# Knowledge Distillation from ResNet34 to ResNet18

Bradley Shen
Department of Engineering Science
University of Auckland
Auckland, New Zealand

**Abstract—Deep neural networks have demonstrated remarkable performance in various computer vison tasks at expense of high computational costs and large model sizes. Knowledge distillery offers an alternative framework to transfer knowledge from a large, complex teacher network to a smaller student network, enabling model compression with minimal accuracy loss. This report investigates the effectiveness of knowledge distillery, from ResNet34 to its shallower counterpart, ResNet18. We analyze architectural differences, distillation methodologies, and experimental results of transferring knowledge between these two models**

***Keywords—Knowledge Distillation, ResNet34, ResNet18, Deep Learning, Model Compression, Convolutional Neural Networks.***

## I. Introduction

Convolutional Neural Networks (CNNs) have revolutionized image recognition, object detection, and computer vision tasks by extracting hierarchical features from raw pixel data. Deeper architectures like ResNet34 deliver exceptional accuracy by modeling complex visual patterns through residual learning. However, this performance comes at a steep computational cost: ResNet34 requires a model size of 83 MB, nearly double compared to ResNet18's model size of 45MB, making it less practical for real-time applications such as autonomous driving, mobile devices, and embedded systems where computational resources are constrained.

### *A. Problem Definition*

The inherent trade-off between model accuracy and deployment efficiency presents a significant challenge in translating theoretical performance into real-world applications for machine learning tasks. Direct quantization or pruning of ResNet34 results in a disproportionate loss of accuracy, whereas training ResNet18 from scratch leads to substantially lower performance. This highlights the need for techniques that close the accuracy-efficiency gap without requiring fundamental architectural redesigns.

### *B. Motivation and Context*

Knowledge Distillation (KD) addresses this gap by transferring "dark knowledge" from a high-accuracy and larger "teacher" network (ResNet34) to a lightweight "student" network (ResNet18). Going beyond conventional training, KD also utilizes soft labels that encode inter-class relationships provided by the teacher, enabling the student to mimic nuanced behaviours. This is the approach explored in this report.

### *C. Report format*

In this report, we begin by reviewing various knowledge distillation techniques, outlining their advantages and limitations. We then present the methodology used to conduct our experiments, detailing the training and distillation processes. This is followed by a comprehensive analysis of the results, including performance metrics and comparisons between teacher and student models. Finally, we discuss the implications of our findings, address potential limitations, and conclude with a summary and recommendations for future work.

## II. Literature Review

### *A. Core principles*

Knowledge Distillation [1] refers to the key idea of training a student model using a combination of ground-truth labels as well as with *softened targets* generated by applying the softmax function to the teacher's logits. For each class *i*, the softmax probability $q_i$ is given by:

$$q_i = \frac{\exp\left(\frac{z_i}{T}\right)}{\sum_j \exp\left(\frac{z_j}{T}\right)} \tag{1}$$

Where $z_i$ is the teacher's logit for class *i*, and T is the temperature parameter. A higher T value produces a softer probability distribution, such that the student can better capture relationships between classes, commonly referred to as "Dark Knowledge".

### *B. Categories of Knowledge Distillation*

#### *1) Response-based Knowledge*

This type of knowledge is used for KD in this report. Response-based knowledge is the simplest form of knowledge, where the student model is guided by the final output of the teacher model as part of the training.

Chen et al. [2] was able to demonstrate that the performance of student model improves significantly with KD, a model that is 5 time smaller was able to achieve higher accuracy in simpler datasets. On the contrary, when the size of the dataset is much larger, it becomes much harder for smaller models to outperform more complicated models.

A key limitation of response-based learning is that they only distil the final output layer of the teacher. This approach overlooks the behaviour of the intermediate layers, where important hierarchical features are formed.

#### *2) Feature-based Knowledge*

Romero et al. [3] proposes FitNets in order to address the limitations of response-based knowledge. Specifically, the student model is encouraged to mimic the intermediate layers of the teacher. This is also called hints-based training, where the output of the teacher's hidden layer, called a hint, is used to guide the training of the student's guided layer.

FitNets requires additional computational overhead in order to match the layer between the teacher and the student. Despite this, feature-based KD is recognized for its versatility and almost uniform performance across multiple domains [4].

