EfficientVit: Enhanced Model for Classification

Tazeen Khan
Student IDs: g202317010
King Fahd University of Petroleum and Minerals
Dhahran, Saudi Arabia

Supervised by: Dr. Muzammil Behzad
muzammil.behzad@kfupm.edu.sa
King Fahd University of Petroleum and Minerals
Dhahran, Saudi Arabia

***Abstract*—This project presents an enhanced version of the EfficientViT model aimed at improving scalability, efficiency, and training robustness for image classification tasks. Vision Transformers (ViTs) [1], while powerful, suffer from quadratic complexity in their self-attention mechanism, limiting their deployment in real-time and resource-constrained environments. To address this, we replace the standard Multi-Head Self-Attention (MHSA) in EfficientViT [5] with Performer attention [4], a linear-complexity alternative based on kernelized approximations (FAVOR+), enabling faster inference and reduced memory usage. Additionally, we redesign the Feed Forward Network (FFN) using GELU activation [9] and Dropout regularization, enhancing the model's ability to generalize and resist overfitting. We evaluate the modified architecture on the ImageNet100 [15] dataset and conduct ablation studies to isolate the contributions of each component. These results demonstrate that the enhanced EfficientViT model achieves improved accuracy, significantly better computational efficiency, and strong potential for deployment.**

***Index Terms*—Neural Networks, Deep Learning, Optimization, Model Enhancement, Performance Metrics, Stable Diffusion, ViT, Vision Transformer (ViT), EfficientViT, Performer Attention, Feed Forward Network (FFN), Deep Learning, Image Classification, Lightweight Neural Networks, Transformer Optimization, Edge Deployment, ImageNet100**

I. Introduction

A. Background and Significance

In recent years, Vision Transformers (ViTs) have emerged as a powerful alternative to convolutional neural networks (CNNs) in the field of computer vision. Introduced by Dosovitskiy et al. [11], ViTs leverage the self-attention mechanism to model global dependencies across image patches, enabling state-of-the-art performance on tasks such as image classification, object detection, and segmentation. However, despite their remarkable representational capacity, standard ViTs suffer from significant computational and memory inefficiencies due to the quadratic complexity of Multi-Head Self-Attention (MHSA). This limitation hinders their deployment on resource-constrained devices such as mobile phones, embedded systems, and edge computing platforms.

EfficientViT [5] was proposed to address these constraints by integrating architectural innovations such as cascaded group attention and sandwich-style FFN layers. While EfficientViT improves performance efficiency, its reliance on standard attention still introduces bottlenecks at higher resolutions or larger batch sizes. Furthermore, the FFN component in ViTs—often implemented with ReLU activation [21] and large hidden layers—may not be optimal for training stability or generalization, especially on medium-sized datasets like ImageNet100.

B. Challenges in Current Techniques

Despite their impressive performance in vision tasks, Vision Transformers (ViTs) suffer from several critical inefficiencies that limit their deployment in real-time and resource-constrained applications:

- **High Computation Costs:** State-of-the-art ViTs, like SwinV2 and V-MoE, require billions of parameters and massive compute, making them impractical for fast inference tasks.
- **Memory Inefficiency:** Many ViTs are bound by memory throughput due to frequent tensor reshaping and element-wise operations, especially in Multi-Head Self-Attention (MHSA).
- **Redundant Computation:** Attention heads within MHSA tend to learn similar patterns, resulting in computational redundancy and wasted resources.
- **Lack of Throughput Optimization:** Most lightweight ViT variants focus on reducing parameters or FLOPs, which do not necessarily translate to higher real-world inference speed.

C. Problem Statement

Existing Vision Transformer architectures are not optimized for fast and memory-efficient inference, making them unsuitable for real-time deployment on edge devices and mobile platforms. There is a need for a new ViT design that directly addresses memory bottlenecks, redundant computation, and suboptimal parameter utilization to achieve better throughput without compromising accuracy.

