2024 18th International Conference on Automatic Face and Gesture Recognition (FG)

Adaptive Cross-architecture Mutual Knowledge Distillation

Jianyuan Ni[1], Hao Tang[2], Yuzhang Shang[3], Bin Duan[3], Yan Yan[3]
[1]Texas State University, USA [2]Carnegie Mellon University, USA [3]Illinois Institute of Technology, USA

***Abstract*— Knowledge distillation (KD), which distills knowledge from complex networks (teacher) to lightweight (student) networks, has been actively studied recently. Despite previous studies have proposed several advanced KD losses or intricate training strategies, the core concept of KD proves ineffective if the student model is too weak to mimic the teacher's performance. In this study, we aim to narrow the performance discrepancy between Transformer-based teacher and student models by incorporating the inductive biases of several heterogeneous student models. To this end, we put forward a novel cross-architecture knowledge distillation approach called Adaptive Cross-architecture Mutual Knowledge Distillation (ACMKD), which tries to mitigate the performance gap issue using a multi-students mutual learning strategy. Specifically, we utilize three mainstream models associated with various inductive biases (CNN, INN, and Transformer) as the student models. In addition, we propose an effective attention similarity mechanism to facilitate the student models in mimicking specific portions of the teacher model. Drawing inspiration from the Cannikin Law, we devise a unique second-stage KD process that dynamically enables the weakest student model to learn from other stronger student models again. We validate our proposed methods on ImageNet and CIFAR100 datasets, and the results confirm that our ACMKD method significantly narrows the performance gap compared to other KD methods.**

I. INTRODUCTION

Over recent decades, deep neural networks (DNNs) have shown remarkable success across various tasks [12], [30]. However, deploying such large models on resource-limited embedded systems poses challenges. Knowledge distillation (KD) addresses this by enhancing the performance of smaller networks (*i.e.*, student model) by emulating more complex networks (*i.e.*, teacher model) [9]. Yet, KD sometimes fails to achieve reliable accuracy performance, especially when the student model's capacity is insufficient compared to the teacher model [19], [4], [10]. To mitigate this performance discrepancy (referred to as the performance gap in this study), approaches like online KD [37], [38], where multiple student models are trained simultaneously to achieve better performance, has emerged. However, those works mainly focus on the homologous-architecture KD process and fail to consider scenarios when teacher and student models have heterogeneous architectures.

While Transformer models achieve impressive performance when trained with large-scale datasets, they struggle with medium-scale or small-scale datasets due to lacking certain inductive biases [5], [33], [29], [6]. To address this problem, KD processes have been applied to Transformer models [29], [32], [1], [14], [21]. For example, DeiT [29] leveraged a powerful CNN teacher to enhance Transformer performance. However, these efforts mainly focused on improving the Transformer-based student model's accuracy. None of them try to solve the performance gap problem and answer the question: *How to take advantage of inductive bias to reduce the performance gap between Transformer-based teacher and heterogeneous student models in KD process?*

In this paper, we show an intriguing interaction between three heterogeneous models: **CNN**, **INN**, and **Transformer**. We refer to the KD framework between various models as Adaptive Cross-architecture Mutual Knowledge Distillation (**ACMKD**). The schematic overview of the proposed method is shown in Fig.1. We first adopt one Transformer model (Swin-T [16]) as the teacher and employ the mutual learning strategy to train three heterogeneous student models (Swin-T [16], ResNet50 [8], and RedNet [13]) so that the student models which associated with various inductive biases can learn from each other. Additionally, we present an attention similarity mechanism that guides student models on how to imitate the teacher's performance. We further design a dynamic second-stage distillation strategy that enables the student model with the lowest accuracy performance can learn from the other higher-performing student models once again. Our contributions can be summarized as follows: 1) We propose a novel method of distilling the Transformer-based teacher model into heterogeneous students with different inductive biases to alleviate the issue of performance gap in the KD process; 2) We show how adding a second-stage distillation strategy allows the student with the lowest accuracy performance to dynamically learn from other stronger student models, leading to higher accuracy performance of ensembled student models eventually; 3) Our proposed ACMKD outperforms all previous Transformer-based KD methods in addressing the performance gap problem on both ImageNet and CIFAR100 datasets.

II. RELATED WORK

Despite the success of KD, current studies found that a large teacher is often detrimental to the KD process due to the capacity inconsistency problem [19], [4], [10], [7] between the teacher and the student. To address this issue, researchers have proposed to employ an intermediate-sized network [7], or a TA network [19] to enhance the distillation performance. More recently, some studies have considered the KD from another perspective, such as the KD process with multiple teachers [24], [34], [27], [15], [21] or ensemble-based KD [37], [2], [18], [26], [36]. For instance, You *et al.* fused multiple teachers equally to accomplish the KD process [34]. Liu *et al.* proposed to transfer the knowledge of the intermediate layer from various teacher models to guide a group of layers in the student network [15]. Zhang *et al.* claimed that by using cross-entropy loss between each

979-8-3503-9494-8/24/$31.00 ©2024 IEEE

2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG) | 979-8-3503-9494-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/FG59268.2024.10581969