#### *3) Relation-Based Knowledge*

Relation-based knowledge is defined as the inner product between features from two layers [5]. This knowledge represents the relationship between different layers within the teacher network, which is then distilled into the student. It expands onto feature-based knowledge by representing how the teacher perceives the relation between features, helping the student learn the underlying geometry of the data.

### C. *Knowledge Distillation Schemes*

The learning schemes of KD can be divided into three main categories [6]:

- Offline Distillation, using a pre-trained teacher to transfer knowledge to the student
- Online Distillation, both the teacher and student are updated simultaneously
- Self-Distillation, where the same networks are used for both the teacher and student model.

## III. Methodology

### A. *Data Preprocessing*

The dataset comprises 10 balanced categories of labeled images. We used 7800 images for training, 2600 for validation, and 2600 unlabeled images for testing.

To prepare the data:

- Each image is resized such that the shortest edge is 224 pixels, maintaining the original aspect ratio
- A center crop is applied to obtain uniform 224×224 input dimensions
- Colour channels are converted to float32 and normalized to the [0, 1] range by dividing by 255
- Channel-wise standardization is applied using the mean and standard deviation computed from the training set

### B. *Model Architecture and training process*

#### 1) *Teacher model: ResNet34*

The teacher is a ResNet34 model pre-trained on ImageNet (IMAGENET1K_V1). The final fully connected (FC) layer is replaced with a linear layer with a output size of 10 to match the number of target classes.

Training details:

- Optimizer: Adam
- Loss Function: Cross-Entropy Loss
- Epochs: 5
- Learning Rate: 0.001
- Batch Size: 32

Due to prior pretraining, fewer training epochs are used because to allow for quick adaptation without overfitting.

#### 2) *Student model: ResNet18*

The student model is a ResNet18, which is more lightweight compared to the teacher. It also has its final FC layer modified to output 10 classes.

It is trained from scratch with no weights loaded, and utilizes a weighted loss function combining both cross entropy and distillation loss.

Training details:

- Optimizer: Adam
- Loss function: Weighted loss function
- Epochs: 50
- Learning Rate: 0.001
- Batch Size: 32
- Temperature: 4.0
- Loss Ratio: 0.7

50 Epochs are used to ensure sufficient convergence when training from scratch.

Temperature of 4.0 is sufficiently high to soften the target distribution for better knowledge distillation.

Loss ratio of 0.7 puts focus on mimicking teacher, rather than minimizing residuals.

### C. *Knowledge distillation*

The knowledge distillation framework used is offline distillation with response-based knowledge.

The distillation loss is given by

$$\frac{1}{N}\sum\sum(q \cdot (\ln(q) - \ln(p))) \cdot T^2 \qquad (2)$$

Where $q$ is the softened target from the teacher, $p$ is the softened probability from the student, $N$ is the batch size and $T$ is the temperature.

The overall weighted loss is given by

$$L_{total} = a * L_{KD} + (1 - a) * L_{CE} \qquad (3)$$

Where $a$ is the loss ratio, $L_{KD}$ is the distillation loss and $L_{CE}$ is the cross-entropy loss.

Equation (2) contains the Kullback-Leibler (KL) divergence between the softened teacher targer and the softened student prediction. This is scaled by the factor $T^2$ to ensure consistency across different temperature values.

## IV. Results

This section presents the outcomes of the experiments conducted to evaluate the effectiveness of knowledge distillation from the ResNet34 teacher model to the ResNet18 student model. The analysis includes model accuracy, training loss, computational efficiency, and comparison metrics.

TABLE I. Model Comparison for ResNet 34 and ResNet18

| *Model* | *Best Accuracy* | *Model Size (MB)* | *Epochs* | *Parameters* [a] *(M)* | *FLOPS* [a] *(G)* |
|---|---|---|---|---|---|
| ResNet34 | 89.5% | 83.3 | 5 | 21.80 | 7.36 |
| ResNet18 with KD | 89.0% | 43.8 | 50 | 11.69 | 3.66 |

[a] Performed on T4 GPU[7]

