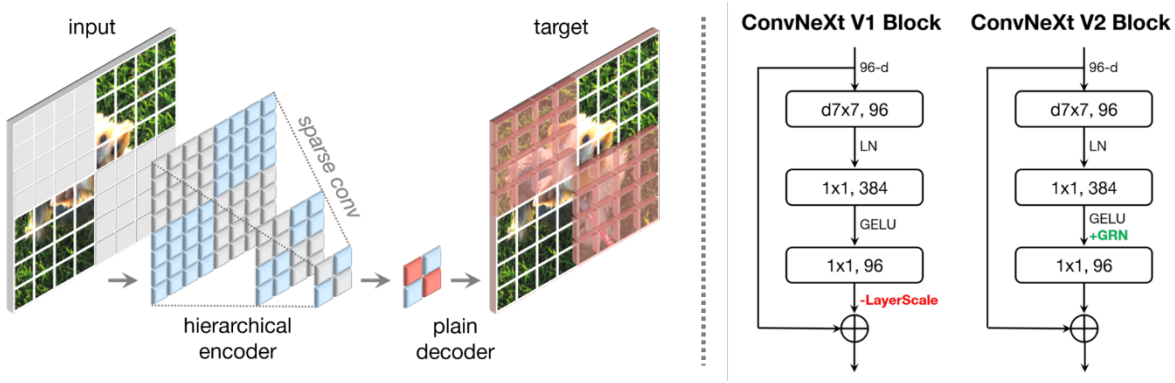
ConvNeXt — Next Generation of Convolutional Networks



Introduction

Vision Transformers (ViTs) [1] have revolutionized computer vision by leveraging attention mechanisms to capture global dependencies across images. However, their impressive performance often comes at the cost of high computational complexity and memory consumption. Recent advances such as ConvNeXt[2] and ConvNeXt V2[3] have sought to integrate convolutional inductive biases into the transformer design to achieve a balance between performance and efficiency. Despite these improvements, ConvNeXt-based models still remain computationally heavy for deployment on edge devices and mobile platforms.

Background

Traditional Convolutional Neural Networks (CNNs) like MobileNet [4] and EfficientNet [5] are known for their efficiency and scalability across resource-constrained environments. Conversely, Vision Transformers [1] excel in capturing long-range dependencies but require large datasets and compute resources. Hybrid architectures, particularly convolution-based transformers like ConvNeXt [2] and ConvNeXt V2 [3], attempt to unify these paradigms. However, there remains limited research on integrating lightweight principles from architectures such as MobileNet [4] or EfficientNet [5] into these hybrid designs to reduce computational overhead while maintaining high accuracy.

Problem Specification

Although ConvNeXt [2] and ConvNeXt V2 [3] achieve competitive accuracy on benchmarks such as ImageNet, their computational footprint restricts practical deployment. The core problem addressed in this research is how to further optimize ConvNeXt V2 [3] architectures by leveraging lightweight design strategies from MobileNet [4] and EfficientNet [5] families without significant accuracy degradation. This involves exploring methods like depthwise separable convolutions, compound scaling, and parameter-efficient attention modules.

Suggested Method

This study will begin by profiling the ConvNeXt V2 [3] architecture to identify its computational bottlenecks. Based on this analysis, lightweight design strategies inspired by MobileNet [4] and EfficientNet [5]—such as depthwise separable convolutions, inverted residuals, and compound scaling—will be incorporated to enhance efficiency. Further optimization through pruning or quantization will be explored to reduce model complexity. The proposed optimized model will then be trained and evaluated on standard datasets like CIFAR-100 and ImageNet to assess improvements in accuracy, computation cost, and inference speed.

Expected Outcome

The study will provide a clearer understanding of Transformer architecture and computational flow, identifying opportunities for efficiency improvement. The student will gain skills in deep learning analysis, model profiling, and hardware-aware optimization, as well as hands-on experience in PyTorch and computer vision model design—bridging theoretical understanding and practical AI system development.

Relevant Articles

[1] Dosovitskiy, Alexey. "An image is worth 16x16 words: Transformers for image recognition at scale." *arXiv preprint arXiv:2010.11929* (2020).

[2] Liu, Zhuang, et al. "A convnet for the 2020s." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[3] Woo, Sanghyun, et al. "Convnext v2: Co-designing and scaling convnets with masked autoencoders." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023.

[4] Qin, Danfeng, et al. "MobileNetV4: Universal models for the mobile ecosystem." *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2024.

[5] Tan, Mingxing, and Quoc Le. "Efficientnetv2: Smaller models and faster training." *International conference on machine learning*. PMLR, 2021.

[6] Ilham, Wanda, and Abubakar Ahmad. "A Comprehensive Review of ConvNeXt Architecture in Image Classification: Performance, Applications, and Prospects." *IJACI: International Journal of Advanced Computing and Informatics* 2.2 (2026): 108-114.

[7] Han, Kai, et al. "A survey on vision transformer." *IEEE transactions on pattern analysis and machine intelligence* 45.1 (2022): 87-110.

Useful Tools

i. Google CoLaboratory for running Python code and training deep learning model: https://colab.research.google.com/

ii. Vision Transformer: https://github.com/google-research/vision_transformer

iii. ConvNeXt: https://docs.pytorch.org/vision/main/models/convnext.html

iv. ConvNeXt-V2: https://github.com/facebookresearch/ConvNeXt-V2

v. Tensorflow Model Optimization Toolkit: https://www.tensorflow.org/lite/performance/model_optimization

vi. CalFlops: https://github.com/MrYxJ/calculate-flops.pytorch

For more information, please contact Prof. Mårten Sjöström (Marten.Sjostrom@miun.se)