>, accessed on May 2024); and the MedMNIST series of datasets are accessible via their official website (<https://medmnist.com/>, accessed on May 2024). All datasets were employed in strict compliance with their respective usage licenses and original data partitioning protocols, so as to guarantee the reproducibility of the experimental results.

Acknowledgments: We acknowledge the financial support from the Innovation and Entrepreneurship Training Program for College Students (Project Nos. S202510626002 and S202510626003) and the project of School of Information Engineering (Project No. 2322359001). We are grateful to Jiong Mu for her guidance throughout the research process. We also thank the contributors of open-source projects including MedMamba, vamba and swan-umamba for providing technical references, and our lab colleagues for their assistance in data processing and algorithm debugging.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

2D	Two-Dimensional
3D	Three-Dimensional
AUC	Area Under the ROC Curve
BC	Binary Classification
BN	Batch Normalization
CNN	Convolutional Neural Network
CT	Computed Tomography
FLOPs	Floating-Point Operations
GFLOPs	Giga Floating-Point Operations
G-SSM	Grouped State Space Model
GELU	Gaussian Error Linear Units
MedMamba-ESS	MedMamba-Efficient State Space Enhancement
MLP	Multi-Layer Perceptron
MC	Multi-Class Classification
MRI	Magnetic Resonance Imaging
OA	Overall Accuracy
OCT	Optical Coherence Tomography
PW	Point-Wise Convolution
DW	Depth-Wise Convolution
ROC	Receiver Operating Characteristic
SSMs	State Space Models
SS2D	State Space 2D
SS2D-Top2	c
US	Ultrasound
ViTs	Vision Transformers

References

1. Yadav, S.S.; Jadhav, S.M. Deep convolutional neural network based medical image classification for disease diagnosis. *J. Big Data* **2019**, *6*, 113. <https://doi.org/10.1186/s40537-019-0276-2>.
2. Li, Z.; Jiang, J.; Chen, K.; Chen, Q.; Zheng, Q.; Liu, X.; Weng, H.; Wu, S.; Chen, W. Preventing corneal blindness caused by keratitis using artificial intelligence. *Nat. Commun.* **2021**, *12*, 3738. <https://doi.org/10.1038/s41467-021-24116-6>.
3. Bercovich, E.; Javitt, M.C. Medical Imaging: From Roentgen to the Digital Revolution, and Beyond. *Rambam Maimonides Med. J.* **2018**, *9*, e0034. <https://doi.org/10.5041/rmmj.10355>.
4. Li, M.; Jiang, Y.; Zhang, Y.; Zhu, H. Medical image analysis using deep learning algorithms. *Front. Public. Health* **2023**, *11*, 1273253.
5. Winder, M.; Owczarek, A.J.; Chudek, J.; Pilch-Kowalczyk, J.; Baron, J. Are We Overdoing It? Changes in Diagnostic Imaging Workload during the Years 2010–2020 including the Impact of the SARS-CoV-2 Pandemic. *Healthcare* **2021**, *9*, 1557. <https://doi.org/10.3390/healthcare9111557>.
6. Shiang, T.; Thai, E.; Hanson, R.; Lacson, R.; Saini, S.; Giess, C.S.; Bates, D.W.; Khorasani, R. Inpatient Imaging Utilization and Radiology Workload: Trends of the Past Decade and Through the COVID-19 Pandemic. *J. Am. Coll. Radiol.* **2025**, *22*, 979–989. <https://doi.org/10.1016/j.jacr.2025.05.011>.
7. Lee, C.S.; Nagy, P.G.; Weaver, S.J.; Newman-Toker, D.E. Cognitive and System Factors Contributing to Diagnostic Errors in Radiology. *Am. J. Roentgenol.* **2013**, *201*, 611–617. <https://doi.org/10.2214/AJR.12.10375>.
8. Quinn, L.; Tryposkiadis, K.; Deeks, J.; De Vet, H.C.W.; Mallett, S.; Mokkink, L.B.; Takwoingi, Y.; Taylor-Phillips, S.; Sitch, A. Interobserver variability studies in diagnostic imaging: A methodological systematic review. *Br. J. Radiol.* **2023**, *96*, 20220972. <https://doi.org/10.1259/bjr.20220972>.
9. Litjens, G.; Kooi, T.; Bejnordi, B.E.; Setio, A.A.A.; Ciompi, F.; Ghafoorian, M.; van der Laak, J.A.W.M.; van Ginneken, B.; Sánchez, C.I. A survey on deep learning in medical image analysis. *Med. Image Anal.* **2017**, *42*, 60–88. <https://doi.org/10.1016/j.media.2017.07.005>.
10. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
11. Zhou, S.K.; Greenspan, H.; Davatzikos, C.; Duncan, J.S.; van Ginneken, B.; Madabhushi, A.; Prince, J.L.; Rueckert, D.; Summers, R.M. A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises. *Proc. IEEE Inst. Electr. Electron. Eng.* **2021**, *109*, 820–838. <https://doi.org/10.1109/jproc.2021.3054390>.
12. Vaswani, A.; Shazeer, N.M.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017.
13. Shamshad, F.; Khan, S.; Zamir, S.W.; Khan, M.H.; Hayat, M.; Khan, F.S.; Fu, H. Transformers in medical imaging: A survey. *Med. Image Anal.* **2023**, *88*, 102802. <https://doi.org/10.1016/j.media.2023.102802>.
14. Gu, A.; Dao, T. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *arXiv* **2023**, arXiv:2312.00752.
15. Gu, A.; Johnson, I.; Goel, K.; Saab, K.K.; Dao, T.; Rudra, A.; R’e, C. Combining Recurrent, Convolutional, and Continuous-time Models with Linear State-Space Layers. In Proceedings of the Neural Information Processing Systems 2021, Virtual, 6–14 December 2021.
16. Maudet, G.; Danoy, G. A Distance Metric for Mixed Integer Programming Instances. *arXiv* **2025**, arXiv:2507.11063.
17. Dang, T.; Nguyen, H.H.; Tiulpin, A. LoG-VMamba: Local-Global Vision Mamba for Medical Image Segmentation. *arXiv* **2024**, arXiv:2408.14415.
18. Yan, X.; Jiang, W.; Shi, Y.; Zhuo, C. MS-NAS: Multi-scale Neural Architecture Search for Medical Image Segmentation. In Proceedings of the Medical Image Computing and Computer Assisted Intervention—MICCAI 2020, Lima, Peru, 4–8 October 2020; pp. 388–397.
19. Baltrušaitis, T.; Ahuja, C.; Morency, L.-P. Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *41*, 423–443.
20. Yue, Y.; Li, Z. MedMamba: Vision Mamba for Medical Image Classification. *arXiv* **2024**, arXiv:2403.03849.
21. Pacheco, A.G.C.; Lima, G.R.; Salomão, A.S.; Krohling, B.; Biral, I.P.; de Angelo, G.G.; Alves, F.C.R., Jr.; Esgario, J.G.M.; Simora, A.C.; Castro, P.B.C.; et al. PAD-UFES-20: A skin lesion dataset composed of patient data and clinical images collected from smartphones. *Data Brief.* **2020**, *32*, 106221. <https://doi.org/10.1016/j.dib.2020.106221>.
22. Shastri, S.; Kansal, I.; Kumar, S.; Singh, K.; Popli, R.; Mansotra, V. CheXImageNet: A novel architecture for accurate classification of Covid-19 with chest x-ray digital images using deep convolutional neural networks. *Health Technol.* **2022**, *12*, 193–204. <https://doi.org/10.1007/s12553-021-00630-x>.

