Tree-like Decision Distillation
– *Supplementary Material* –

Jie Song[1,*], Haofei Zhang[1,*], Xinchao Wang[2,4], Mengqi Xue[1], Ying Chen[1], Li Sun[1,†], Dacheng Tao[3], and Mingli Song[1]
[1]Zhejiang University, [2]National University of Singapore,
[3]The University of Sydney, [4]Stevens Institute of Technology
{sjie,haofeizhang,mqxue,lynesychen,lsun,brooksong}@zju.edu.cn,
xinchao@nus.edu.sg, dacheng.tao@sydney.edu.au

We provide in this document additional details and results that cannot fit into the main manuscript due to the page limit.

1. Updating Rules

Here we introduce the updating rules of the within- and between-class scatter matrices when two sibling nodes are merged into their parent node. Let u and v be two sibling nodes of a common parent node w. If u and v are merged into node w, then the average feature vector of all the categories associated with node w is computed by:

$$\overline{\boldsymbol{z}^{(w)}} = \frac{1}{|\mathcal{Y}_w| \cdot K} \sum\nolimits_{y \in \mathcal{Y}_w} \sum\nolimits_{k=1}^{K} \boldsymbol{z}_k^y, \tag{1}$$

where categories associated with node w is the union of categories from u and v, *i.e.*, $\mathcal{Y}_w = \mathcal{Y}_u \cup \mathcal{Y}_v$. With $\overline{\boldsymbol{z}^{(w)}}$, the between-class scatter matrix is computed as follows:

$$\boldsymbol{S}_B^w = \frac{1}{|\mathcal{Y}_v|} \sum\nolimits_{y \in \mathcal{Y}_w} (\overline{\boldsymbol{z}^y} - \overline{\boldsymbol{z}^{(w)}})(\overline{\boldsymbol{z}^y} - \overline{\boldsymbol{z}^{(w)}})^T. \tag{2}$$

The within-class scatter matrix is updated by directly combining the within-class scatter matrices of node u and v:

$$\boldsymbol{S}_W^w = \frac{|\mathcal{Y}_u|}{|\mathcal{Y}_u| + |\mathcal{Y}_v|} \boldsymbol{S}_W^u + \frac{|\mathcal{Y}_u|}{|\mathcal{Y}_u| + |\mathcal{Y}_v|} \boldsymbol{S}_W^v. \tag{3}$$

Note that the updated scatter matrices should be properly normalized as the determinant is very sensitive to the scaling factor of matrices.

2. Model Architectures

For most experimental settings, we follow the setup adopted in [13]. Here we list the architecture details for better clarification.

- **Wide Residual Network (WRN)** [16]. "WRN-40-2" represnets wide ResNet with depth 40 and width factor 2.
- **ResNet** [2]. ResNet56 and ResNet20 are *cifar*-style ResNets with 3 groups of basic blocks, each with 16, 32, and 64 channels respectively. ResNet50 follows the *ImageNet*-style ResNet with Bottleneck blocks and more channels.
- **MobileNetV2** [11]. We set the width multiplier to be 0.5 in the experiments.
- **VGG** [12]. The VGG used in our experiments are adapted from its original ImageNet counterpart.
- **ShuffleNetV1** [18]. ShuffleNets are proposed for efficient training and we adapt them to input of size 32x32.

3. Additional Experimental Results

3.1. Top-5 accuracy results

As top-5 accuracy is a widely-used evaluation metric on ImageNet, here we make comparisons between the proposed TDD and other competitors on tiny-ImageNet using top-5 accuracy to give a more comprehensive view of the proposed method. Experiments are conducted under the homogeneous distillation settings (ResNet56 $\rightarrow$ ResNet20, WRN_40_2 $\rightarrow$ WRN_16_2, WRN_40_2 $\rightarrow$ WRN_40_1) and the heterogeneous distillation settings (ResNet50 $\rightarrow$ MobileNetV2, WRN_40_2 $\rightarrow$ ShuffleNetV1, ResNet50 $\rightarrow$ VGG8). The experimental results are provided in Table 1. Based on these results, we make following three main observations.

- Similar to the results of top-1 accuracy, most prior distillation methods yield inferior performance to the vanilla KD in the metric of top-5 accuracy. It again

[*]Equal contribution
[†]Corresponding author

Table 1. Top-5 accuracy of the homogeneous and the heterogeneous distillation on tiny-ImageNet (in %). Experiments are repeated for five times and the average results are provided.

	Homogeneous Distillation			**Heterogeneous Distillation**		
Teacher **Student**	ResNet56 ResNet20	WRN_40_2 WRN_16_2	WRN_40_2 WRN_40_1	ResNet50 MobileNetV2	WRN_40_2 ShuffleNetV1	ResNet50 VGG8
Teacher	81.79	83.45	83.45	86.82	83.45	86.82
Student	78.11	81.66	80.58	81.79	82.35	79.33
KD [4]	79.42	82.80	82.31	82.05	85.69	82.43
Fitnets [10]	78.02	81.65	N/A	80.81	N/A	79.66
AT [17]	79.23	82.70	81.46	76.83	85.06	77.71
FSP [15]	77.59	81.26	N/A	N/A	N/A	N/A
FT [5]	79.28	82.40	80.95	82.04	83.73	80.83
PKT [8]	78.79	82.26	81.68	**82.50**	84.63	80.73
SPKD [14]	79.38	80.78	79.37	81.61	85.71	81.11
VID [1]	78.93	82.22	81.16	81.38	83.96	73.30
CC [9]	78.08	81.34	80.39	81.46	82.73	79.03
RKD [7]	78.45	81.27	81.02	82.10	83.79	79.47
TDD (Ours)	**79.56**±0.04	**82.92**±0.09	**82.57**±0.11	82.20±0.08	**85.96**±0.15	**82.82**±0.12

