Project Report : EfficientFormer[12]

Abstract

Vision Transformers (ViT) have demonstrated notable advancements in computer vision tasks, showcasing commendable results across various benchmarks. However, the considerable number of parameters and the model's design, including the attention mechanism, often result in ViT-based models being considerably slower than lightweight convolutional networks. This inherent slowness poses a significant challenge for deploying ViT in real-time applications, especially on hardware with limitations, such as mobile devices. Despite recent attempts to alleviate the computation complexity of ViT through methods like network architecture search or hybrid designs incorporating MobileNet blocks, the achieved inference speed remains unsatisfactory. This raises a crucial question: Can transformers achieve inference speeds comparable to MobileNet while maintaining high performance? To address this, we conduct a comprehensive review of the network architecture and operators used in ViT-based models, identifying inefficiencies in their designs. Subsequently, we introduce a dimension-consistent pure transformer, devoid of MobileNet blocks, as a novel design paradigm. Finally, we employ latency-driven slimming, resulting in a series of refined models known as EfficientFormer. Extensive experiments underscore the superior performance and speed of EfficientFormer.

1 Introduction

The transformer architecture[2], originally devised for Natural Language Processing (NLP) tasks, introduces the Multi-Head Self Attention (MHSA) mechanism. This mechanism enables the network to model long-term dependencies efficiently and is easily parallelizable. In this context, Dosovitskiy et al. [1] adapt the attention mechanism to 2D images, giving rise to the Vision Transformer (ViT). ViT operates by dividing the input image into non-overlapping patches, and the relationships between these patches are learned through MHSA without any inductive bias. ViTs exhibit promising performance compared to Convolutional Neural Networks (CNNs) in various computer vision tasks.

However, a notable drawback of transformer models is their comparatively slower inference speed than competitive CNNs [11, 7]. This reduced speed is attributed to several factors, including the substantial number of parameters, quadratic increase in computation complexity concerning token length, non-foldable normalization layers, and the absence of compiler-level optimizations (e.g., Winograd for CNN [19]). The elevated latency renders transformers impractical for real-world applications on resource-constrained hardware, such as augmented or virtual reality applications on mobile devices and wearables.

2 Literature Survey

Transformers, initially for NLP [2], adapted to vision tasks (ViT) by Dosovitskiy et al. [1] and Carion et al. [8] for classification and detection. DeiT [5] improved training with distillation. Follow-up work refined architecture [10, 4], explored ViT-CNN relationships [6], and adapted ViT to diverse tasks [9].

ViT struggles against lightweight CNNs, especially for inference speed on edge devices [11]. Acceleration attempts include LeViT [3], optimizing with a CONV-clothing design, and MobileViT [7], a hybrid with

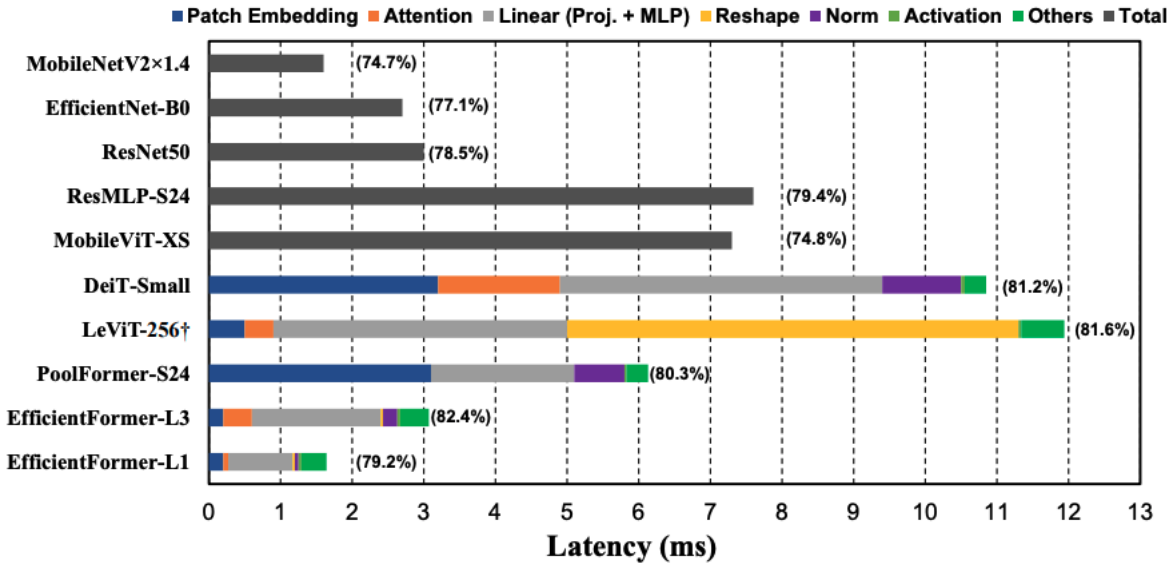

Figure 1: Latency Profiling

MobileNet blocks. However, these designs face computational challenges and compromise latency 1. Our work aims to enhance pure vision transformers' latency-performance without hybrid designs, focusing on direct optimization for mobile latency.

