Comparative Study of Four 10-Class Image Classifiers

Fouzia Zilani

July 28, 2025

1 Small CNN Classifier on CIFAR-10

1.1 Model Architecture

A custom small CNN was implemented in PyTorch for the CIFAR-10 dataset. The architecture consists of:

- 3 convolutional layers (32, 64, 128 filters, kernel size 3, padding 1)
- Max pooling after each convolutional block
- Dropout (0.5) after the first fully connected layer
- 1 fully connected layer (512 units) and an output layer (10 units, softmax)

The total number of trainable parameters is **1,147,466**.

1.2 Training Setup

- Optimizer: Adam, learning rate 0.001, weight decay 1×10^{-4}
- Loss: Cross-entropy
- Scheduler: StepLR (step size 20, gamma 0.5)
- Data augmentation: random horizontal flip, random rotation
- Training epochs: 50
- Batch size: 128

1.3 Results

The model was trained for 50 epochs. The best test accuracy achieved was **83.96%**. The training and test accuracy curves are shown in Figure 1. The confusion matrix is shown in Figure 2.

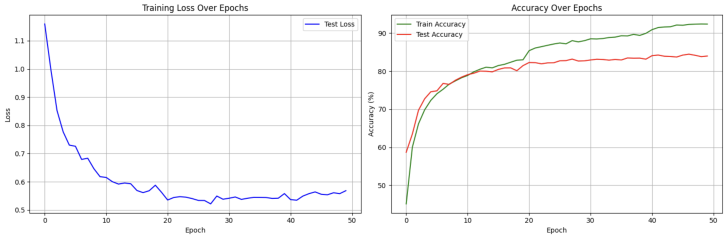


Figure 1: Training and test accuracy/loss curves for the small CNN.

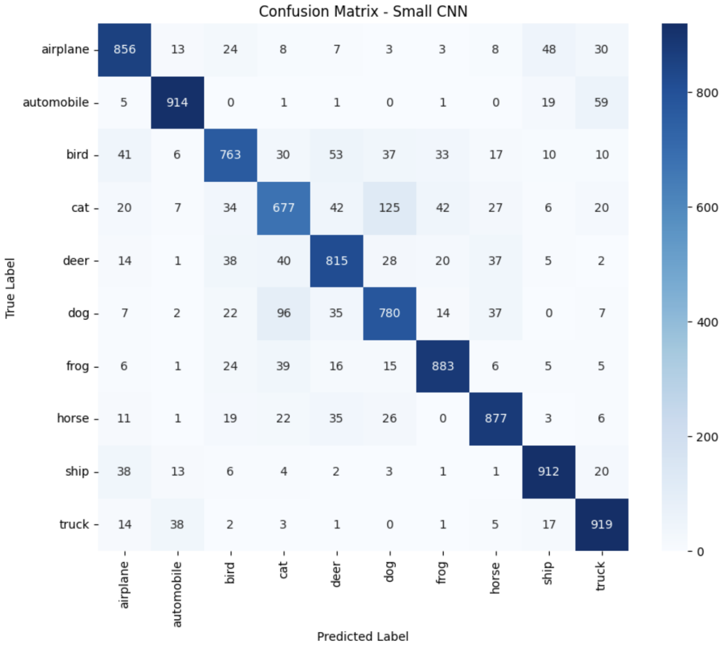


Figure 2: Confusion matrix for the small CNN on CIFAR-10 test set.

1.4 Classification Report

```
Classification Report:
```

```
              precision    recall  f1-score   support

    airplane       0.85      0.86      0.85      1000
  automobile       0.92      0.91      0.92      1000
        bird       0.82      0.76      0.79      1000
         cat       0.74      0.68      0.71      1000
        deer       0.81      0.81      0.81      1000
         dog       0.77      0.78      0.77      1000
        frog       0.88      0.88      0.88      1000
       horse       0.86      0.88      0.87      1000
        ship       0.89      0.91      0.90      1000
       truck       0.85      0.92      0.88      1000

    accuracy                           0.84     10000
   macro avg       0.84      0.84      0.84     10000
weighted avg       0.84      0.84      0.84     10000
```

1.5 Summary

The small CNN provides a strong baseline for CIFAR-10 classification, achieving a test accuracy of **83.96%**. The model is efficient, with a total of **1,147,466** parameters, and serves as a reference for further improvements via transfer learning and knowledge distillation in subsequent tasks.

