Supplementary: Deep ensemble learning by diverse knowledge distillation for fine-grained object classification

Naoki Okamoto, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi

Chubu University, Kasugai, Aichi, Japan
{naok,hirakawa}@mprg.cs.chubu.ac.jp,{takayoshi,fujiyoshi}@isc.chubu.ac.jp

1 Comparison of loss designs

In this section, we show the effect that loss designs have on ensembles in terms of bringing knowledge closer and separating knowledge. We trained four networks and compared the ensemble accuracy using them. We used ResNet-18 [2] as the network, the probability distribution and attention map as the knowledge, and Stanford Dogs [3] as the dataset. The attention map was created from the output of ResBlock4 using attention transfer [8]. The loss design for the probability distribution used KL-divergence (KL) and cosine similarity (cos). The loss design for the attention map used mean square error (MSE) and cosine similarity (cos).

Tables 1 and 2 show the results of the loss design for the probability distribution and the attention map. "+" means the loss design for bringing knowledge closer and "−" means the loss design for separating knowledge. With the exception of several loss designs, the ensemble accuracy did not change significantly with the loss design. Loss designs that use division have the possibility of dividing by zero. Therefore, to train the network as a minimization problem, we selected different loss designs for bringing knowledge closer and separating knowledge.

2 Types of knowledge and effects of loss design

In this section, we show the effect of knowledge type and loss design on network accuracy. In Sec. 3.2 of the main paper, the ensemble accuracy (Fig. 3 in the

Table 1: Comparison of ensemble accuracy for different loss designs with probability distribution [%].

Loss design	Knowledge	Ensemble accuracy
KL	+	69.95
−cos	+	70.00
1/cos	+	69.12
cos	−	69.88
−KL	−	57.87
1/KL	−	67.88

Table 2: Comparison of ensemble accuracy for different loss designs with attention map [%].

Loss design	Knowledge	Ensemble accuracy
MSE	+	68.91
−cos	+	68.93
1/cos	+	68.98
cos	−	70.35
−MSE	−	64.73
1/MSE	−	68.50

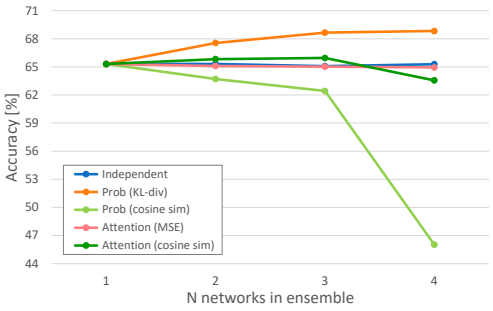


Fig. 1: Network accuracy by diverse knowledge distillation [%].

main paper) is comparable depending on the loss design. The accuracy of the network in Sec. 3.2 is shown in Fig. 1. The accuracy of the network tends to vary depending on the loss design. Therefore, we think that different loss designs have different learning effects. Separating the probability distributions causes a decrease in the accuracy of the network, whereas separating the attention maps prevents a decrease in the accuracy of the network.

3 Process of graph optimization

Algorithm 1 shows process of graph optimization. Each graph is evaluated at the timing of 2^k epochs. If the ensemble accuracy of the graph is higher than the median ensemble accuracy of previously evaluated graphs at the same epoch timing, training is continued to the next 2^k epochs and evaluated again. If the ensemble accuracy of the graph is lower than the median, the training of the graph is terminated. Then, a new graph is created for the structure that has not yet been evaluated, and training of the new graph is started from 1 epoch. These processes are repeated until the number of graphs created reaches 6,000.

4 Visualization of optimized graphs

In the main paper, we showed a graph structure that was automatically designed using Stanford Dogs. In this section, we show graph structures that were automatically designed using datasets other than Stanford Dogs. Figures 2, 3, 4, and 5 show graphs automatically designed using CIFAR-10 [5], CIFAR-100 [5], Caltech-UCSD Birds-200-2011 (CUB-200-2011) [7], and Stanford Cars [4]. A red node represents an ensemble node, a gray node represents a network node, and "Label" represents supervised labels. At each edge, the selected loss design and gate are shown, excluding the cutoff gate. The accuracy in parentheses is the result for the dataset used for automatic graph design.