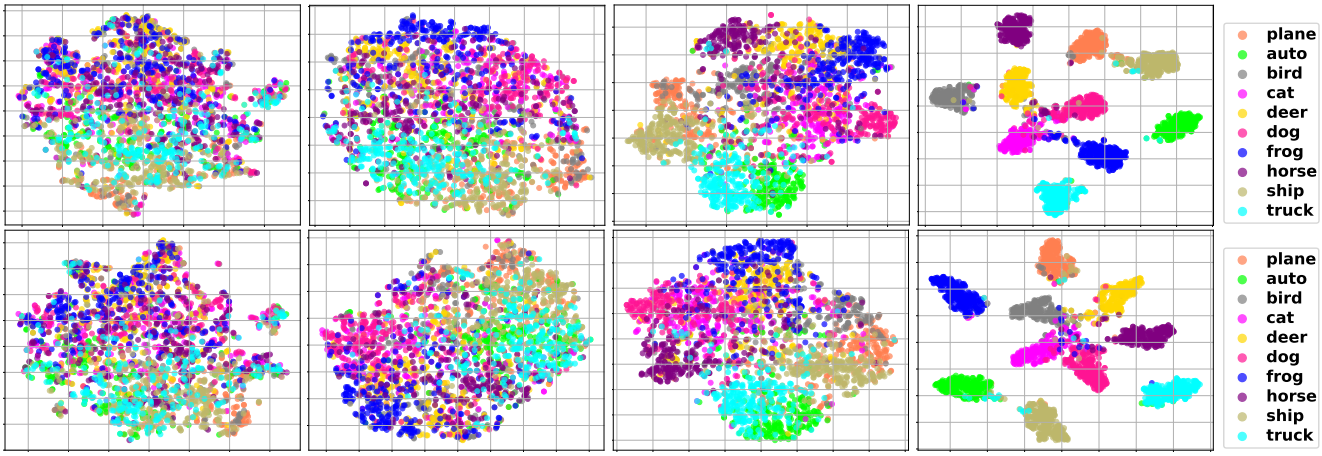


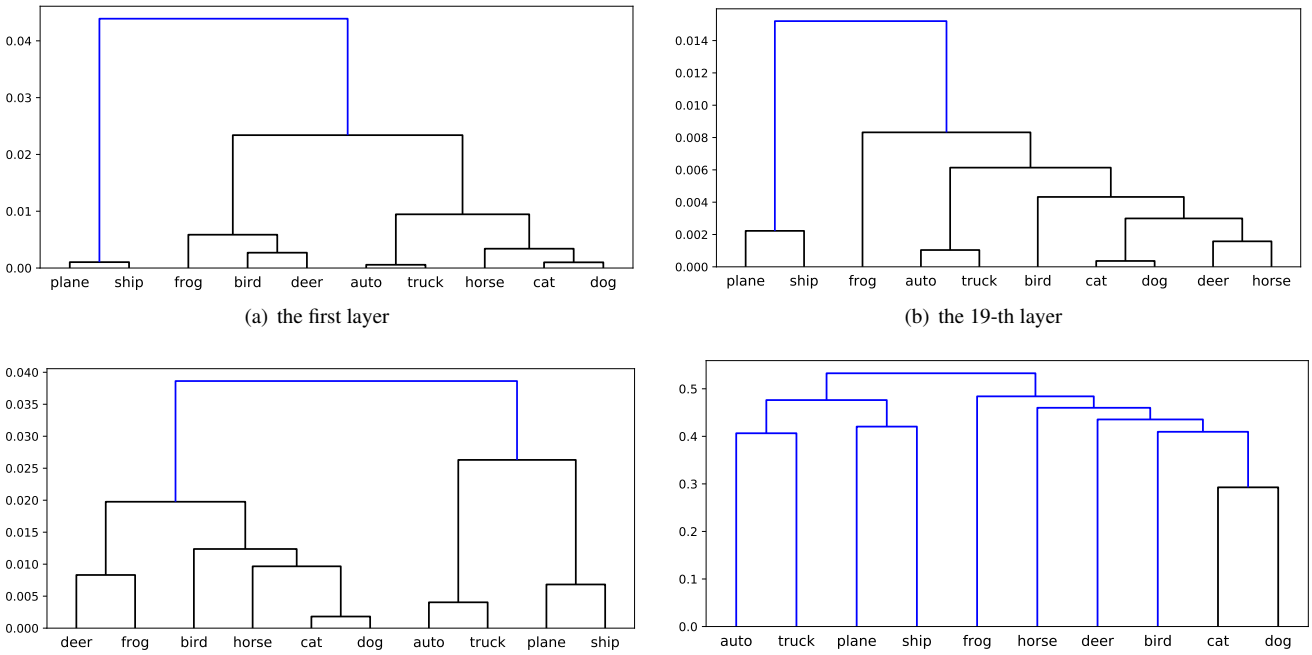
Figure 1. Visualization of feature distributions after linear discriminant analysis using t-SNE [3]. **The First Row**: the 1-st, the 19-th, the 37-th and the 55-th layers (from left to right) in ResNet56. **The Second Row**: the 1-st, the 7-th, the 13-th and the 19-th layers (from left to right) in ResNet20.

proves that the vanilla KD is a strong baseline in the field of knowledge distillation.