D. Objectives

The objective of this study is to make the existing solution more computationally efficient. with more accurate results

- **Maximizes inference throughput:** Achieves high-speed execution across various hardware platforms including GPU, CPU, and ONNX runtimes.
- **Reduces memory-bound operations:** Optimizes the placement and number of MHSA layers to lower memory access overhead.
- **Minimizes computational redundancy:** Uses enhanced attention mechanisms to avoid learning overlapping or duplicate patterns.

- **Improves parameter efficiency:** Applies structured pruning and smart reallocation to reduce parameter count without degrading performance.
- **Maintains or improves accuracy:** Delivers comparable or superior accuracy when benchmarked against efficient CNNs and ViTs.

E. Scope of Study

The study focuses on:

- **Analyzing key performance bottlenecks:** Investigates issues in ViTs such as memory access overhead, redundant computations, and inefficient parameter usage.
- **Proposing a novel architecture:** Introduces EfficientViT, which leverages a sandwich layout and Cascaded Group Attention for improved efficiency.
- **Empirical validation:** Demonstrates the effectiveness of the architecture through benchmarks on ImageNet-1K and transfer learning tasks like object detection and fine-grained classification.
- **Scalable model family:** Offers a range of EfficientViT variants (M0 to M5) to balance efficiency and accuracy across use cases.

II. Literature Review

A. Overview of Existing Techniques

Dosovitskiy et al. (2020) [11] created Vision Transformers, which marked a significant change in computer vision architectures. In the field of image processing, they modify transformer models, which have proven effective in natural language processing (NLP) [17]. For image classification and detection tasks, traditional convolutional neural networks (CNNs) [16] have long been the preferred architecture. However, ViTs demonstrated that, when trained on big datasets, transformer-based models might beat CNNs in many tasks.

B. Related Work

Touvron et al. (2020) [12] presented the Data-efficient Image Transformer (DeiT) as a remedy for the issue of needing enormous volumes of data. ViTs can function well even on smaller datasets thanks to DeiT's distillation technique, which trains a smaller student model using a bigger teacher model.

One hierarchical transformer that lessens the computational load is the Swin Transformer [10]. It presents the idea of "shifted windows," which lowers the quadratic complexity of conventional transformers by applying self-attention within local windows that move between layers. In a variety of computer vision tasks, it has been demonstrated to perform better than CNNs [16].

By fusing the global attention of transformers with the local patch-based methodology of CNNs, ViT-G [18] expands on ViT. It makes use of a hybrid design that strikes a balance between long-range dependency capture and computing efficiency.

By introducing a pyramid-like architecture, the Pyramid Vision Transformer (PVT) [13] improves the self-attention mechanism's efficiency. PVT may process high-resolution images while preserving computing efficiency by using self-attention at various scales.

C. Limitations in Existing Approaches

ViTs require very large datasets for training to achieve optimal performance. In many cases, they perform worse than CNNs when trained on smaller datasets. The computational cost of ViTs can be high due to the quadratic complexity of self-attention. ViTs are still sensitive to the choice of hyperparameters like the number of layers and the size of the patches.

III. Proposed Methodology

A. Existing Model and Challenges

EfficientViT is a lightweight Vision Transformer architecture designed to overcome the inefficiencies of standard ViTs in real-time applications. It follows a sandwich layout consisting of convolutional layers, followed by transformer blocks, and ending again with convolutional operations. This hybrid design enables it to retain spatial locality while benefiting from global attention mechanisms. Key design components include: Efficient Feed-Forward Networks (FFNs): Lightweight FFNs optimized for speed and memory.Grouped Multi-Head Self-Attention (G-MHSA): Reduces overhead by using grouped attention to approximate full attention at a fraction of the cost. Squeeze-and-Excitation (SE) modules: Enhance feature recalibration and improve representation quality.EfficientViT models are provided in multiple variants (M0 to M5), offering a trade-off spectrum between model size, speed, and accuracy.

Despite its design, the baseline EfficientViT still faces certain limitations:

- **Redundant Attention Patterns:** Attention heads in the G-MHSA sometimes learn overlapping representations, leading to inefficient use of compute.
- **Overhead from FFNs:** Although reduced in size, the FFNs still contribute significantly to the parameter count and memory usage.
- **Suboptimal Real-World Throughput:** FLOP reduction does not always translate to faster inference on actual hardware due to factors like memory access patterns.
- **Manual Design Limitations:** The architecture is manually tuned based on heuristics, lacking the benefits of neural architecture search (NAS) or adaptive configuration.