Algorithm 1 Optimizing the graph of M network nodes

Require: Number of searches N_{search}, Number of GPUs N_{gpu}
Require: Training epochs T, Train set D_{train}, Test set D_{test}
Require: Networks set $\mathcal{F} : \{f_1, ..., f_M\}$, Search space of graphs $\mathcal{S} : \{S_1, ..., S_i\}$
Require: Random sampling function ϕ, Pruning function ASHA,
The function computing ensemble accuracy ψ

$A_{\text{best}} \leftarrow 0$
$N_{\text{end}} \leftarrow 0$
Start the asynchronous search in N_{gpu} environments
while $N_{\text{end}} \geq N_{\text{search}}$ **do**
  Initialization of networks $\mathcal{F}$
  $S \leftarrow \phi(\mathcal{S})$ // Selection of graph structure by random sampling
  for $t \leftarrow 1$ to T **do**
    $S(\mathcal{F}, D_{\text{train}})$ // Training of networks by graph
    $A_{\text{ens}} \leftarrow \psi(\mathcal{F}, D_{\text{test}})$ // Evaluation of ensemble accuracy
    // Judgement of pruning by ASHA
    if ASHA(A_{ens}) **then**
      break
    end if
  end for
  $A_{\text{ens}} \leftarrow \psi(\mathcal{F}, D_{\text{test}})$
  if $A_{\text{ens}} > A_{\text{best}}$ **then**
    $S_{\text{best}} \leftarrow S$
    $A_{\text{best}} \leftarrow A_{\text{ens}}$
  end if
  $N_{\text{end}} \leftarrow N_{\text{end}} + 1$
end while
return S_{best}

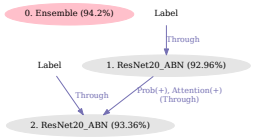


Fig. 2: Two-node graph optimized on CIFAR-10.

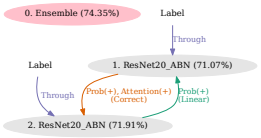
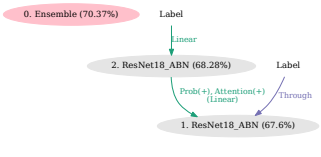
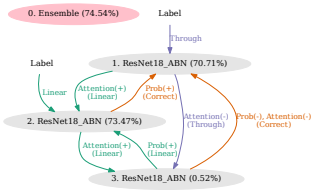


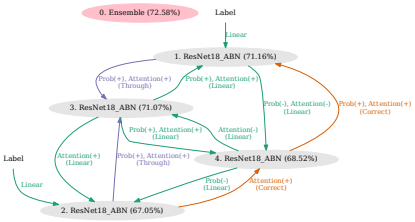
Fig. 3: Two-node graph optimized on CIFAR-100.



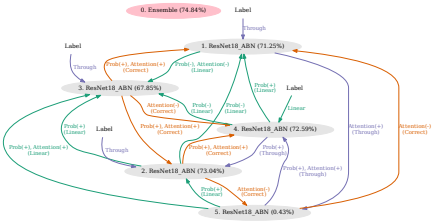
(a) Two nodes



(b) Three nodes

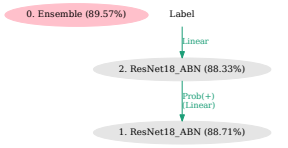


(c) Four nodes

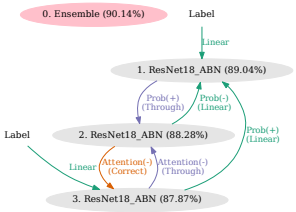


(d) Five nodes

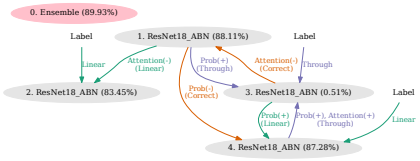
Fig. 4: Graph optimized on CUB-200-2011.



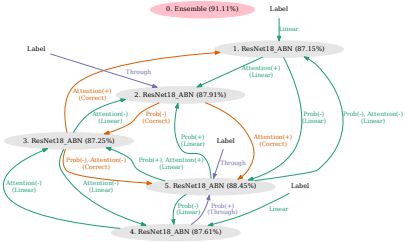
(a) Two nodes



(b) Three nodes



(c) Four nodes



(d) Five nodes

Fig. 5: Graph optimized on Stanford Cars.

5 Generalizability of graphs

In the main paper, we showed the generalizability of graph structures automatically designed using Stanford Dogs for four datasets. In this section, we evaluate five graphs automatically designed using five different datasets. We used attention branch network (ABN) [1] based on ResNet as the network. We used Stanford Dogs, Stanford Cars, CUB-200-2011, CIFAR-10, and CIFAR-100 as the datasets. Stanford Dogs, Stanford Cars, and CUB-200-2011 belong to the fine-grained object classification task. CIFAR-10 and CIFAR-100 belong to the general object classification task.