- Surprisingly, we can see that under the top-5 accuracy, many prior methods can not match the trivial student without any distillation, in both the homogeneous distillation settings and the heterogeneous distillation settings.
- The propose TDD consistently outperforms the vanilla KD and other competitors, which demonstrates the superiority of the proposed method to state-of-the-arts.

3.2. Visualization of Feature Distributions

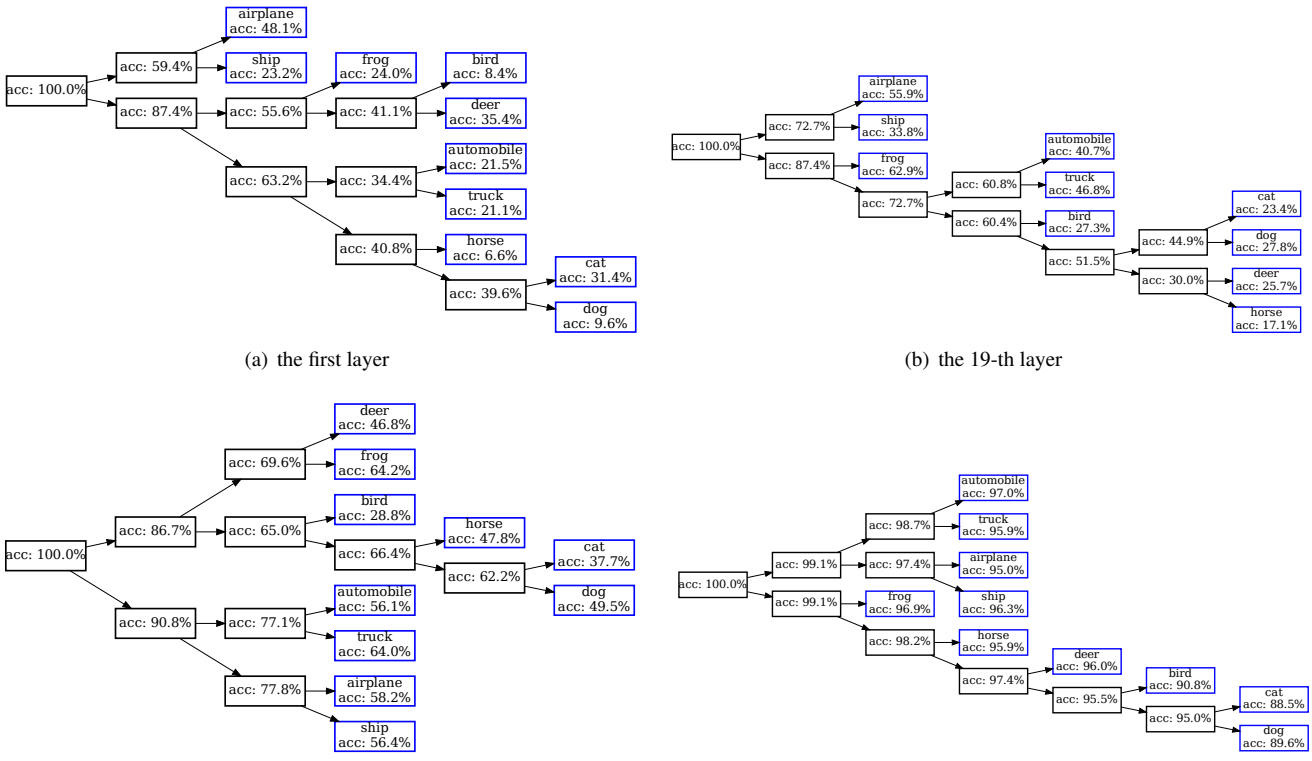
Here we plot the feature distributions in different layers of both the teacher (ResNet56) and the student (ResNet20) models. Note that here the student model is trivially trained without any distillation methods. Results are shown in Figure 1. It can be seen that although the student mode is trained without any knowledge distilled from the teacher,



(a) the first layer (b) the 19-th layer

(c) the 37-th layer (d) the last layer

Figure 2. The dendrograms of category relationships produced from different layers from ResNet56 on CIFAR-10.



(a) the first layer (b) the 19-th layer

(c) the 37-th layer (d) the last layer

Figure 3. The decision trees produced from different layers from ResNet56 on CIFAR-10. "Acc" in each node denotes the accuracy via nearest neighbor search in the feature subspace after linear discriminant analysis.

