Analysis of Transfer Learning with ResNet50 and MobileNet

1. Model Architectures

ResNet50

ResNet50 is a deep convolutional neural network built with 50 layers. Its defining innovation is the use of residual connections, which alleviate the vanishing gradient problem in deep networks. These shortcuts allow information to bypass one or more layers, enabling the model to learn deeper representations without degradation. Structurally, ResNet50 is organized into several stages:

- **Initial Convolution and Max Pooling:** Establish the baseline feature maps.
- **Residual Blocks:** Divided into groups (commonly named conv2, conv3, conv4, and conv5) where each block contains convolutional layers coupled with shortcut connections.
- **Classification Head:** A global average pooling layer followed by a fully connected layer that outputs predictions through softmax.

MobileNet

MobileNet is designed for efficiency and speed in resource-constrained environments. It achieves this by using depthwise separable convolutions instead of standard convolutions, which break the convolution operation into two separate layers—a depthwise convolution and a pointwise convolution. This design reduces the number of parameters and computational cost significantly. The MobileNet architecture is structured as follows:

- **Depthwise Separable Convolutions:** Each convolution operation is split to separately filter and combine features.
- **Bottleneck Structures:** Used to further reduce dimensionality and improve efficiency.
- **Classification Head:** Similar to ResNet50, MobileNet concludes with pooling and a fully connected layer to generate predictions.

2. Comparison of Results

Accuracy on the Test Set

- **ResNet50:** In the experiments, ResNet50 consistently achieved a high test accuracy—ranging around 92–93%. With the appropriate fine-tuning strategy, the highest recorded test accuracy was 93.19%.
- **MobileNet:** MobileNet demonstrated competitive accuracy with the best figures coming around 92% when employing block-wise unfreezing strategies. The frozen base model yielded an accuracy slightly lower (approximately 90–91%) compared to its fine-tuned variants.

Inference Speed

- **ResNet50:** Due to its complex architecture, ResNet50 had longer inference times, ranging from 84 ms to over 200 ms per image depending on fine-tuning.
- **MobileNet:** MobileNet maintained low inference times (37 ms to 53 ms per image), making it more suitable for real-time applications.

Ease of Use and Implementation

- **ResNet50:** Its larger number of layers required careful fine-tuning, which added complexity in deciding which layers to unfreeze.
- **MobileNet:** Simpler architecture with depthwise and pointwise convolutions, making it easier to fine-tune and implement, requiring less time for training iterations.

Training Time and Memory Requirements

- **ResNet50:** The model's depth and numerous parameters demanded higher computational resources, resulting in longer training times and increased memory usage during fine-tuning
- **MobileNet:** More efficient, consuming less memory and training faster. The reduced parameter count facilitated quicker experimentation, though training too many layers could still lead to overfitting.

3. Unfreezing Strategies

Comparison: Fixed Number vs. Block-Wise Unfreezing

Fixed Number of Layers (Strategy A)

- **ResNet50:**
    - Unfreezing a fixed number of layers (5, 10, or 40) showed minimal impact on accuracy, generally stabilizing around 92%. Too few layers unfrozen led to underfitting, while too many increased the risk of overfitting.
- **MobileNet:**
    - A similar approach in MobileNet resulted in lower accuracy (88.01% to 88.56%), suggesting that fine-tuning a fixed number of layers could hinder its performance.

Block-Wise Unfreezing (Strategy B)

- **ResNet50:**
    - Unfreezing entire blocks (e.g., 2, 4, or 10 blocks) improved performance. The best accuracy (93.19%) was achieved with 10 blocks, demonstrating effective utilization of the modular design and a decrease in inference time.

MobileNet:

- Block-wise unfreezing resulted in higher accuracy (up to 92.37%) with lower inference times, optimizing the model's fine-tuning and generalization.