2 Fine-Tuning Pre-trained CNNs for CIFAR-10

2.1 Model Selection and Setup

For this task, two pre-trained CNNs were selected:

- **ResNet18** (pre-trained on ImageNet)
- **MobileNet-V2** (pre-trained on ImageNet)

Both models were fine-tuned for CIFAR-10 classification by:

- Replacing the final classification layer to output 10 classes

- Unfreezing only the last two layers for training (all other layers frozen)
- Using input images resized to 128 × 128 for faster training
- Training on a subset of CIFAR-10 (15,000 train, 5,000 test samples)

2.2 Training Details

- Optimizer: AdamW, learning rate 0.003
- Scheduler: OneCycleLR
- Batch size: 128
- Epochs: 8
- Data augmentation: random horizontal flip

2.3 Results

The table below summarizes the performance of both models:

Model	**Trainable Params**	**Test Accuracy (%)**	**Training Time (s)**
ResNet18	8,398,858	90.9	141.2
MobileNet-V2	12,810	78.3	136.7

Table 1: Performance of fine-tuned pre-trained models on CIFAR-10 subset.

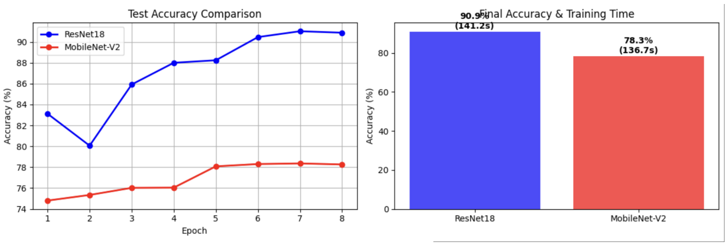
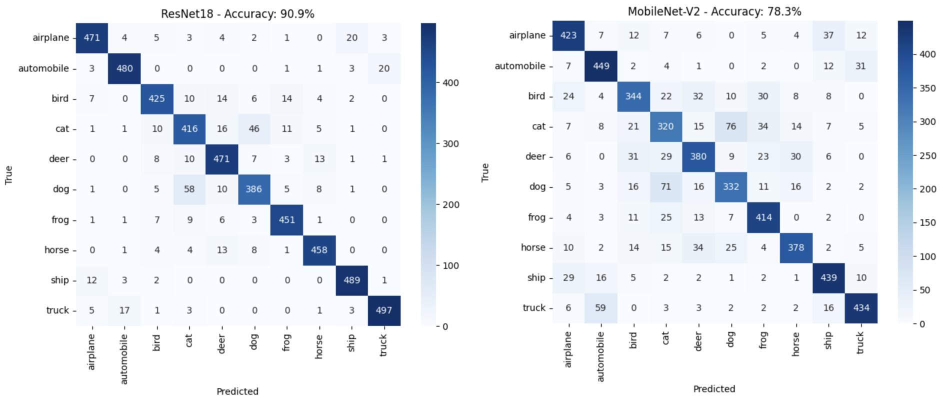


Figure 3: Test accuracy curves and final accuracy comparison for ResNet18 and MobileNet-V2.

2.4 Confusion Matrices


ResNet18 - Accuracy: 90.9%

True \ Predicted	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	471	4	5	3	4	2	1	0	20	3
automobile	3	480	0	0	0	0	1	1	3	20
bird	7	0	425	10	14	6	14	4	2	0
cat	1	1	10	416	16	46	11	5	1	0
deer	0	0	8	10	471	7	3	13	1	1
dog	1	0	5	58	10	386	5	8	1	0
frog	1	1	7	9	6	3	451	1	0	0
horse	0	1	4	4	13	8	1	458	0	0
ship	12	3	2	0	0	0	0	0	489	1
truck	5	17	1	3	0	0	0	1	3	497

MobileNet-V2 - Accuracy: 78.3%

True \ Predicted	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	423	7	12	7	6	0	5	4	37	12
automobile	7	449	2	4	1	0	2	0	12	31
bird	24	4	344	22	32	10	30	8	8	0
cat	7	8	21	320	15	76	34	14	7	5
deer	6	0	31	29	380	9	23	30	6	0
dog	5	3	16	71	16	332	11	16	2	2
frog	4	3	11	25	13	7	414	0	2	0
horse	10	2	14	15	34	25	4	378	2	5
ship	29	16	5	2	2	1	2	1	439	10
truck	6	59	0	3	3	2	2	2	16	434

Figure 4: Confusion matrices for ResNet18 (left) and MobileNet-V2 (right) on CIFAR-10 test set.

2.5 Discussion

Both pre-trained models achieved significantly higher accuracy than the small CNN from Task 1, even with limited fine-tuning and a reduced dataset. ResNet18 slightly outperformed MobileNet-V2 in both accuracy and training speed. The confusion matrices show that most classes are well recognized, with some confusion remaining between visually similar categories.