Table 3 shows the ensemble accuracy of the two-node graph. "Independent" is the result of an individually trained network. "DML" is the result of a network trained with deep mutual learning [9]. The bold text in the Ensemble column shows that the ensemble accuracy was higher than Independent and DML. The graphs automatically designed using the datasets of the fine-grained object classification task showed an improved ensemble accuracy, especially in the fine-grained object classification task. The graphs automatically designed using the datasets of the general object classification task showed an improved ensemble accuracy, especially in the general object classification task. We believe that there was generalizability in the graph structure when the problem set was the same and that optimization resulted in a graph structure that corresponded to the problem set.

Tables 4, 5, and 6 show the results for various numbers of network nodes, from two to five, between the fine-grained object classification tasks. We believe that there was generalizability in the graph structure even when the number of network nodes was increased. Focusing on the ensemble accuracy of "Ours," the ensemble accuracy varied depending on the dataset used for the automatic design. We believe that this is because the number of combinations of graph structures was huge, and automatic design using random search finally resulted in different graph structures.

Figures 6a and 6b show the attention maps of the five-node graph optimized by Stanford Dogs for training CUB-200-2011 and Stanford Cars, respectively. Each point in the graph is two-dimensional because the attention maps were reduced to two dimensions by UMAP [6]. We see that the attention maps of each dataset are similar even when training on a dataset different from the optimization. These dimensionally reduced maps have a similar trend for each node regardless of the sample. This indicates that the graph structure is generalizable in terms of the attention map.

Table 3: Ensemble accuracy of reused two-node graphs optimized on another dataset [%].

Method	Training Graph	Optimizing Graph	Ensemble
Independent	Stanford Dogs	-	70.90
DML	Stanford Dogs	-	71.45
Ours	Stanford Dogs	Stanford Dogs	**73.86**
Ours	Stanford Dogs	CUB-200-2011	**72.43**
Ours	Stanford Dogs	Stanford Cars	**72.79**
Ours	Stanford Dogs	CIFAR-100	**71.53**
Ours	Stanford Dogs	CIFAR-10	70.08
Independent	CUB-200-2011	-	65.26
DML	CUB-200-2011	-	66.90
Ours	CUB-200-2011	Stanford Dogs	**72.06**
Ours	CUB-200-2011	CUB-200-2011	**69.81**
Ours	CUB-200-2011	Stanford Cars	**71.27**
Ours	CUB-200-2011	CIFAR-100	66.43
Ours	CUB-200-2011	CIFAR-10	66.42
Independent	Stanford Cars	-	88.49
DML	Stanford Cars	-	88.89
Ours	Stanford Cars	Stanford Dogs	**89.76**
Ours	Stanford Cars	CUB-200-2011	**89.50**
Ours	Stanford Cars	Stanford Cars	**89.44**
Ours	Stanford Cars	CIFAR-100	**88.90**
Ours	Stanford Cars	CIFAR-10	88.63
Independent	CIFAR-100	-	73.16
DML	CIFAR-100	-	73.61
Ours	CIFAR-100	Stanford Dogs	72.19
Ours	CIFAR-100	CUB-200-2011	**73.66**
Ours	CIFAR-100	Stanford Cars	72.53
Ours	CIFAR-100	CIFAR-100	**74.18**
Ours	CIFAR-100	CIFAR-10	73.37
Independent	CIFAR-10	-	93.99
DML	CIFAR-10	-	93.97
Ours	CIFAR-10	Stanford Dogs	93.87
Ours	CIFAR-10	CUB-200-2011	**94.09**
Ours	CIFAR-10	Stanford Cars	93.93
Ours	CIFAR-10	CIFAR-100	**94.37**
Ours	CIFAR-10	CIFAR-10	**94.15**

Table 4: Ensemble accuracy on Stanford Dogs [%].