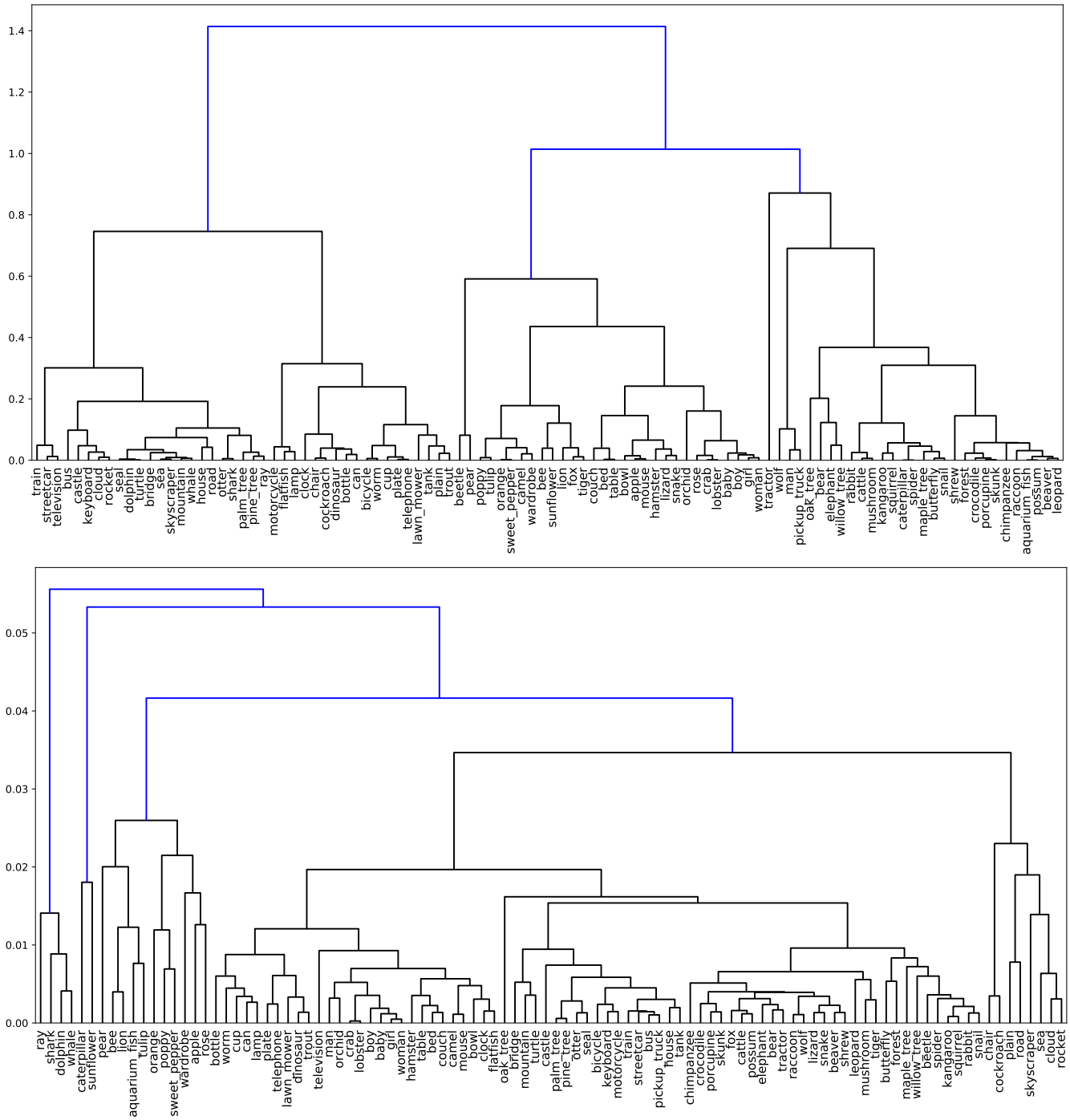


Figure 4. Visualization of the dendrograms produced from the 1-st (top) and the 13-th (bottom) layers in WRN_40_2.

the feature distributions from them share several similarities.

- In both the teacher and the student models, the features are becoming increasingly distinguishable from the first layer to the last layer. In early layers, the features from different categories are mixed together in early layers. However, when it reaches the last layer, the features from different categories cluster into different regions, which makes the feature space ready for final classification.

- Although the early layers exhibit poor discriminant ability for precise classification, they show great po-