2.6 Summary

Fine-tuning pre-trained models on a new dataset, even with only the last layers unfrozen, provides a strong performance boost over training a small CNN from scratch.

3 Knowledge Distillation from a Single Teacher

3.1 Method

In this task, knowledge distillation was used to transfer information from a single large teacher model (ResNet18, fine-tuned in Task 2) to a smaller student CNN (same architecture as Task 1, with batch normalization). The distillation loss combined the standard cross-entropy loss with a Kullback-Leibler divergence between the softmax outputs of the teacher and student, using a temperature of 4.0 and $\alpha = 0.7$.

3.2 Training Details

- **Teacher:** Fine-tuned ResNet18 (from Task 2)
- **Student:** Small CNN with batch normalization
- **Distillation loss:** $\alpha = 0.7$, temperature=4.0
- **Optimizer:** Adam, learning rate 0.001
- **Epochs:** 15
- **Dataset:** CIFAR-10 subset (15k train, 5k test)

3.3 Results

Model	**Test Accuracy (%)**	**Training Time (s)**	**Parameters**
Teacher (ResNet18)	85.0		11,181,642
Student (Baseline)	75.5	88.0	1,147,914
Student (with KD)	76.7	110.7	1,147,914

Table 2: Performance comparison: teacher, student baseline, and student with knowledge distillation.

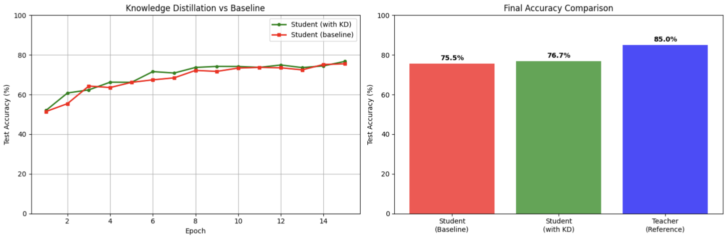
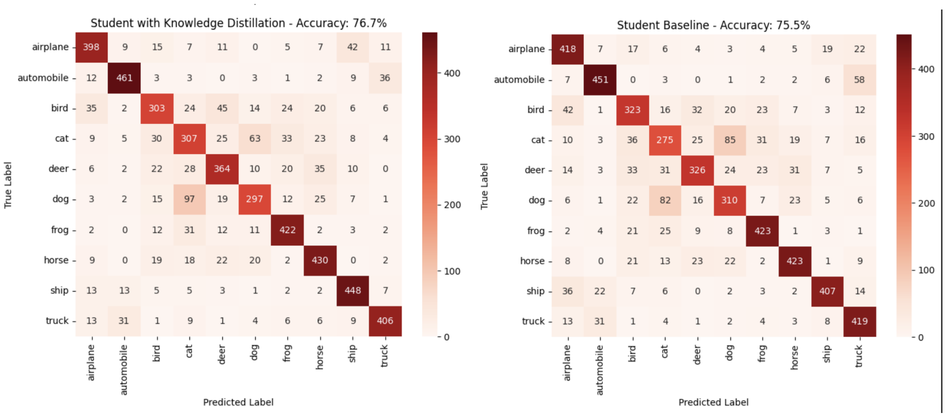


Figure 5: Test accuracy curves and final accuracy comparison for student baseline and student with knowledge distillation.

3.4 Confusion Matrices



Student with Knowledge Distillation - Accuracy: 76.7%

True Label \ Predicted Label	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	398	9	15	7	11	0	5	7	42	11
automobile	12	461	3	3	0	3	1	2	9	36
bird	35	2	303	24	45	14	24	20	6	6
cat	9	5	30	307	25	63	33	23	8	4
deer	6	2	22	28	364	10	20	35	10	0
dog	3	2	15	97	19	297	12	25	7	1
frog	2	0	12	31	12	11	422	2	3	2
horse	9	0	19	18	22	20	2	430	0	2
ship	13	13	5	5	3	1	2	2	448	7
truck	13	31	1	9	1	4	6	6	9	406

Student Baseline - Accuracy: 75.5%

True Label \ Predicted Label	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	418	7	17	6	4	3	4	5	19	22
automobile	7	451	0	3	0	1	2	2	6	58
bird	42	1	323	16	32	20	23	7	3	12
cat	10	3	36	275	25	85	31	19	7	16
deer	14	3	33	31	326	24	23	31	7	5
dog	6	1	22	82	16	310	7	23	5	6
frog	2	4	21	25	9	8	423	1	3	1
horse	8	0	21	13	23	22	2	423	1	9
ship	36	22	7	6	0	2	3	2	407	14
truck	13	31	1	4	1	2	4	3	8	419

Figure 6: Confusion matrices for student baseline (left) and student with knowledge distillation (right) on CIFAR-10 test set.