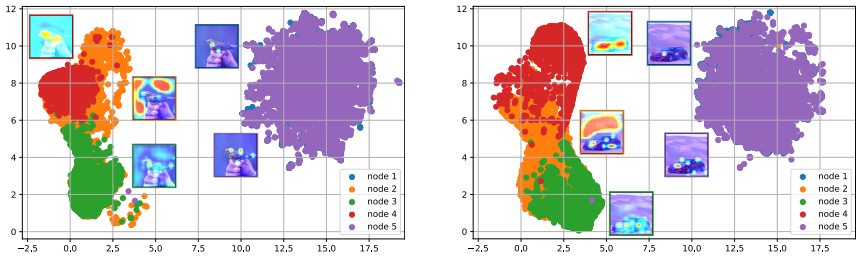
Method	Optimizing Graph	Number of nodes			
		2	3	4	5
Independent	-	70.09	71.41	72.06	72.32
Ours	Stanford Dogs	**73.86**	73.41	**74.16**	74.14
Ours	CUB-200-2011	72.43	**73.78**	74.03	74.55
Ours	Stanford Cars	72.79	73.35	72.86	**74.56**

Table 5: Ensemble accuracy on CUB-200-2011 [%].

Method	Optimizing Graph	Number of nodes			
		2	3	4	5
Independent	-	65.26	65.27	66.40	66.66
Ours	Stanford Dogs	**72.06**	71.82	**73.03**	72.13
Ours	CUB-200-2011	69.81	**74.17**	71.27	**74.05**
Ours	Stanford Cars	71.19	72.56	70.26	73.43

Table 6: Ensemble accuracy on Stanford Cars [%].

Method	Optimizing Graph	Number of nodes			
		2	3	4	5
Independent	-	88.49	89.23	89.55	89.48
Ours	Stanford Dogs	**89.76**	89.94	**89.98**	90.41
Ours	CUB-200-2011	89.58	**90.39**	89.95	**90.81**
Ours	Stanford Cars	89.44	90.04	89.57	90.73



(a) CUB-200-2011 (b) Stanford Cars

Fig. 6: Attention map of five nodes optimized by Stanford Dogs trained on other datasets with dimensionality reduction by UMAP.

Table 7: Comparison of ensemble accuracy by nodes used in ensemble. [%].

Used for ensemble					Ensemble
node 1	node 2	node 3	node 4	node 5	
✓	✓	✓	✓	✓	**74.50**
✓	✓	✓	✓		74.30
✓	✓	✓		✓	74.06
✓	✓		✓	✓	74.31
✓		✓	✓	✓	74.23
	✓	✓	✓	✓	**74.44**
✓	✓	✓			73.76
✓	✓		✓		73.80
✓	✓			✓	73.71
✓		✓	✓		73.87
✓		✓		✓	73.28
✓			✓	✓	73.65
	✓	✓	✓		73.51
	✓	✓		✓	**73.88**
	✓		✓	✓	73.40
		✓	✓	✓	73.01
✓	✓				72.54
✓		✓			72.69
✓			✓		**72.74**
✓				✓	72.28
	✓	✓			72.62
	✓		✓		72.69
	✓			✓	72.18
		✓	✓		72.56
		✓		✓	71.52
			✓	✓	72.44

6 Relationship between network nodes and ensemble accuracy

On the basis of the trend in Fig. 7 of the main paper, training with the five-node graph automatically designed using Stanford Dogs (Fig. 6d of the main paper) lead to a different attention map being acquired for each network node. As a result, node 2 became a network that strongly focused on the background, which may have a negative impact on the ensemble. In this section, we show the impact on the ensemble accuracy of each network node trained with the five-node graph automatically designed using Stanford Dogs. We used ABN based on ResNet as the network and Stanford Dogs as the dataset.

Table 7 shows the ensemble accuracy using selected network nodes. "✓" means the network node used in the ensemble. The bold text in the Ensemble column shows the highest ensemble accuracy for each number of network nodes. The ensemble accuracy was highest when all network nodes were used. Therefore, we believe that the five-node graph is a training method that achieves high performance by acquiring different attention maps among network nodes that cooperate with each other.

References

1. Fukui, H., Hirakawa, T., Yamashita, T., Fujiyoshi, H.: Attention branch network: Learning of attention mechanism for visual explanation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2019)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
3. Khosla, A., Jayadevaprakash, N., Yao, B., Fei-Fei, L.: Novel dataset for fine-grained image categorization. In: First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition (2011)
4. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition. Sydney, Australia (2013)
5. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. rep., Citeseer (2009)
6. McInnes, L., Healy, J., Saul, N., Grossberger, L.: Umap: Uniform manifold approximation and projection. The Journal of Open Source Software **3**(29), 861 (2018)
7. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)
8. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In: International Conference on Learning Representations (2017)
9. Zhang, Y., Xiang, T., Hospedales, T.M., Lu, H.: Deep mutual learning. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)