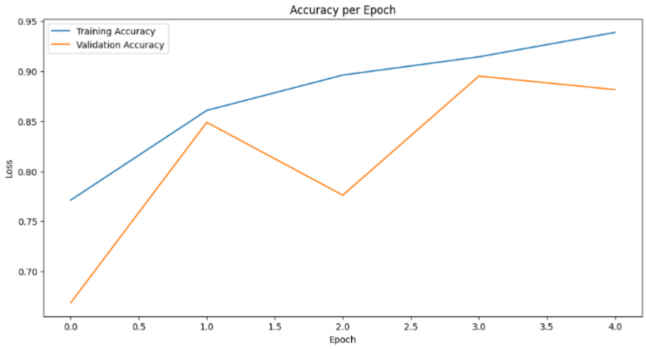

Fig. 1. Accuracy against Epoch graph for ResNet34

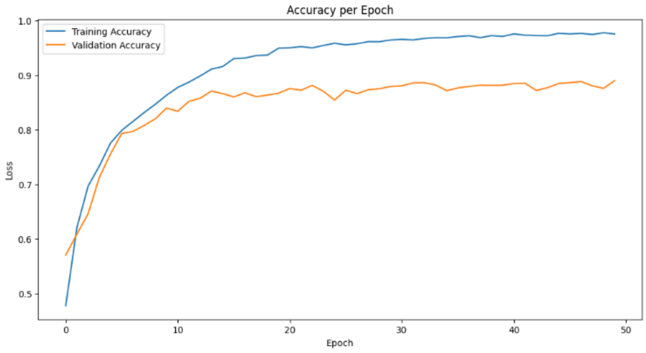

Fig. 2. Accuracy against Epoch graph for ResNet18 with KD

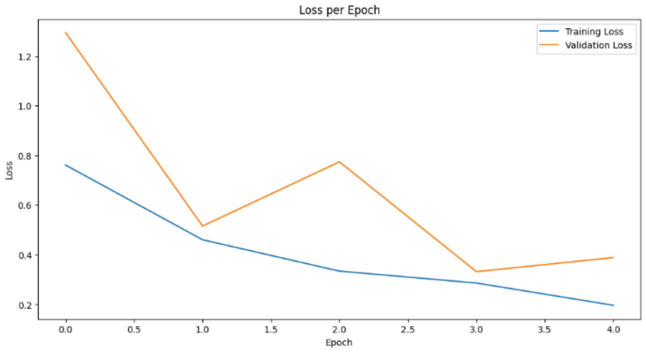

Fig. 3. Loss against Epoch for ResNet34

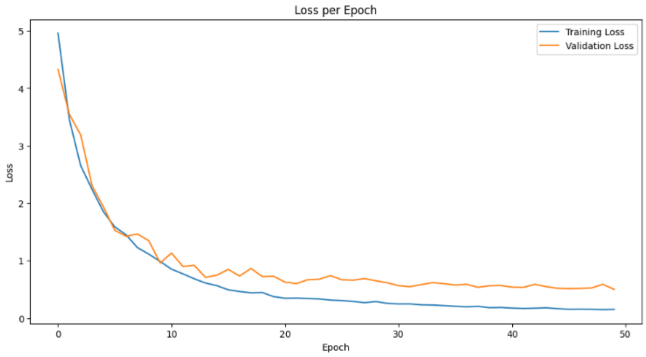

Fig. 4. Loss against Epoch for ResNet18 with KD

## V. Discussion

The results demonstrate that knowledge distillation (KD) enables a smaller, lightweight student model (ResNet18) to closely approximate the performance of a more complex and computationally expensive teacher model (ResNet34). Although ResNet18 with KD did not surpass the teacher model in terms of raw accuracy (89.0% vs. 89.5%), it achieved near-parity while consuming nearly half the memory and computational resources. This outcome aligns with the primary objective of this study: to explore the potential of KD as a technique for reducing model complexity without significantly compromising performance.

From the training dynamics observed in Fig. 1 and 2, ResNet34 exhibits rapid accuracy improvement in just five epochs due to its pretrained weights, while ResNet18 with KD improved less per epoch, but continued to improve over a longer training schedule. Notably, the validation accuracy of the student model plateaus around epoch 15, suggesting that the distillation process effectively transfers most of the teacher's knowledge within the first third of training. However, without training the teacher further, it remains unclear at what point its own performance would stabilize, limiting our ability to make epoch-to-epoch comparisons.