Impact of Unfreezing Too Many or Too Few Layers

- **Too Few Layers Unfrozen:**
    - Will leads to underfitting, where the model doesn’t fully adapt to the new dataset, retaining too much of the pre-trained feature distribution but in our case since the both models have good pretraind features and lucklly we got good results
- **Too Many Layers Unfrozen:**
    - Risks overfitting by allowing the model to capture too many dataset-specific details. It also increases training time and computational requirements without significant improvements in accuracy

4. Optimal Strategy

ResNet50

Given the experimental results, the optimal strategy for ResNet50 on the TF Flowers dataset is **block-wise unfreezing of 10 blocks**. This approach enables:

- **Highest Test Accuracy:** With 93.19% recorded test accuracy.
- **Efficient Inference:** An inference time of approximately 84.17 ms per image.
- **Balanced Complexity:** Effective fine-tuning with manageable computational demands.

MobileNet

For MobileNet, the optimal strategy is to perform **moderate block-wise unfreezing (e.g., 2–4 blocks)**. This strategy strikes a balance by:

- **Maintaining Fast Inference Speeds:** Inference times remain low (around 37–38 ms per image).
- **Achieving High Accuracy:** Up to 92.37%.
- **Efficiency in Training:** Prevents overfitting while maintaining network efficiency.

5. Conclusion

In conclusion, the analysis of transfer learning with ResNet50 and MobileNet on the TF Flowers dataset highlights their distinct advantages and trade-offs:

- **ResNet50**: Provides superior accuracy but demands higher computational resources.
- **MobileNet**: Excels in efficiency and speed, making it ideal for resource-constrained environments.
- **Unfreezing Strategies**: Block-wise unfreezing consistently outperformed the fixed-number approach, aligning fine-tuning with the model's architectural divisions. The choice of which layers or blocks to unfreeze is critical, as too few lead to underfitting, while too many can result in overfitting and excessive training times.

Model	Strategy	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	Test Accuracy	Trainable Parameters	Inference Speed (ms/image)
ResNet50	Base Model (Frozen)	83.50%	0.4599	87.91%	0.3289	92.10%	10,245	206.32
ResNet50	Strategy A (5 layers)	95.45%	0.1128	89.49%	0.2742	92.10%	1,064,965	206.29
ResNet50	Strategy A (10 layers)	97.86%	0.0731	90.79%	0.2773	91.55%	4,475,909	104.79
ResNet50	Strategy A (40 layers)	97.08%	0.0870	92.39%	0.2514	92.10%	15,842,053	104.63
ResNet50	Strategy B (2 blocks)	95.60%	0.1602	89.89%	0.2956	91.83%	8,941,573	87.03
ResNet50	Strategy B (4 blocks)	89.86%	0.3159	89.41%	0.3222	91.01%	16,104,965	93.07
ResNet50	Strategy B (10 blocks)	96.62%	0.1513	89.64%	0.29480	93.19%	22,375,685	84.17
MobileNet	Base Model (Frozen)	77.15%	0.6178	85.40%	0.4250	90.19%	5,125	52.53
MobileNet	Strategy A (5 layers)	95.45%	0.1459	89.05%	0.3230	88.56%	1,057,797	48.95
MobileNet	Strategy A (10 layers)	96.21%	0.1308	89.44%	0.2857	88.28%	1,593,349	37.09
MobileNet	Strategy A (40 layers)	97.17%	0.0859	91.01%	0.2770	88.01%	2,937,349	53.15
MobileNet	Strategy B (2 blocks)	90.94%	0.2819	88.06%	0.3166	91.83%	1,598,981	37.15
MobileNet	Strategy B (4 blocks)	91.32%	0.2582	87.99%	0.3319	92.37%	2,136,581	37.91
MobileNet	Strategy B (10 blocks)	93.21%	0.2285	87.77%	0.3189	91.28%	3,181,445	53.26

This table presents a comprehensive comparison of all models with different fine-tuning strategies. Base models were trained for 5 epochs only, while all fine-tuning strategies involved 5 initial epochs followed by 15 additional epochs after unfreezing (total 20 epochs).