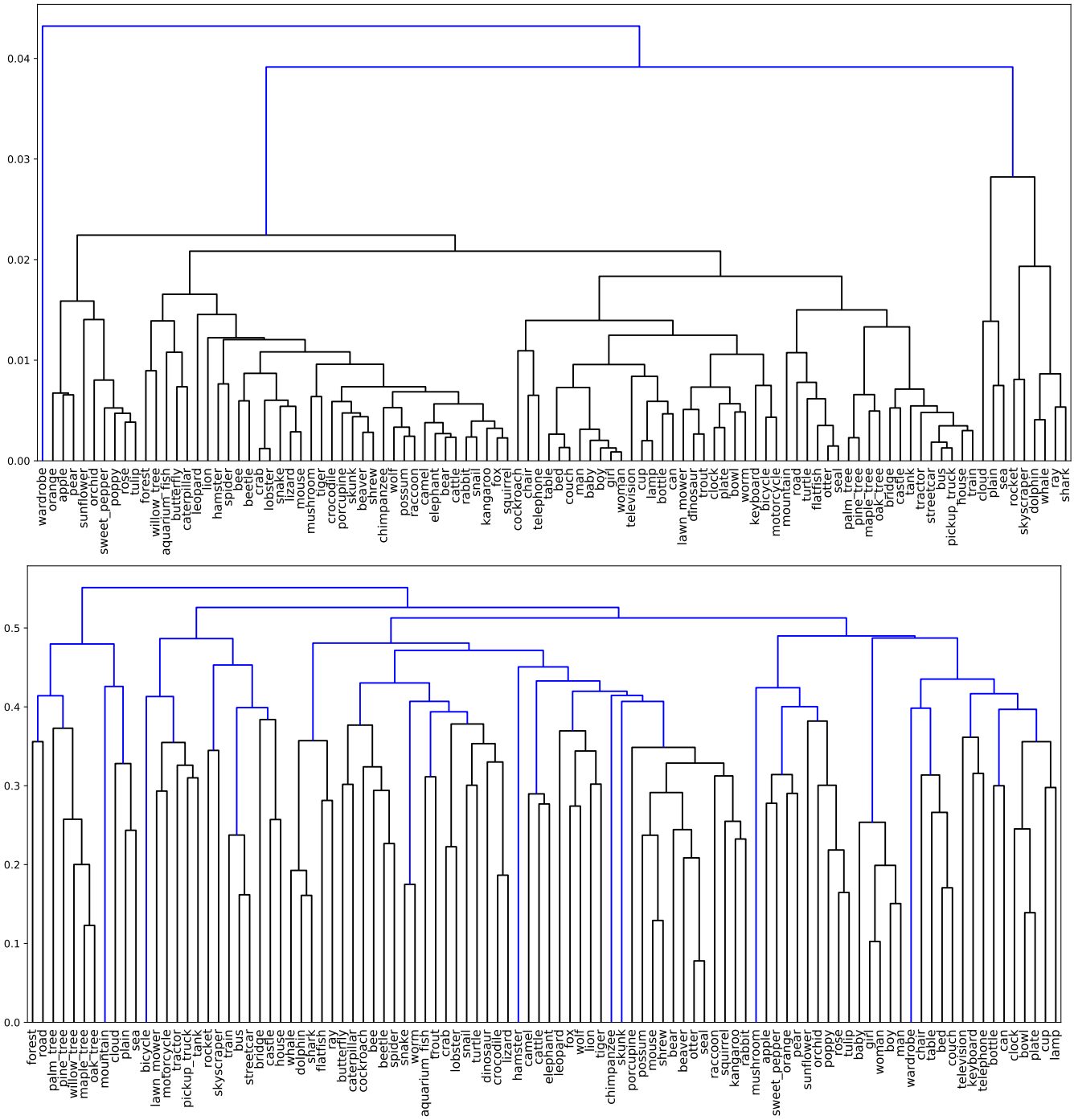


Figure 5. Visualization of the dendrograms produced from the 25-st (top) and the 37-th (bottom) layers in WRN_40_2.

tential to make some coarse-grained recognition. For example, we can find that even in the first layer, the features from *automobile*, *truck*, *plane* and *ship*, which are under the umbrella of *vehicles*, tend to cluster together. Features from the other categories that are under the umbrella of *animals* cluster into another region. Vehicles and animals can be easily differentiated in these early layers.

3.3. Visualization of the Decision Process

Here we provide more visualization results for a better understanding of the decision process in deep neural networks. The dendrograms and the decision trees from different layers of ResNet56 are provided in Figure 2 and Fig-

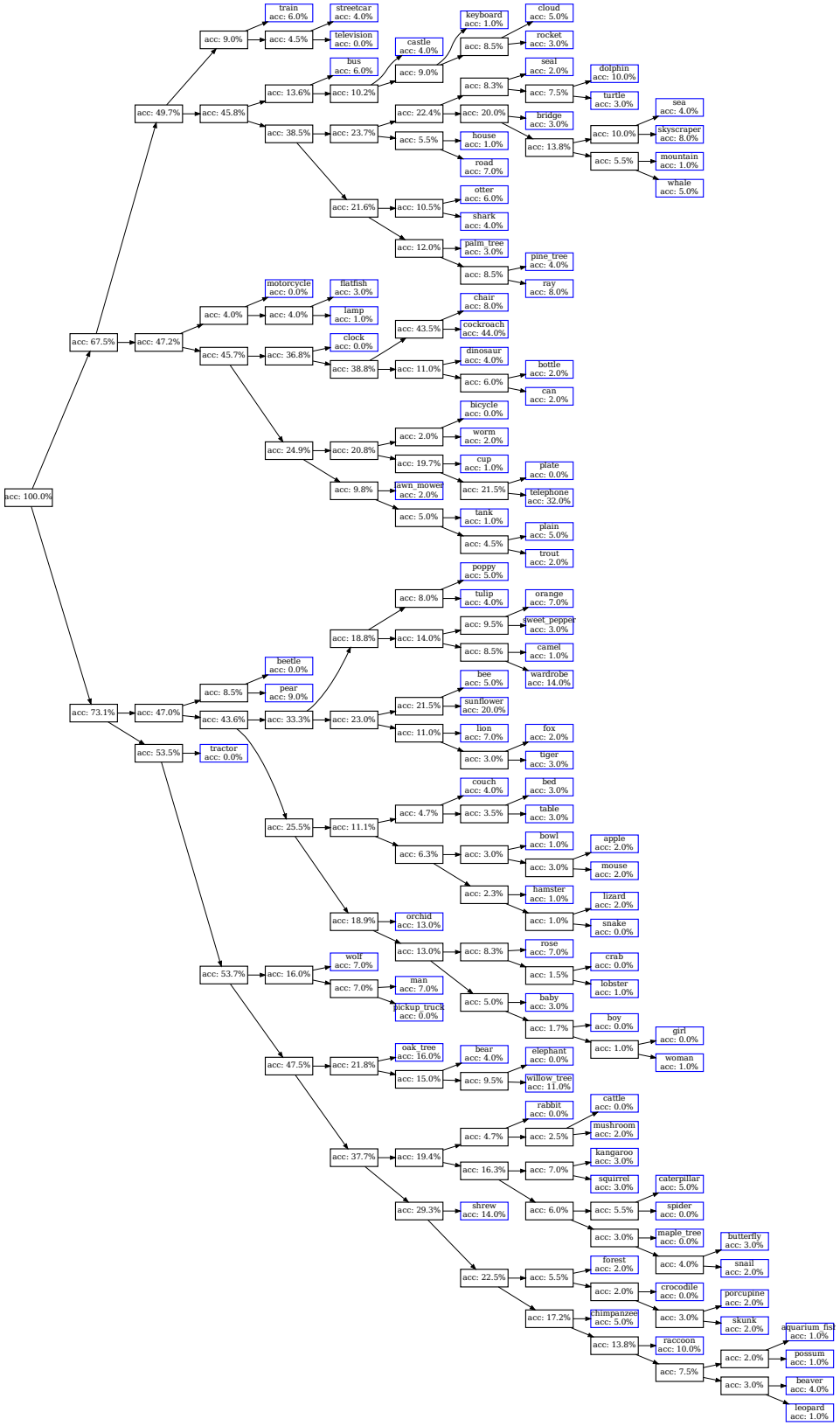


Figure 6. The decision tree produced after the first layer in WRN_40_2 on CIFAR-100. The leaf nodes are painted with blue borders. "Acc" denotes the accuracy via nearest neighbor search in the feature subspace after linear discriminant analysis.