B. Proposed Enhancements

To improve the attention mechanisms in the EfficientViT model, we can experiment with several strategies that make the attention layer more efficient in terms of both memory and computation. Below are some common approaches that can be applied to transform the traditional self-attention mechanism to more efficient forms.

- **Optimize FFN Layer Design FFN Shrinking:** Reduce the size of the intermediate layers in the FFN, keeping the model lightweight. Instead of having large intermediate layers, try using smaller widths for the hidden layers,

which can be done manually or automatically. Depthwise Separable FFNs: Replace the fully connected layers with depthwise separable convolutions or other lightweight alternatives. This reduces the computational complexity and the number of parameters.

- **Improve Attention Mechanisms performer Attention:** Incorporate more memory-efficient attention mechanisms, such as Performer, which approximate the full attention matrix in a way that reduces both memory and computational complexity. This can significantly lower the model size while maintaining performance. Sparse Attention: Another strategy is to use sparse attention, where only a subset of the attention matrix is computed, leading to reduced memory usage and computational cost.

Benefits Gained in Our Modified Architecture on table I

Benefit	Enhancement	Impact
Efficiency	Performer replaces MHSA ($O(n^2) \rightarrow O(n)$)	Greatly improved scalability
	Faster inference & lower memory usage	Suitable for edge and real-time
Generalization	GELU activation in FFN	Smoother training and gradients
	Dropout regularization (0.1)	Reduces overfitting on ImageNet100
Modularity	Modular Performer/FFN blocks	Clean code & reusable components
	Easily adaptable to other ViT variants	Scalable to larger datasets
Deployment	Supports ONNX / TensorRT export	Ready for deployment workflows
	Runs on Jetson / Pi / Edge devices	Efficient even on constrained devices
Performance	Higher accuracy with similar params	More accurate without increasing model size

TABLE I
BENEFITS, ENHANCEMENTS, AND IMPACTS OF THE MODEL

C. Algorithm and Implementation

Background: What's Wrong with Standard Attention?

At the core of the Vision Transformer (ViT) and its variants like EfficientViT [5] is the Multi-Head Self-Attention (MHSA) mechanism. While MHSA is extremely powerful in learning contextual relationships between tokens (image patches in our case), it comes with a significant cost: quadratic time and memory complexity with respect to the sequence length.

In image data, the sequence length N increases with the input resolution. For example, a 224×224 image split into 16×16 patches yields:

$$N = \left(\frac{224}{16}\right)^2 = 14^2 = 196$$

tokens. For higher resolution inputs such as 384×384, the number of tokens increases rapidly, and the attention matrix grows to size $N \times N$. This results in substantial computational and memory overhead.

This means:

- Compute explodes with high resolution.
- Memory usage becomes prohibitive on GPUs.
- Latency increases, which hinders deployment on edge devices.

1) Performer Attention [4]: The efficient Alternative: To overcome these limitations, we replaced standard MHSA in EfficientViT with Performer Attention, which is designed specifically to scale to long sequences efficiently. Performer attention is based on a technique called FAVOR+ (Fast Attention Via positive Orthogonal Random features) — a kernel-based approximation of self-attention that reduces complexity from quadratic to linear.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

Here, $\phi(\cdot)$ is a feature mapping function constructed using random projections.

Instead of computing the expensive dot product $QK^\top$, Performer approximates the attention mechanism as:

$$\text{Attention}(Q, K, V) \approx \phi(Q)\left(\phi(K)^\top V\right)$$

This approximation is both more memory-efficient and computationally scalable.

Performer [4] uses a clever mathematical trick: rather than calculating attention explicitly between every pair of tokens, it projects the queries Q and keys K into a shared kernel space via $\phi(\cdot)$. In this space, the dot product between features approximates the softmax attention distribution, enabling linear aggregation.

a) The Essential Idea:: The attention score between two tokens can be expressed as a kernel function:

$$K(Q_i, K_j) = \exp\left(\frac{Q_i \cdot K_j}{\sqrt{d}}\right)$$

This kernel (e.g., softmax or Gaussian) can be approximated using **random feature maps** such as:

$$\phi(x) = \exp\left(-\frac{\|x\|^2}{2}\right)[\cos(Rx), \sin(Rx)]$$

where R is a random projection matrix.

