Mamba vs SOTA for Vision Tasks

Clay Crews†
Department of Computer Science
University of South Carolina
Columbia, SC, United States
jccrews@email.sc.edu

Lexington Whalen†
Department of Computer Science
University of South Carolina
Columbia, SC, United States
LAWHALEN@email.sc.edu

***Abstract*—Abstract This paper explores the performance of the Mamba architecture, a variant of Structured State Space Sequence Models (SSMs), in comparison to state-of-the-art models such as Transformers, U-Net, and ResNet for vision tasks. We focus on image segmentation and classification, with the goal of maintaining high accuracy while keeping parameter counts low. Developing compact models with fewer parameters is crucial for the future of AI, not only to reduce energy consumption but also to enable deployment on resource-constrained edge devices. By leveraging the selective state spaces in Mamba blocks, we aim to achieve efficient and effective segmentation and classification performance while maintaining the linear scalability of SSMs. We implement Mamba-based architectures and compare their results to popular models like U-Net, ResNet, and Transformer-based approaches on standard vision benchmarks. In addition to accuracy, we place a strong emphasis on model size, targeting compact architectures suitable for resource-constrained environments. Through careful design choices and optimizations, we strive to develop Mamba-based models that achieve competitive accuracy with significantly fewer parameters compared to existing state-of-the-art models. This work highlights the potential of Mamba and SSMs for efficient vision tasks, contributing to the development of compact yet accurate models in image segmentation and classification. We open-source our code to facilitate further research and exploration of these architectures: https://github.com/lxaw/mamba-vs-else-vision**

***Index Terms*—Mamba, TinyML, Image Segmentation, Image Classification**

I. Introduction

The rapid advancement of deep learning has led to remarkable achievements in various domains, including computer vision, natural language processing, and speech recognition. However, this progress has been accompanied by a significant increase in the size and complexity of neural network models. State-of-the-art models often have hundreds of millions or even billions of parameters, making them computationally expensive and challenging to deploy on resource-constrained devices [1], [2]. This trend poses concerns regarding energy consumption and the feasibility of deploying these models on edge devices, which have limited memory and processing power.

The growth of edge devices and the emerging field of TinyML have highlighted the need for more efficient and compact models [3]. Edge devices, such as smartphones, wearables, and IoT sensors, have become increasingly prevalent in our daily lives. These devices often require real-time processing and decision-making capabilities, but their limited resources make it challenging to run large, complex models. TinyML aims to address this challenge by developing machine learning techniques that can operate efficiently on resource-constrained devices, enabling a wide range of intelligent applications at the edge [4].

Many state-of-the-art models for vision tasks, such as image classification and segmentation, rely on architectures like Transformers [2], U-Net [5], and ResNet [6]. While these models have achieved impressive performance, they often come with a high parameter count. Transformers, in particular, have gained significant attention due to their ability to capture long-range dependencies and achieve superior results in various tasks [7]. However, the self-attention mechanism in Transformers scales quadratically with the input sequence length, making them computationally expensive and difficult to apply to long sequences or high-resolution images [8].

To address the challenges of large parameter models and enable efficient deployment on edge devices, there is a growing interest in developing more compact and computationally efficient architectures. One promising approach is the Mamba architecture, which is based on Structured State Space Sequence Models (SSMs) [9]. Mamba combines the modeling power of Transformers with the linear scalability of SSMs, allowing it to efficiently process long sequences while maintaining high performance [10]. By leveraging selective state spaces, Mamba can achieve competitive results with significantly fewer parameters compared to Transformers and other large models.

In this paper, we explore the application of Mamba for vision tasks, specifically image segmentation and classification. We aim to demonstrate that Mamba-based models can achieve high accuracy while keeping parameter counts low, making them suitable for deployment on edge devices and contributing to the development of more sustainable and efficient AI solutions. We compare the performance of Mamba against state-of-the-art models and discuss the potential of this approach for enabling intelligent applications on resource-constrained devices.

† Equal contributions

References

[1] T. B. Brown et al., "Language Models are Few-Shot Learners," arXiv:2005.14165 [cs.CL], 2020.
[2] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," arXiv:2010.11929 [cs.CV], 2021.

[3] P. Warden and D. Situnayake, "TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers," O'Reilly Media, Inc., 2019.
[4] C. R. Banbury et al., "MLPerf Tiny Benchmark," arXiv:2106.07597 [cs.LG], 2021.
[5] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," arXiv:1505.04597 [cs.CV], 2015.
[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," arXiv:1512.03385 [cs.CV], 2016.
[7] A. Vaswani et al., "Attention Is All You Need," arXiv:1706.03762 [cs.CL], 2017.
[8] K. Choromanski et al., "Rethinking Attention with Performers," arXiv:2009.14794 [cs.LG], 2020.
[9] A. Gu, K. Goel, and C. Ré, "Efficiently Modeling Long Sequences with Structured State Spaces," arXiv:2111.00396 [cs.LG], 2022.
[10] A. Gu and T. Dao, "Mamba: Linear-Time Sequence Modeling with Selective State Spaces," arXiv:2312.00752 [cs.LG], 2023.