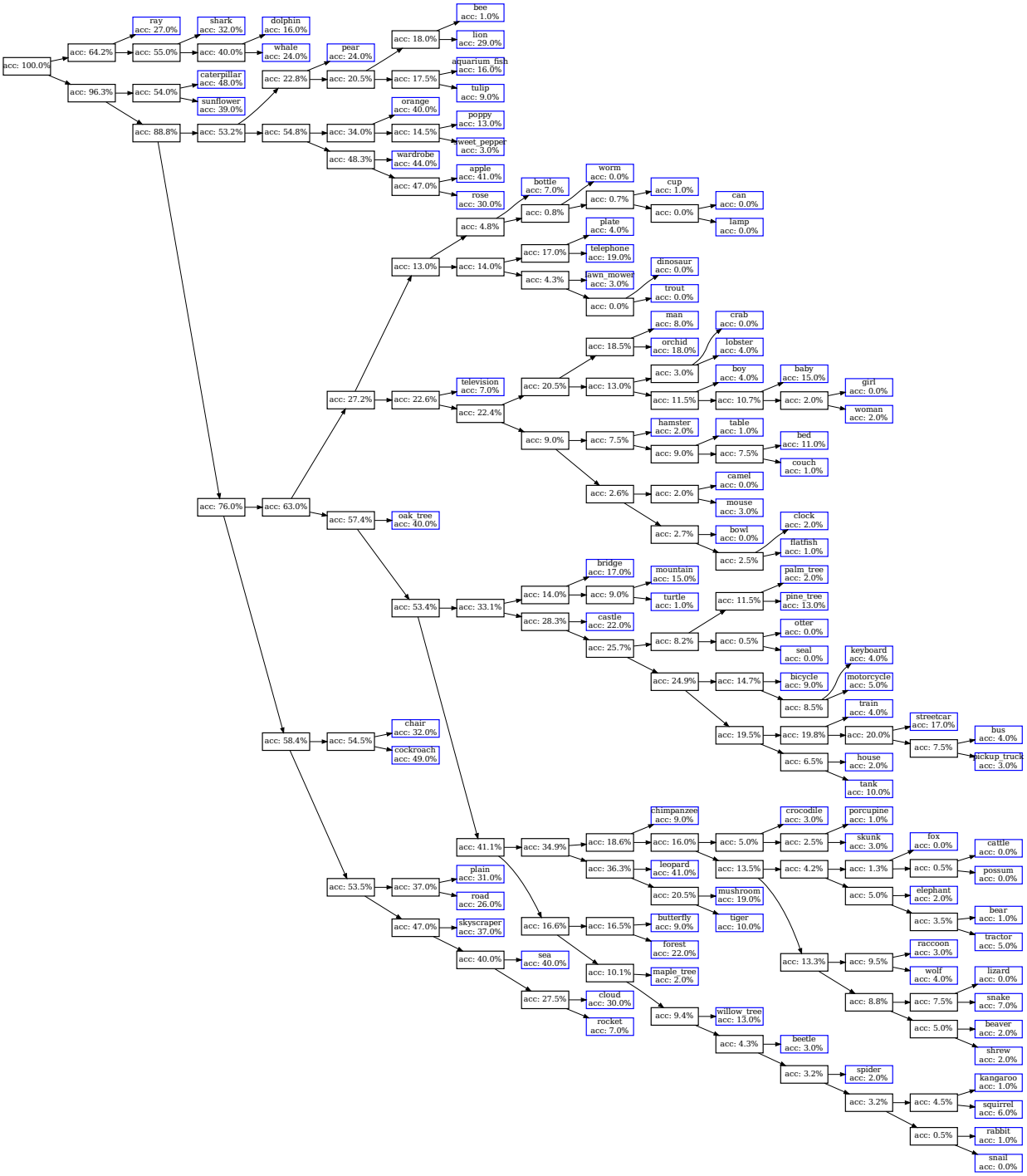


Figure 7. The decision tree produced after the 13-th layer in WRN_40_2 on CIFAR-100. The leaf nodes are painted with blue borders. "Acc" denotes the accuracy via nearest neighbor search in the feature subspace after linear discriminant analysis.

ure 3, respectively. We can see that the dendrograms and the decision trees change from layer to layer, which implies that different layers are actually making different decisions. However, they will not change dramatically between two successive layers as ReLU [6] is actually a sublinear function. Another observation is that within each layer, the decision process from the root node to the leaves in the decision tree is conducted in a coarse-to-fine manner, with high-to-