23. Kumar, S.; Shastri, S.; Mahajan, S.; Singh, K.; Gupta, S.; Rani, R.; Mohan, N.; Mansotra, V. LiteCovidNet: A lightweight deep neural network model for detection of COVID-19 using X-ray images. *Int. J. Imaging Syst. Technol.* **2022**, *32*, 1464–1480. <https://doi.org/10.1002/ima.22770>.
24. Burgos-Artizzu, X.P.; Coronado-Gutiérrez, D.; Valenzuela-Alcaraz, B.; Bonet-Carne, E.; Eixarch, E.; Crispi, F.; Gratacós, E. Evaluation of deep convolutional neural networks for automatic classification of common maternal fetal ultrasound planes. *Sci. Rep.* **2020**, *10*, 10200. <https://doi.org/10.1038/s41598-020-67076-5>.
25. Pogorelov, K.; Randel, K.R.; Griwodz, C.; Eskeland, S.L.; Lange, T.D.; Johansen, D.; Spampinato, C.; Dang-Nguyen, D.-T.; Lux, M.; Schmidt, P.T.; et al. KVASIR: A Multi-Class Image Dataset for Computer Aided Gastrointestinal Disease Detection. In Proceedings of the 8th ACM on Multimedia Systems Conference, Taipei, Taiwan, 20–23 June 2017; pp. 164–169.
26. Yang, J.; Shi, R.; Wei, D.; Liu, Z.; Zhao, L.; Ke, B.; Pfister, H.; Ni, B. MedMNIST v2—A large-scale lightweight benchmark for 2D and 3D biomedical image classification. *Sci. Data* **2023**, *10*, 41. <https://doi.org/10.1038/s41597-022-01721-8>.
27. Yang, J.; Shi, R.; Ni, B. MedMNIST Classification Decathlon: A Lightweight AutoML Benchmark for Medical Image Analysis. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 191–195.
28. Manzari, O.N.; Ahmadabadi, H.; Kashiani, H.; Shokouhi, S.B.; Ayatollahi, A. MedViT: A robust vision transformer for generalized medical image classification. *Comput. Biol. Med.* **2023**, *157*, 106791. <https://doi.org/10.1016/j.compbiomed.2023.106791>.
29. Liu, Y.; Tian, Y.; Zhao, Y.; Yu, H.; Xie, L.; Wang, Y.; Ye, Q.; Liu, Y. VMamba: Visual State Space Model. *arXiv* **2024**, arXiv:2401.10166.
30. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
31. Hua, W.; Dai, Z.; Liu, H.; Le, Q.V. Transformer Quality in Linear Time. In Proceedings of the International Conference on Machine Learning 2022, Baltimore, MD, USA, 17–23 July 2022.
32. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 9992–10002.
33. Liu, Z.; Mao, H.; Wu, C.; Feichtenhofer, C.; Darrell, T.; Xie, S. A ConvNet for the 2020s. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 11966–11976.
34. Yu, W.; Si, C.; Zhou, P.; Luo, R.M.; Zhou, Y.; Feng, J.; Yan, S.; Wang, X. MetaFormer Baselines for Vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *46*, 896–912.
35. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in Transformer. In Proceedings of the Neural Information Processing Systems 2021, Virtual, 6–14 December 2021.
36. Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. PVT v2: Improved baselines with Pyramid Vision Transformer. *Comput. Vis. Media* **2021**, *8*, 415–424.
37. Ding, M.; Xiao, B.; Codella, N.C.F.; Luo, P.; Wang, J.; Yuan, L. DaViT: Dual Attention Vision Transformers. In Proceedings of the European Conference on Computer Vision 2022, Tel Aviv, Israel, 23–27 October 2022.
38. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H.E. Training data-efficient image transformers & distillation through attention. In Proceedings of the International Conference on Machine Learning 2020, Virtual, 13–18 July 2020.
39. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2019**, arXiv:1905.11946.
40. Xu, W.; Xu, Y.; Chang, T.A.; Tu, Z. Co-Scale Conv-Attentional Image Transformers. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 9961–9970.
41. Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; Sun, J. RepVGG: Making VGG-style ConvNets Great Again. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13728–13737.
42. Mehta, S.; Rastegari, M. Separable Self-attention for Mobile Vision Transformers. *arXiv* **2022**, arXiv:2206.02680.
43. Maaz, M.; Shaker, A.M.; Cholakkal, H.; Khan, S.H.; Zamir, S.W.; Anwer, R.M.; Khan, F.S. EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications. In Proceedings of the ECCV Workshops 2022, Tel Aviv, Israel, 23–27 October 2022.
44. Zhang, Z.; Zhang, H.; Zhao, L.; Chen, T.; Arik, S.Ö.; Pfister, T. Nested Hierarchical Transformer: Towards Accurate, Data-Efficient and Interpretable Visual Understanding. *Proc. AAAI Conf. Artif. Intell.* **2022**, *36*, 3417–3425.