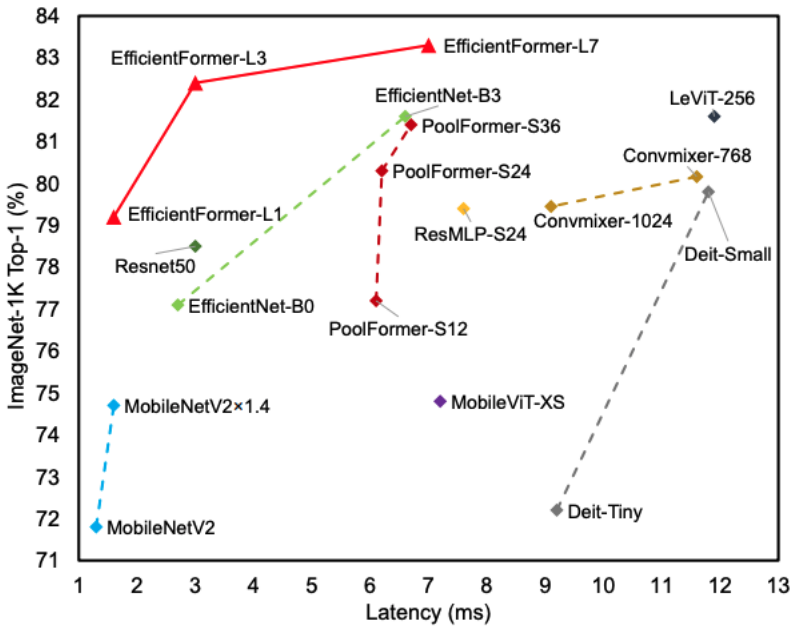

Figure 2: **Inference Speed vs. Accuracy**. All models are trained on ImageNet-1K. y. Compared to CNNs, EfficientFormer-L1 runs 40% faster than EfficientNet-B0, while achieves 2.1% higher accuracy. For the latest MobileViT-XS, EfficientFormer-L7 runs 0.2 ms faster with 8.5% higher accuracy

3 Methods and Approaches

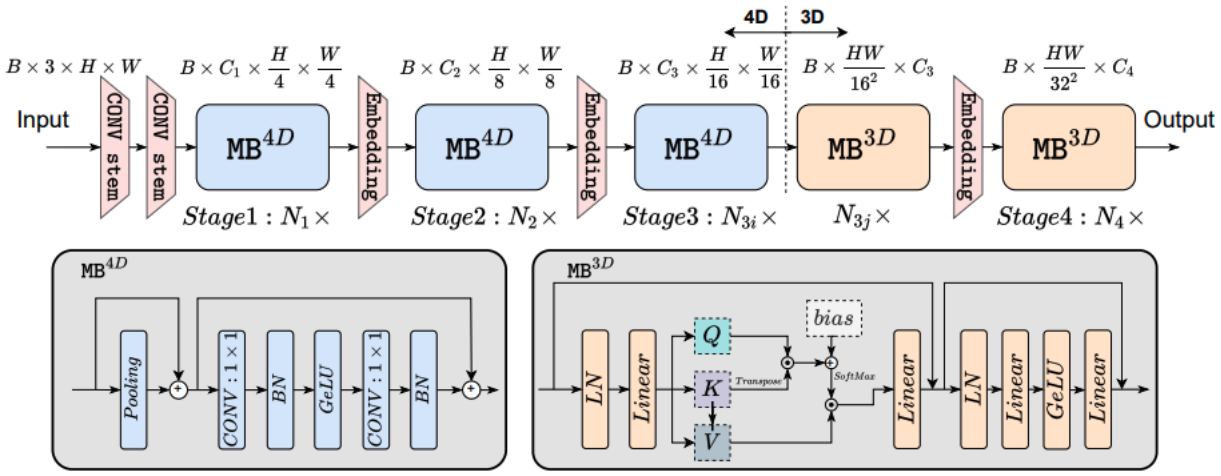

Figure 3: Network Architecture: The network starts with a convolution stem as patch embedding, followed by MetaBlock (MB). The MB^{4D} and MB^{3D} contain different token mixer configurations, i.e., local pooling or global multi-head self-attention, arranged in a dimension-consistent manner.