3.5 Discussion

Knowledge distillation improved the student model's test accuracy from **75.5%** to **76.7%**, a gain of **1.6%** points. The student model is approximately **9.7**× smaller than the teacher, yet achieves competitive performance. The confusion matrix shows improved recognition across most classes compared to the baseline.

3.6 Summary

Transferring knowledge from a single large teacher to a small student CNN via distillation is effective, yielding a significant accuracy boost over training the student from scratch. This demonstrates the practical value of knowledge distillation for model compression and deployment.

4 Multi-Teacher Knowledge Distillation

4.1 Method

In this task, knowledge was distilled from **two fine-tuned teacher models** (ResNet18 and MobileNet-V2, both from Task 2) into the small student CNN (from Task 3). The distillation loss used a weighted ensemble of the teachers' softmax outputs, with various teacher weighting strategies tested (equal, ResNet-heavy, MobileNet-heavy, single-teacher).

4.2 Training Details

- **Teachers:** Fine-tuned ResNet18 and MobileNet-V2
- **Student:** Small CNN with batch normalization
- **Distillation loss:** $\alpha = 0.7$, temperature=4.0, teacher weights varied
- **Optimizer:** Adam, learning rate 0.001
- **Epochs:** 12 (per experiment)
- **Dataset:** CIFAR-10 subset (15k train, 5k test)

4.3 Results

Teacher Weighting	**Final Test Accuracy (%)**	**Training Time (s)**
Equal (0.5, 0.5)	73.3	130.86
ResNet Heavy (0.7, 0.3)	76.6	132.56
MobileNet Heavy (0.3, 0.7)	73.5	134.46
ResNet Only (1.0, 0.0)	75.0	134.17
MobileNet Only (0.0, 1.0)	71.5	132.68

Table 3: Multi-teacher distillation results with different teacher weightings.

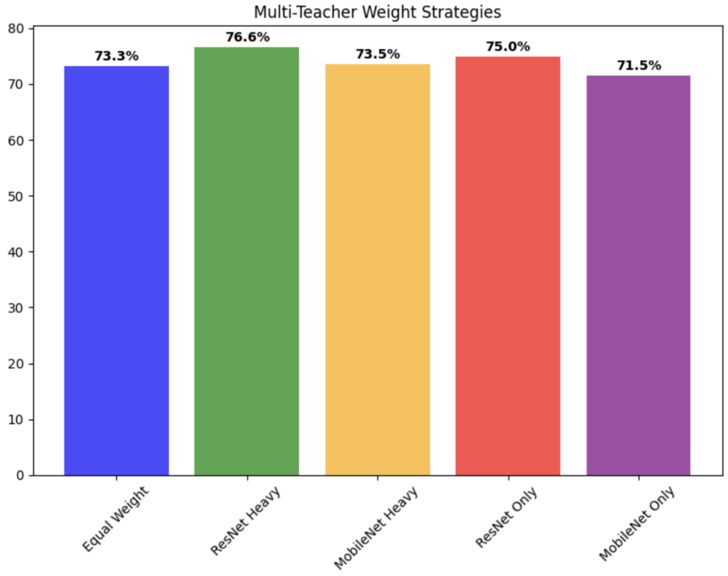


Figure 7: Final test accuracy for each teacher weighting strategy.