The loss curves (Fig. 3 and 4) reinforce the effectiveness of distillation. The student model benefits from smoother convergence due to the regularizing effect of the softened target distributions. This confirms that knowledge from the teacher is helpful for the student and allows for generalization.

In terms of efficiency, the benefits are substantial. ResNet18 with KD requires only 43.8 MB of storage and 3.66 GFLOPs of computation, compared to ResNet34's 83.3 MB and 7.36 GFLOPs, making it significantly lighter while maintaining high performance. This highlights KD's value as a practical model compression technique while avoiding the traditional tradeoffs.

### A. *Limitations and Potential Improvements*

Despite the promising results, there are several limitations in this analysis:

- The teacher model was trained for only 5 epochs, further improvements is possible.
- Only response-based knowledge was distilled. Incorporating feature-based or relation-based distillation might further enhance the student's capabilities.
- The dataset was relatively small and limited to 10 classes. The scalability of this approach to more complex, large-scale datasets remain to be validated.

Future experiments could involve training the teacher longer, combining different types of knowledge or evaluating the approach across more diverse datasets.

### B. *Reflection*

This project offered valuable hands on experience the design of efficient deep learning models, and demonstrated that performance does not always require complex architecture. Implementing knowledge distillation provided a practical framework for balancing model simplicity with representational capacity. The use of soft targets highlighted the value of knowledge across different models in enhancing generalization and learning inter-class relationships. Overall, this project strengthened understanding of model compression techniques and their deployability in real-world applications.

## VI. CONCLUSION

This report explored knowledge distillation as a strategy to compress deep convolutional neural networks by transferring knowledge from a larger teacher model (ResNet34) to a smaller student model (ResNet18). Through empirical evaluation, it was demonstrated that the student model, when trained using response-based knowledge distillation, achieved 89.0% accuracy while using ~50% fewer resources in terms of both memory and computation.

The findings confirm that KD is an effective model compression method, especially valuable for scenarios requiring deployment on resource-constrained devices. Despite training the student from scratch, its performance was significantly boosted through distilled soft targets, validating the theoretical advantages of KD in practical settings.

### A. *Future Directions*

- Feature and relation-based distillation: Explore deeper forms of knowledge transfer and their results.
- Self-distillation and online KD: Implementing these alternative schemes can potentially reduce training time and provide additional flexibility.
- Broader datasets and tasks: Validating across more complex, real-world datasets (e.g., CIFAR-100) to test the limitations of KD in a shallow model.
- Different types of teachers: Study the effects of KD using specialist models.

### B. *Final Remarks*

Overall, this work contributes to the growing field of efficient deep learning applications by demonstrating how knowledge distillation allows for a bypass of the trade-off between performance and efficiency. It underscores the feasibility of deploying compact models in real-world systems, bridging the gap between research-grade networks and deployable AI solutions.

## REFERENCES

[1] G. Hinton, O. Vinyals, and J. Dean, “Distilling the Knowledge in a Neural Network,” *arXiv:1503.02531 [cs, stat]*, Mar. 2015, Available: https://arxiv.org/abs/1503.02531

[2] Guobin Chen, Wongun Choi, Xiang Yu, Tony X Han, and Manmohan Chang. Learning efficient object detection models with knowledge distillation. Advances in Neural Information Processing Systems, 30, 2017.

[3] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, “FitNets: Hints for Thin Deep Nets,” arXiv:1412.6550 [cs], Mar. 2015, Available: https://arxiv.org/abs/1412.6550

[4] H. Su, Z. Jian, and S. Yu, “Task Integration Distillation for Object Detectors,” arXiv preprint arXiv:2404.01699, 2024. Available: https://arxiv.org/abs/2404.01699

[5] J. Yim, D. Joo, J. Bae and J. Kim, "A Gift from Knowledge Distillation: Fast Optimization, Network Minimization and Transfer Learning," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 7130-7138, doi: 10.1109/CVPR.2017.754.

[6] J. Gou, B. Yu, S. J. Maybank, and D. Tao, “Knowledge distillation: A survey,” arXiv preprint arXiv:2006.05525, 2020. Available: https://arxiv.org/abs/2006.05525

[7] “2.3. ResNet and ResNet_vd series — PaddleClas documentation,” Readthedocs.io,2015.https://paddleclas.readthedocs.io/en/latest/models/ResNet_and_vd_en.html