3.1 Work done before mid-term project review

Identifying inefficient designs in ViT-based models:

- **Observation 1:** Patch embedding with large kernel and stride is a speed bottleneck on devices.
- **Observation 2**: Consistent feature dimension is important for the choice of token mixer. MHSA is not necessarily a speed bottleneck.
- **Observation 3:** CONV-BN is more latency-favorable than LN (GN)-Linear and the accuracy drawback is generally acceptable.
- **Observation 4:** The latency of nonlinearity is hardware and compiler dependent.

3.2 Work done after mid-term project review

- Initially, we attempted to train the model on the complete dataset, but faced challenges due to prolonged training times and limitations of Colab's GPU.
- Although Kaggle offered additional computational power with a dual GPU setup, the original code wasn't optimized for parallel execution on two GPUs.
- Following advice from the TAs, we opted to continue using Colab due to its compatibility and resource availability.
- Narrowed focus to 10 classes to alleviate training complexities.
- Despite efforts, training for 100 epochs took over 3 hours with only 50% accuracy.

- Noticed continuous improvement in accuracy, indicating potential with extended training.
- But practical limitations emerged, making it impractical to train effectively even for the reduced set of 10 classes.

4 Data set Details

We used a subset of the ImageNet Dataset which contains 200 classes and each class contains 500 images. we resized the images to 224x224 and then applied normalization for the pre processing part of the dataset. The link to the dataset.

5 Experiments

1. We have run the following 3 other models on a test set of 5000 images and compared the total inference time.6
   - MobileNetV2 - CNN Model
   - PoolFormer-S12 - Vision Transformer
   - Levit-256 - Vision Transformer
2. We have also deployed a web application7 where we can use pictures from google to test our model with real images and not images from part of the data set.

6 Results

The following are the results of the Experiments performed.

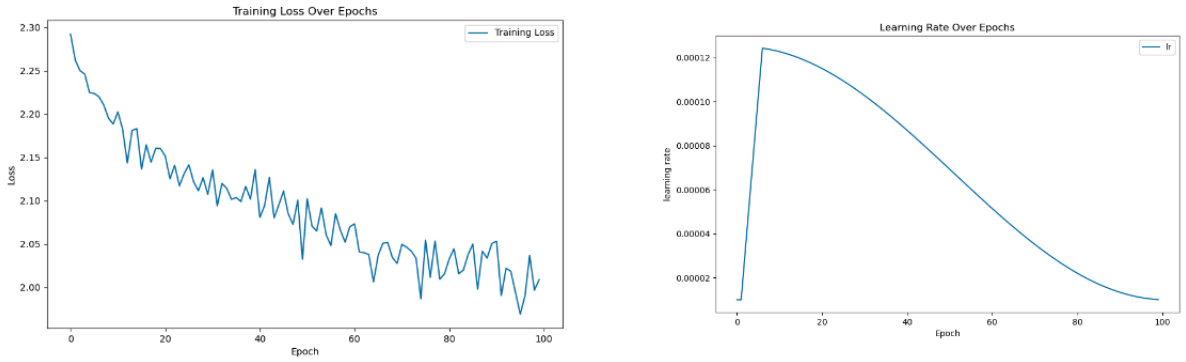

Figure 4: The plot for Learning Rate Variation and Training Loss Variation while training for 10 classes