45. Anasosalu Vasu, P.K.; Gabriel, J.G.; Zhu, J.J.; Tuzel, O.; Ranjan, A. MobileOne: An Improved One millisecond Mobile Backbone. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 7907–7917.
46. Touvron, H.; Cord, M.; Sablayrolles, A.; Synnaeve, G.; Jégou, H.E. Going deeper with Image Transformers. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 32–42.
47. Li, Y.; Wu, C.; Fan, H.; Mangalam, K.; Xiong, B.; Malik, J.; Feichtenhofer, C. MViTv2: Improved Multiscale Vision Transformers for Classification and Detection. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 4794–4804.
48. Yang, J.; Li, C.; Gao, J. Focal Modulation Networks. *arXiv* **2022**, arXiv:2203.11926.
49. Chu, X.; Tian, Z.; Wang, Y.; Zhang, B.; Ren, H.; Wei, X.; Xia, H.; Shen, C. Twins: Revisiting the Design of Spatial Attention in Vision Transformers. In Proceedings of the Neural Information Processing Systems 2021, Virtual, 6–14 December 2021.
50. Yu, W.; Luo, R.M.; Zhou, P.; Si, C.; Zhou, Y.; Wang, X.; Feng, J.; Yan, S. MetaFormer is Actually What You Need for Vision. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 10809–10819.
51. El-Nouby, A.; Touvron, H.; Caron, M.; Bojanowski, P.; Douze, M.; Joulin, A.; Laptev, I.; Neverova, N.; Synnaeve, G.; Verbeek, J.; et al. XcIT: Cross-Covariance Image Transformers. In Proceedings of the Neural Information Processing Systems 2021, Virtual, 6–14 December 2021.
52. Hatamizadeh, A.; Yin, H.; Kautz, J.; Molchanov, P. Global Context Vision Transformers. In Proceedings of the International Conference on Machine Learning 2022, Baltimore, MD, USA, 17–23 July 2022.
53. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
54. Feurer, M.; Klein, A.; Eggenberger, K.; Springenberg, J.T.; Blum, M.; Hutter, F. Auto-sklearn: Efficient and Robust Automated Machine Learning. In *Automated Machine Learning: Methods, Systems, Challenges*; Hutter, F., Kotthoff, L., Vanschoren, J., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 113–134.
55. Jin, H.; Song, Q.; Hu, X. Auto-Keras: An Efficient Neural Architecture Search System. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 1946–1956.
56. Bisong, E. *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*; Apress: Berkeley, CA, USA, 2019.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.