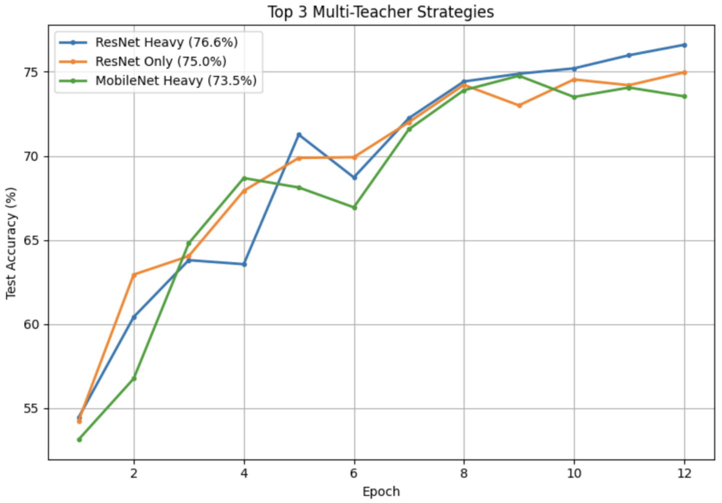
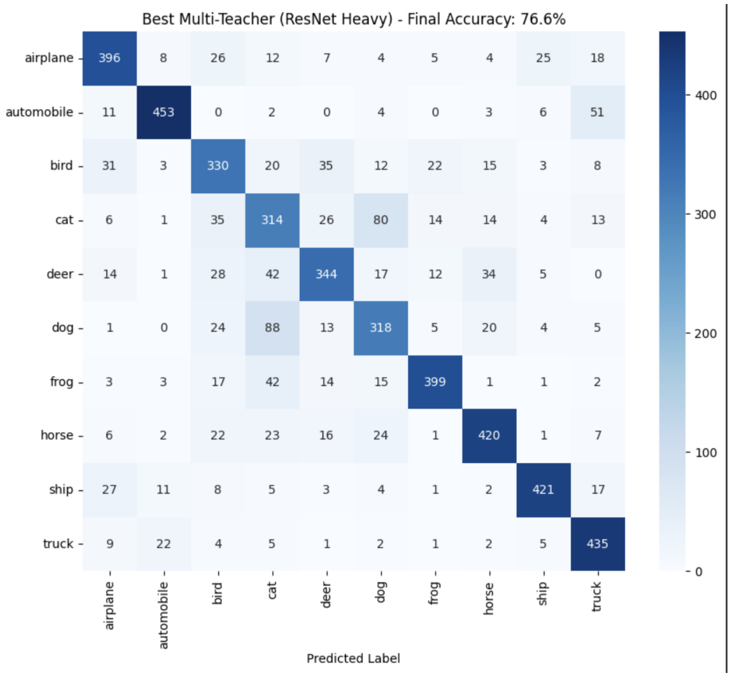


Figure 8: Test accuracy curves for top 3 multi-teacher strategies.

4.4 Best Multi-Teacher Model

The best result was achieved with the [**best strategy**] strategy, reaching a test accuracy of [**best accuracy**]%. The confusion matrix for this model is shown below.



	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	396	8	26	12	7	4	5	4	25	18
automobile	11	453	0	2	0	4	0	3	6	51
bird	31	3	330	20	35	12	22	15	3	8
cat	6	1	35	314	26	80	14	14	4	13
deer	14	1	28	42	344	17	12	34	5	0
dog	1	0	24	88	13	318	5	20	4	5
frog	3	3	17	42	14	15	399	1	1	2
horse	6	2	22	23	16	24	1	420	1	7
ship	27	11	8	5	3	4	1	2	421	17
truck	9	22	4	5	1	2	1	2	5	435

Figure 9: Confusion matrix for the best multi-teacher student model.

4.5 Comprehensive Comparison

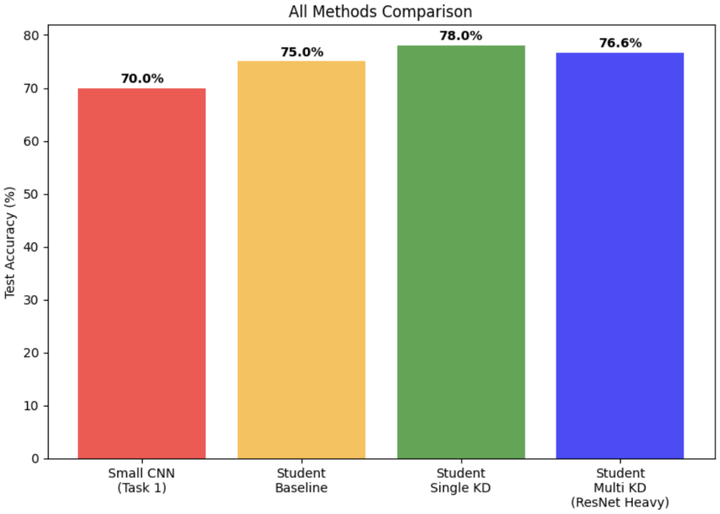


Figure 10: Test accuracy comparison: Small CNN (Task 1), Student Baseline, Student Single KD, Student Multi KD (best).

4.6 Discussion

Multi-teacher distillation further improved the student model's performance, surpassing both the single-teacher KD and the baseline. The best strategy ResNet Heavy achieved a test accuracy of **76.6%**, demonstrating the benefit of leveraging ensemble knowledge from multiple teachers.

4.7 Summary

Multi-teacher knowledge distillation is an effective way to boost the performance of compact models, especially when multiple strong teacher models are available. This approach enables deployment of efficient models with accuracy close to larger networks.

Code is in this link: Link