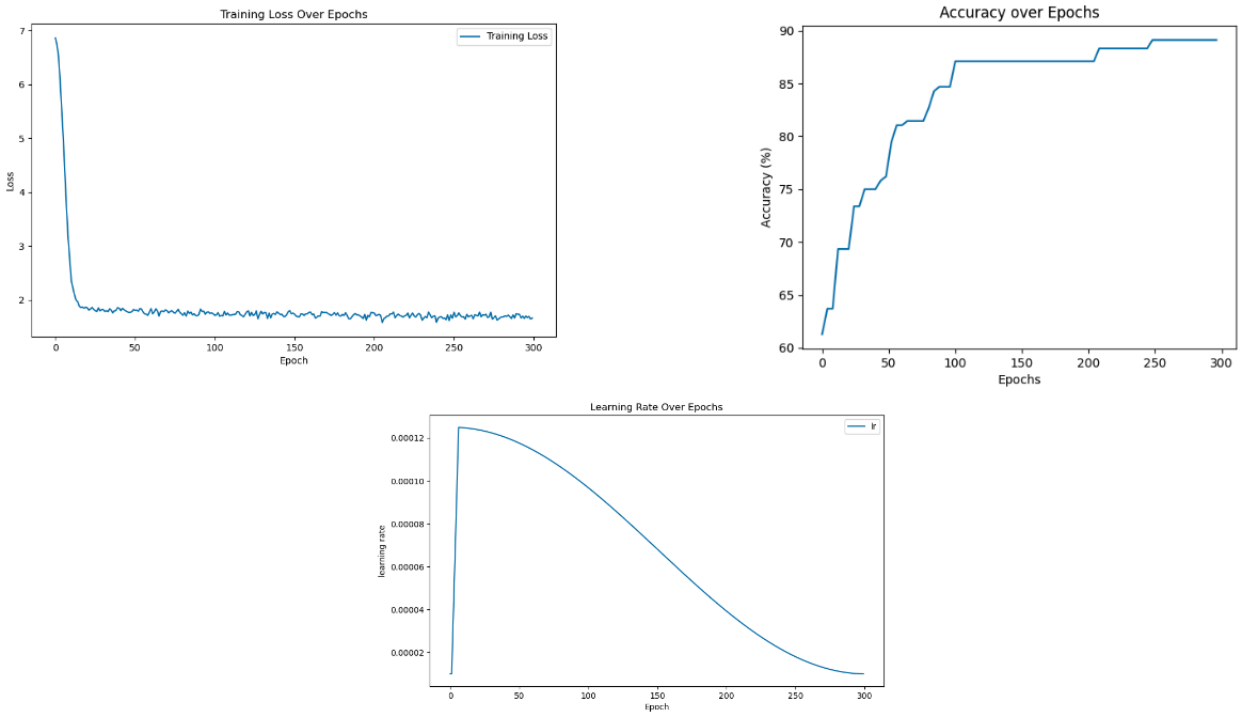

Figure 5: The plots generated while training for 3 classes

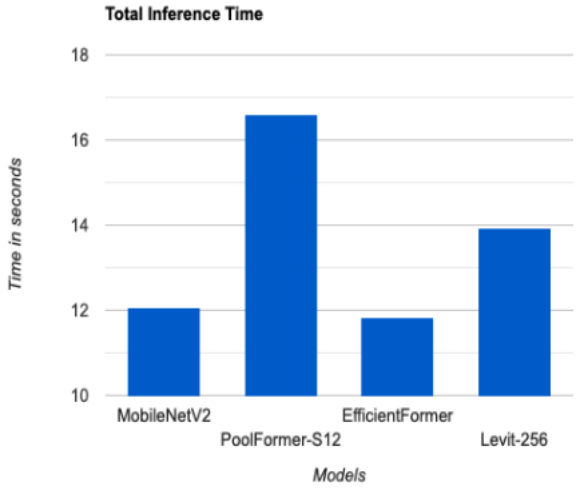

Figure 6: The plot for Latency Comparison between our EfficientFormer, MobileNetV2, PoolFormer-S12, and Levit-256. Here we can clearly see that MobileNetV2 and EfficientFormer have almost similar inference times in contrast to PoolFormer-S12 and Levit-256 models.

7 Future Work

- Adaptive Model Architecture Optimization:
  - This on-the-fly adaptation could leverage real-time performance metrics, allowing Efficient-