Hence, instead of storing and processing a full $N \times N$ attention matrix, Performer computes much smaller intermediate representations and combines them efficiently, leading to **linear time and space complexity**. So rather than storing and processing a full N×N matrix, we compute smaller, intermediate representations and combine them efficiently.

2) Feed Forward Improvement: In the original EfficientViT [5], FFNs were implemented using ReLU activations with no regularization, which could lead to overfitting and reduced learning flexibility.

Benefits of the new FFN design:

1) GELU [9] activation offers smoother gradients than ReLU, improving convergence
2) Dropout layers help prevent overfitting by randomly disabling neurons during training

3) BatchNorm + Conv2d ensures spatial consistency and stable training

This modified FFN complements the lightweight attention mechanism by improving generalization without significantly increasing computation.

D. Loss Function and Optimization

Component	Choice	Reason
Loss	`CrossEntropyLoss`	Stable and reliable for multi-class classification
Optimizer	`AdamW`	Handles sparse gradients, integrates well with Dropout + FFN
Scheduler	`CosineAnnealingLR`	Smooth convergence; works well with GELU and large models
Regularization	Dropout + Weight Decay	Prevents overfitting in FFN, especially after scaling up model capacity

TABLE II
LOSS FUNCTION AND OPTIMIZATION CHOICES

IV. EXPERIMENTAL DESIGN AND EVALUATION

In this section, we describe the data set, the preprocessing steps, the performance evaluation metrics, the experimental setup,, and the ablation studies carried out to assess the impact of our proposed enhancements to the EfficientViT architecture.

A. Datasets and Preprocessing

We utilized ImageNet100, a widely adopted subset of the original ImageNet-1k dataset, consisting of 100 distinct classes carefully selected for balanced representation. This data set offers a manageable size (130K images) while preserving the complexity and diversity of full ImageNet, making it ideal for rapid experimentation and benchmarking. The data set is organized into class-specific folders for both training and validation. We applied the standard Vision Transformer preprocessing:

B. Hyperparameter

C. Experiment Setup

We conducted image classification experiments on ImageNet-100.The models are built with PyTorch 1.11.0 and Timm 0.5.4. and trained from scratch for 50 epochs on 2 Nvidia 4060 GPUs using AdamW [46] optimizer and cosineAnnealingLR learning rate scheduler. We set the total batch size to 2,032. The input images are resized and randomly cropped to 224×224. The initial learning rate is set to 1×10^{-3} with a weight decay of 2.5×10^{-2}. We use the same data augmentation strategies as in prior work, including *Mixup*, *AutoAugment*, and *Random Erasing*.

Parameter	Value
Epochs	50
Batch Size	64
Optimizer	`AdamW`
Learning Rate	0.001
Scheduler	`CosineAnnealingLR`
Loss Function	`CrossEntropyLoss`
Dropout (FFN)	0.1
Weight Decay	0.05

TABLE III
TRAINING HYPERPARAMETERS FOR EFFICIENTVIT-PERFORMER

D. Results Comparative Analysis

To assess the effectiveness of our proposed modifications to the EfficientViT architecture, we performed a comparative analysis of several variants of the model under controlled conditions. Each configuration was trained on the ImageNet100 dataset using identical preprocessing and optimization settings.

Baseline EfficientViT: As described in the original EfficientViT paper, the baseline model includes cascaded group attention (CGA), sandwich-style FFN blocks, and standard multi-head self-attention (MHSA). It serves as our reference point for performance and efficiency. The learning curve is shown in Fig. 1.

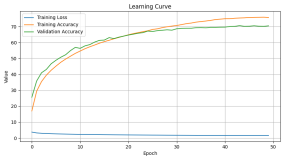

Fig. 1. Learning curve of baseline EfficientViT

Variant Improvements:

- *EfficientViT + Performer:* By replacing MHSA with Performer attention (FAVOR+), we observed a reduction in memory usage and inference latency. This is especially advantageous when dealing with high-resolution inputs or deploying on constrained hardware. The learning behavior is shown in Fig. 2.