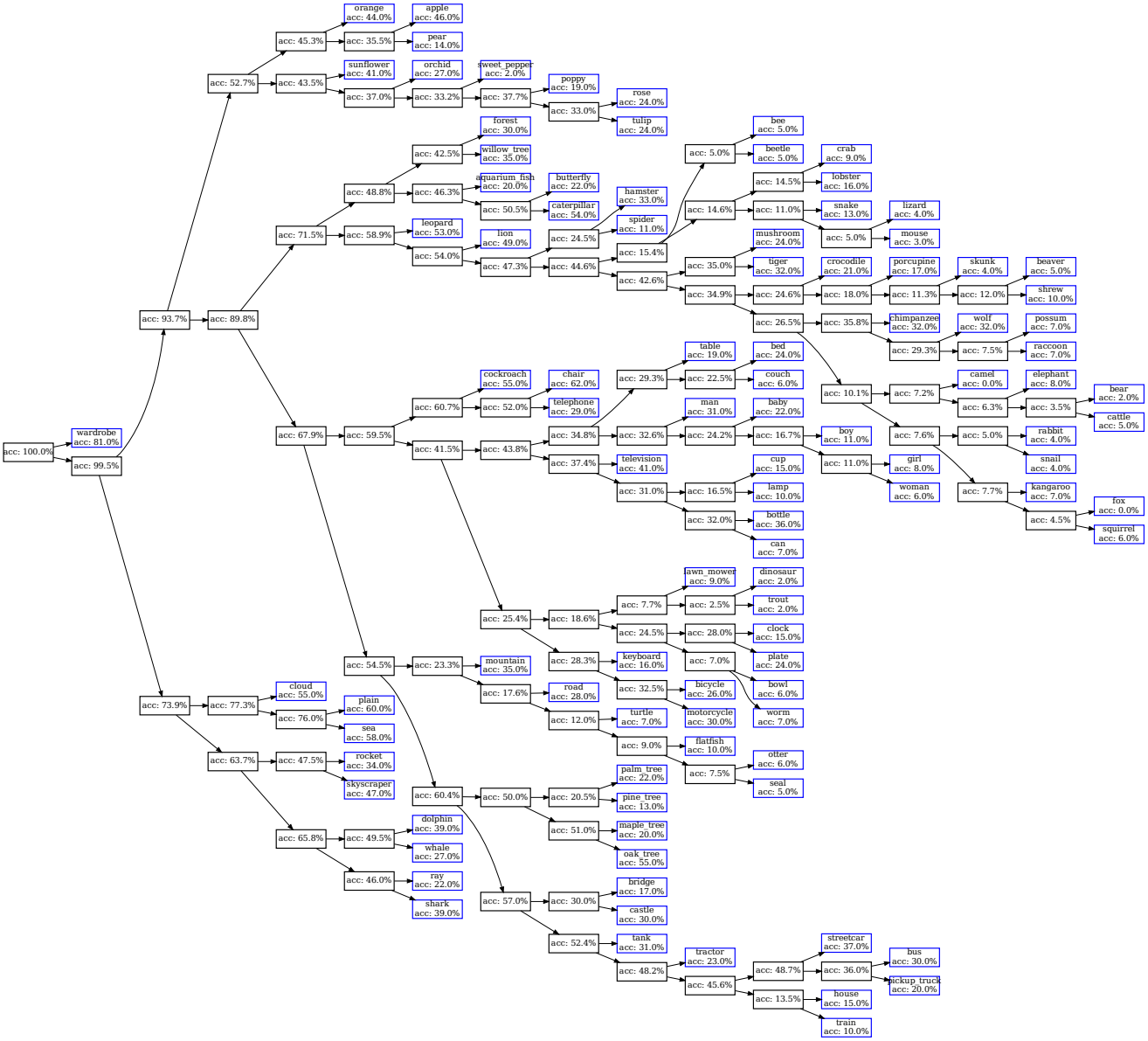


Figure 8. The decision tree produced after the 25-th layer in WRN_40_2 on CIFAR-100. The leaf nodes are painted with blue borders. "Acc" denotes the accuracy via nearest neighbor search in the feature subspace after linear discriminant analysis.

low accuracy. The higher accuracy of coarse-grained classification is attributed to two main factors. (1) The coarse-grained classification has fewer category options, and thus the baseline accuracy of random guess is higher. For example, a random 2-way classification attains 50% accuracy, while a random 10-way classification reaches only 10%. (2) The misclassification usually takes places within the coarse-grained superclass, which is the property we exploit in the proposed distillation method. To demonstrate the contribution of the second factor better, we compute the baseline accuracy of random guess as follows. Here we take the decision tree from the first layer of ResNet56 as an example, as shown in Figure 3 (a). The average classification accuracy ACC_{avg} for all the categories in the leaf nodes is 22.9%. Based on this, for the 2-way coarse-grained classification, *i.e.*, $\mathcal{S}_1 = \{$*airplane, ship*$\}$ versus $\mathcal{S}_2 = \{$*frog, bird, deer, automobile, truck, horse, cat, dog*$\}$, the classificaiton accuracy by the random guess for data in $\mathcal{S}_1$ is

$$ACC_{\mathcal{S}_1} = ACC_{avg} + (1 - ACC_{avg})/9 * (|\mathcal{S}_1| - 1) = 31.5\%, \quad (4)$$