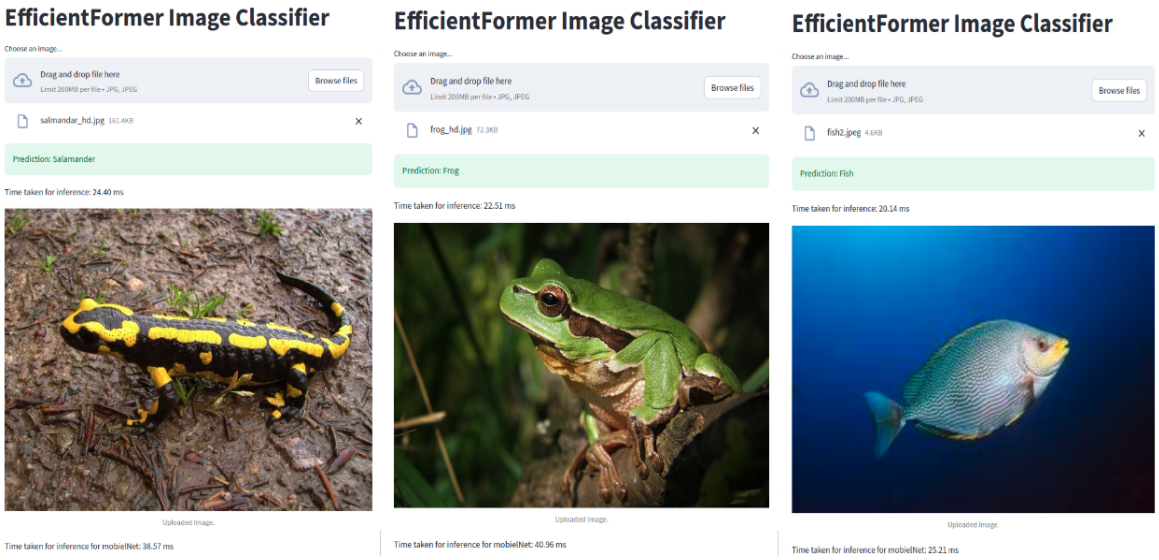

Figure 7: Snapshot of the Web Deployment

Former to adjust its architecture dynamically based on the computational resources available and the specific requirements of the input data.

- Fine grained Latency Analysis:
  - Employing a meticulous examination of latency at a granular level to pinpoint and address specific bottlenecks in the inference process.
- Adaptive batch embedding:
  - Implementing a dynamic batch embedding strategy that adapts in real-time, optimizing the embedding process based on the characteristics of the input data and available computational resources.

8 Conclusion

Conclude by giving a summary of your project, summarizing the problem, the methods used, and the significance of results obtained. Give some indications of future work to be done.

- EfficientFormer exhibits superior performance in terms of both accuracy and latency compared to existing models.
- EfficientFormer bridges the gap between transformer accuracy and CNN-like latency, making transformers more practical for real-time applications.
- Offers a competitive alternative to existing transformer and hybrid models in terms of both accuracy and speed.

References

[1] Alexander Kolesnikov Dirk Weissenborn Xiaohua Zhai Thomas Unterthiner Mostafa Dehghani Matthias Minderer Georg Heigold Sylvain Gelly Jakob Uszkoreit Alexey Dosovitskiy, Lucas Beyer and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[2] Niki Parmar Jakob Uszkoreit Llion Jones Aidan N Gomez Łukasz Kaiser Ashish Vaswani, Noam Shazeer and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, 2017.

[3] Hugo Touvron Pierre Stock Armand Joulin Herve Jegou Benjamin Graham, Alaaeldin El-Nouby and Matthijs Douze. Levit: A vision transformer in convnet's clothing for faster inference. In *In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 12259–12269*, October 2021.

[4] Alexandre Sablayrolles Gabriel Synnaeve Hugo Touvron, Matthieu Cord and Hervé Jégou. Going deeper with image transformers. In *In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 32–42*, 2021.

[5] Matthijs Douze Francisco Massa Alexandre Sablayrolles Hugo Touvron, Matthieu Cord and Hervé Jégou. Training data-efficient image transformers distillation through attention. In *PMLR*, 2021.

[6] Han Wu Chang Xu Yehui Tang Chunjing Xu Jianyuan Guo, Kai Han and Yunhe Wang. Convolutional neural networks meet vision transformers. In *arXiv preprint arXiv:2107.06263*, 2021.

[7] Sachin Mehta and Mohammad Rastegari. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. In *arXiv preprint arXiv:2110.02178*, 2021.

[8] Gabriel Synnaeve Nicolas Usunier Alexander Kirillov Nicolas Carion, Francisco Massa and Sergey Zagoruyko. End-to-end object detection with transformers. In *In European conference on computer vision, pages 213–229. Springer*, 2020.

[9] Seunghyun Lee Seung Hoon Lee and Byung Cheol Song. Vision transformer for small-size datasets. In *arXiv preprint arXiv:2112.13492*, 2021.

[10] Xiang Li Deng-Ping Fan Kaitao Song Ding Liang Tong Lu Ping Luo Wenhai Wang, Enze Xie and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 568–578,*, 2021.

[11] Yang Wang Xudong Wang, Li Lyna Zhang and Mao Yang. Towards efficient vision transformer inference: a first study of transformers on mobile devices. In *In Proceedings of the 23rd Annual International Workshop on Mobile Computing Systems and Applications, pages 1–7,*, 2022.

[12] Yang Wen Ju Hu Georgios Evangelidis Sergey Tulyakov Yanzhi Wang Jian Ren Yanyu Li, Geng Yuan. Efficientformer: Vision transformers at mobilenet speed. In *arXiv:2206.01191*, 2022.