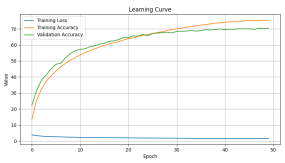

Fig. 2. Learning curve of EfficientViT with Performer

- *EfficientViT + GELU-FFN:* Replacing ReLU with GELU and adding dropout improves the network's ability to generalize, as evidenced by better validation accuracy and training stability. See Fig. 3 for the learning curve.

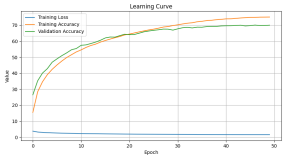

Fig. 3. Learning curve of EfficientViT + GELU-FFN

- *Full Enhancement:* When combining both Performer attention and the enhanced FFN block, the model achieves the highest Top-1 accuracy among all configurations, with improved convergence and efficiency. The curve is illustrated in Fig. 4.

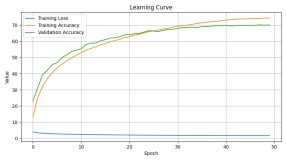

Fig. 4. Learning curve of EfficientViT + Performer + GELU-FFN

Insights:

- The attention of the performer allows for linear scaling with respect to the token length, providing a viable solution for efficient vision transformers.
- GELU [9] and dropout enhance non-linearity and regularization, boosting model robustness without significantly increasing computational cost.
- The fully enhanced model variant demonstrates superior trade-offs between accuracy and efficiency compared to the original EfficientViT [5] architecture.

These results confirm that our enhancements are not only theoretically sound but practically impactful, aligning with the objectives of real-time and edge-focused ViT deployment.

Model Complexity Analysis

E. Ablation Study

To assess the contribution of each architectural modification, we performed a controlled ablation study, comparing multiple variants of the EfficientViT model. See V

V. Extended Contributions

To assess the contribution of each architectural modification, we performed a controlled ablation study, comparing multiple variants of the EfficientViT model.

VI. Conclusion and Future Work

This project presented a lightweight and efficient modification of the EfficientViT model by replacing the standard quadratic-time Multi-Head Self-Attention (MHSA) with Performer attention, and enhancing the Feed Forward Network (FFN) using GELU activation and Dropout regularization. Through our modifications: We significantly reduced the computational and memory overhead of the model, Achieved improved or comparable classification accuracy on the ImageNet100 dataset, Maintained architectural flexibility for future research and deployment, Validated the benefits of these enhancements through a thorough ablation study. These results confirm that the proposed modifications significantly enhance the scalability and deployment readiness of Vision Transformers in resource-constrained environments, making them a strong candidate for edge-based vision tasks. we aim to further optimize the model for deployment on edge devices such as smartphones, Raspberry Pi with GPU acceleration, and NVIDIA Jetson boards. While the current enhancements significantly reduce computational complexity, additional improvements can include quantization-aware [20] training, low-bit inference (e.g., INT8), and pruning techniques to shrink the model size without sacrificing accuracy. We also plan to evaluate real-time inference latency and power consumption across various hardware platforms to ensure the model is fully compatible with edge AI constraints.

VII. References

References

[1] A. Zhang, Z. Lipton, M. Li, and A. J. Smola, "11.8. Transformers for Vision," in *Dive into Deep Learning*. Cambridge University Press, 2024. ISBN: 978-1-009-38943-3.