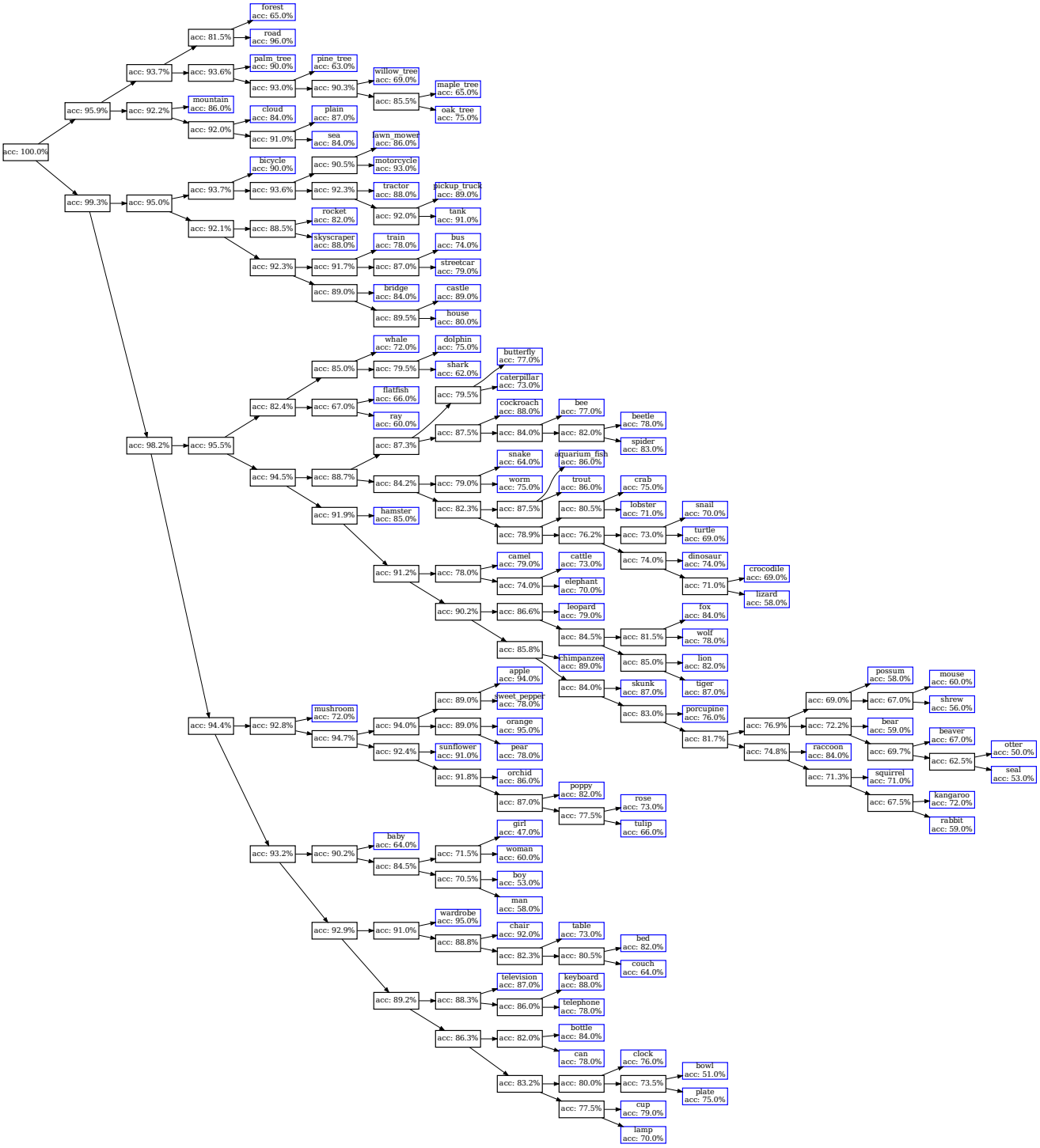


Figure 9. The decision tree produced after the 37-th layer in WRN_40_2 on CIFAR-100. The leaf nodes are painted with blue borders. "Acc" denotes the accuracy via nearest neighbor search in the feature subspace after linear discriminant analysis.

and the accuracy of the random guess for data in $\mathcal{S}_2$ is

$$ACC_{\mathcal{S}_2} = ACC_{avg} + (1 - ACC_{avg})/9 * (|\mathcal{S}_2| - 1) = 82.8\%. \quad (5)$$

It can be seen that the random-guess accuracies of $\mathcal{S}_1$ and $\mathcal{S}_2$ are both significantly lower than the actual accuracy (59.4% for $\mathcal{S}_1$ and 87.4% for $\mathcal{S}_2$) shown in Figure 3 (a), verifying

the coarse-to-fine decision process underlying deep neural networks.

The dendrograms produced from WRN_40_2 on CIFAR-100 are depicted in Figure 4 and 5. The corresponding decision trees are provided in Figure 6, 7, 8 and 9, respectively. On CIFAR-100, similar results are also observed to those from CIFAR-10. All these results validate the universality of the coarse-to-fine decision process underlying deep neural networks.

References

[1] S. Ahn, Shell Xu Hu, A. Damianou, N. Lawrence, and Z. Dai. Variational information distillation for knowledge transfer. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9155–9163, 2019. 2

[2] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1

[3] Geoffrey E. Hinton. Visualizing high-dimensional data using t-sne. 2008. 2

[4] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531, 2015. 2

[5] Jangho Kim, Seonguk Park, and Nojun Kwak. Paraphrasing complex network: Network compression via factor transfer. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 2760–2769. Curran Associates, Inc., 2018. 2

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. 7

[7] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3962–3971, 2019. 2

[8] N. Passalis and A. Tefas. Learning deep representations with probabilistic knowledge transfer. In *ECCV*, 2018. 2

[9] Baoyun Peng, Xiao Jin, Jiaheng Liu, Shunfeng Zhou, Y. Wu, Y. Liu, Dong sheng Li, and Z. Zhang. Correlation congruence for knowledge distillation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5006–5015, 2019. 2

[10] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *CoRR*, abs/1412.6550, 2015. 2

[11] Mark Sandler, A. Howard, Menglong Zhu, A. Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018. 1

[12] K. Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015. 1

[13] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *International Conference on Learning Representations*, 2020. 1

[14] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1365–1374, 2019. 2

[15] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7130–7138, 2017. 2

[16] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *ArXiv*, abs/1605.07146, 2016. 1

[17] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2017. 2

[18] X. Zhang, X. Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6848–6856, 2018. 1