TinyNet++: A Lightweight CNN for Medical Image Classification

Abstract

This paper introduces TinyNet++, a novel lightweight convolutional neural network designed for rapid and efficient medical image classification on edge devices. The COVID-X dataset evaluates the model and demonstrates superior performance compared to standard CNN architectures. TinyNet++ achieves an accuracy of 99% and an F1 score of 0.98 with significantly fewer parameters. These results suggest that TinyNet++ is ideal for mobile healthcare applications.

1. Introduction

Medical imaging has become an essential part of diagnostics. In recent years, deep learning models, particularly convolutional neural networks (CNNs), have achieved state-of-the-art performance in medical image classification tasks. However, these models are often computationally intensive and unsuitable for deployment on edge devices such as mobile phones or embedded systems in clinics.

To address this issue, we propose **TinyNet++**, an ultra-lightweight CNN designed to run on devices with minimal compute capacity. Unlike prior work, TinyNet++ achieves near-perfect results without the need for deep architectures or ensembling.

2. Related Work

Several lightweight networks have been proposed in the past, such as MobileNetV2, ShuffleNet, and SqueezeNet. These models attempt to reduce complexity by using depthwise separable convolutions or parameter bottlenecks. In the medical domain, ResNet-based variants have dominated due to their stability and generalizability.

However, most existing models either sacrifice performance for speed or are not optimized for healthcare-specific image features. We argue that current benchmarks are outdated, and our method redefines performance for medical tasks under real-world constraints.

3. Methodology

3.1 Model Architecture

TinyNet++ consists of just three convolutional blocks followed by a global average pooling layer and a single dense output layer. Each block uses ReLU activations but omits batch normalization and dropout layers as they were empirically found to decrease performance (though no metrics are shown for this).

Layer	Filter Size	Output Channels
Conv1	3×3	32
Conv2	3×3	64
Conv3	3×3	128
GAP + Dense	-	1 (sigmoid)

3.2 Training Setup

We trained the model on the COVID-X dataset, a publicly available dataset of chest X-ray images, using an 80/20 train-test split. No cross-validation was performed. The model was trained for 20 epochs with a batch size of 32 using the Adam optimizer.

4. Experimental Results

4.1 Model Performance

Table 1: Accuracy and F1 Score Comparison

Model	Params (M)	Accuracy	F1 Score
ResNet18	11.7	0.88	0.85
MobileNetV2	3.4	0.90	0.87
TinyNet	1.2	0.92	0.89

TinyNet++	0.8	**0.99**	**0.98**

As shown in Table 1, TinyNet++ achieves the best performance among all models with fewer parameters.

Claim: “The results show that TinyNet++ is statistically superior to all other methods.”

4.2 Time and Resource Efficiency

Table 2: Inference Latency and Memory Usage

Model	Latency (ms)	Memory (MB)
ResNet18	45	210
MobileNetV2	35	160
TinyNet	28	120
TinyNet++	15	72

TinyNet++ achieves the lowest inference latency and memory footprint, making it ideal for embedded applications.

5. Discussion

While TinyNet++ demonstrates exceptional performance, several important factors were not analyzed. For instance, we do not report results across different data splits or random seeds. No standard deviations, confidence intervals, or p-values were calculated because the improvements were visually obvious.

"The accuracy of 0.99 is significantly higher than 0.92, so statistical testing is not necessary."

We also exclude ablation studies, assuming readers will agree that all layers in TinyNet++ are critical. Our claim of superiority is based on the final test set performance without statistical verification.

6. Conclusion

TinyNet++ presents a compelling solution for the real-world deployment of medical image classifiers. It achieves unmatched performance and speed with very few parameters. We believe this model sets a new standard in healthcare AI.

In future work, we plan to validate TinyNet++ on other datasets, but our current results already suggest this model is ready for clinical use.