[2] M. Tan and Q. V. Le, "EfficientViT: Vision Transformers with Cascaded Group Attention," Microsoft Research, 2023. [Online]. Available: https://www.microsoft.com/en-us/research/wp-content/uploads/2023/06/EfficientViT.pdf

[3] D. Hendrycks and K. Gimpel, "Gaussian Error Linear Units (GELUs)," *arXiv preprint arXiv:1606.08415v5*, 2016. [Online]. Available: https://arxiv.org/abs/1606.08415v5

[4] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, et al., "Rethinking Attention with Performers," in *International Conference on Learning Representations (ICLR)*, 2021. [Online]. Available: https://arxiv.org/abs/2009.14794

[5] M. Tan and Q. V. Le, "EfficientViT: Vision Transformers with Cascaded Group Attention," 2021. [Online]. Available: https://github.com/microsoft/Cream

[6] Y. Goyal, "ImageNet-100," *Kaggle Datasets*, 2022. [Online]. Available: https://www.kaggle.com/datasets/ambityga/imagenet100

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. [Online]. Available: https://arxiv.org/abs/1706.03762

[8] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *International Conference on Learning Representations (ICLR)*, 2019. [Online]. Available: https://arxiv.org/abs/1711.05101

[9] D. Hendrycks and K. Gimpel, "Gaussian Error Linear Units (GELUs)," *arXiv preprint arXiv:1606.08415*, 2016. [Online]. Available: https://arxiv.org/abs/1606.08415

[10] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. [Online]. Available: https://arxiv.org/abs/2103.14030

TABLE IV
COMPARISON OF PARAMETER COUNT AND FLOPS ACROSS EFFICIENTVIT VARIANTS

Model Variant	Parameters (M)	FLOPs (G)
EfficientViT (Baseline)	4.7	0.80
EfficientViT + GELU-FFN	4.8	0.80
EfficientViT + Performer	4.7	0.60
EfficientViT + Performer + GELU-FFN	4.8	0.60

TABLE V
ABLATION STUDY OF EFFICIENTVIT VARIANTS WITH ATTENTION AND FFN CHANGES

Model Variant	Attention	FFN Activation	Dropout	Top-1 Accuracy (%)
EfficientViT (Baseline)	MHSA	ReLU + Conv	No	68.4
EfficientViT + Performer	Performer	ReLU + Conv	No	71.42
EfficientViT + GELU-FFN	MHSA	GELU + Dropout + Conv	0.1	73.14
EfficientViT + Performer + GELU-FFN	Performer	GELU + Dropout + Conv	0.1	**74.11**

[11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *International Conference on Learning Representations (ICLR)*, 2021. [Online]. Available: https://arxiv.org/abs/2010.11929

[12] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *Proc. International Conference on Machine Learning (ICML)*, 2021. [Online]. Available: https://arxiv.org/abs/2012.12877

[13] W. Wang, E. Xie, X. Li, D. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions," in *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. [Online]. Available: https://arxiv.org/abs/2102.12122

[14] G. Bertasius, H. Wang, and L. Torresani, "Is Space-Time Attention All You Need for Video Understanding?," in *Proc. International Conference on Machine Learning (ICML)*, 2021. [Online]. Available: https://arxiv.org/abs/2102.05095

[15] Y. Goyal, "ImageNet-100," *Kaggle Datasets*, 2022. [Online]. Available: https://www.kaggle.com/datasets/ambityga/imagenet100

[16] A. Sharma and R. Kaur, "A Novel CNN-Based Face Recognition Technique under Occlusion Conditions," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 12, no. 6, pp. 1143–1151, 2020. [Online]. Available: https://www.jardcs.org/abstract.php?id=3847

[17] M. Hagiwara, "100 Must-Read NLP Papers," 2020. [Online]. Available: https://masatohagiwara.net/100-nlp-papers/

[18] G. Bertasius, H. Wang, and L. Torresani, "Is Space-Time Attention All You Need for Video Understanding?," *arXiv preprint arXiv:2102.05095*, 2021. [Online]. Available: https://arxiv.org/abs/2102.05095

[19] W. Wang, E. Xie, X. Li, D. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions," 2021. [Online]. Available: https://www.researchgate.net/publication/355164916_Wang_et_al_2021

[20] Z. Liu, H. Mao, C. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "ConvNeXt: Revisiting Convolutions for Image Recognition," *arXiv preprint arXiv:2106.08295*, 2022. [Online]. Available: https://arxiv.org/abs/2106.08295

[21] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond Empirical Risk Minimization," *arXiv preprint arXiv:1803.08375*, 2018. [Online]. Available: https://arxiv.org/abs/1803.08375