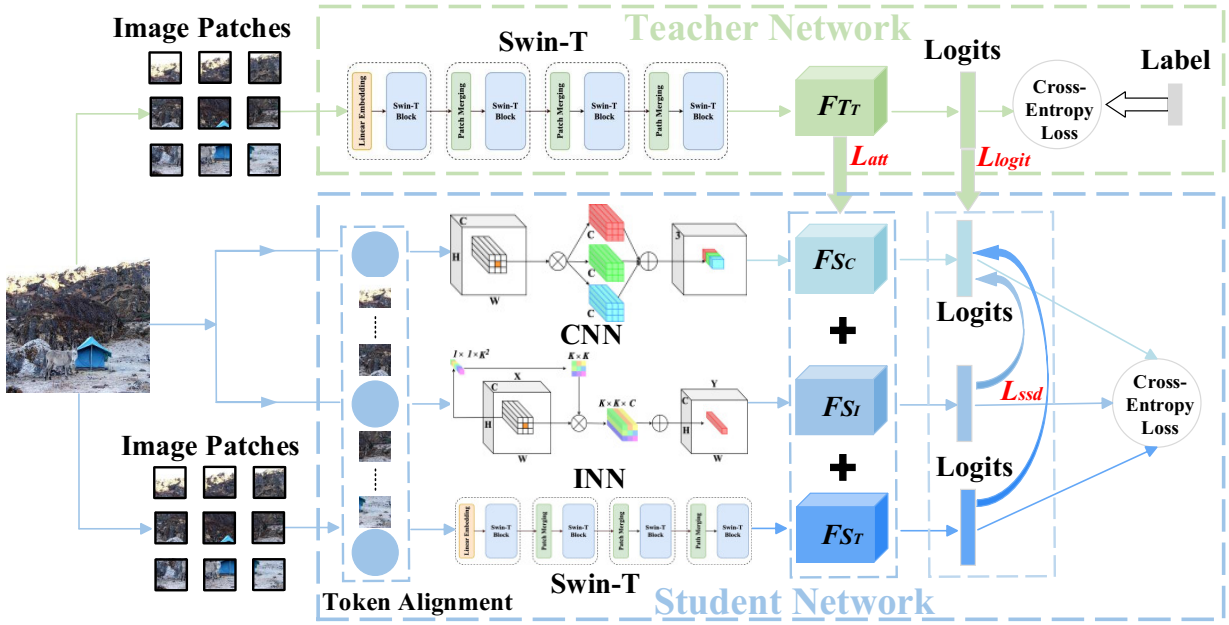

Fig. 1: Schematic overview of the proposed ACMKD method.

pair of student models, the peer students could learn from each other [37]. Son *et al.* presented a densely guided KD process to bridge the performance between teacher and student network [26]. However, all those works use identical or similar architecture in the KD process. In this study, our proposed method differs from existing methods as we use heterogeneous student models to narrow the performance gap from the Transformer-based teacher model.

III. METHODOLOGY

A. Preliminaries

In the KD process, given a teacher model T and a student model S, the softened output $\tilde{y}^T$ produced by the teacher model is considered as high-level knowledge. The loss $\mathcal{L}_{KD}$ when training a student model can be defined as follows:

$$\mathcal{L}_{KD} = \tau^2 L_{KL}(\tilde{y}^S, \tilde{y}^T), \tag{1}$$

where τ is the temperature parameter to control the softened output and $\mathcal{L}_{KL}$ is the Kullback-Leibler (KL) divergence loss [9] and we set $\tau = 1$ in this study. Each network's output is $\tilde{y}^T = softmax(y^T/\tau)$ and $\tilde{y}^S = softmax(y^S/\tau)$. Here y^T and y^T refer to the teacher and student logits, respectively. As a result, the final KD loss function $\mathcal{L}$ is written with the balancing parameter λ as follows:

$$\mathcal{L} = (1-\lambda)\mathcal{L}_{CE} + \lambda\mathcal{L}_{KD}, \tag{2}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss between the student's logits and the label y.

B. Cross-architecture Multi-students KD

In this study, we adopt three mainstreams of models associated with heterogeneous architectures as student networks, including CNN ($\mathcal{S}_C$), INN ($\mathcal{S}_I$), and Transformer ($\mathcal{S}_T$). Also, we select the Transformer-based model ($\mathcal{T}_T$) as the teacher network in this study. The learning objective is expressed as a weighted combination of three Kullback-Leibler divergence losses ($\mathcal{L}_{KD}$) and a cross-entropy loss ($\mathcal{L}_{CE}$). According to Eq. (2), the KD logit loss function $\mathcal{L}_{logit}$ is derived as follows:

$$\mathcal{L}_{logit} = \lambda_0\mathcal{L}_{CE} + \lambda_1\mathcal{L}_{KD}^{T_T\rightarrow\mathcal{S}_C} + \lambda_2\mathcal{L}_{KD}^{T_T\rightarrow\mathcal{S}_I} + \lambda_3\mathcal{L}_{KD}^{T_T\rightarrow\mathcal{S}_T}, \tag{3}$$

where λ_0, λ_1, λ_2, and λ_3 are the weights balancing the importance of four loss terms. In this study, we set $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3 = 1$ according to the previous study [21].

Inspired by [25], we let $\mathcal{F}_{S_{C_i}}$ be the intermediate feature outputs from the i-th layer of the CNN student model. Similarly, $\mathcal{F}_{S_{I_i}}$ and $\mathcal{F}_{S_{T_i}}$ represent the intermediate feature outputs from the i-th layer of the INN and Transformer, respectively. Attention maps regarding $\mathcal{F}_{S_{C_i}}$ can be derived as the $\mathcal{A}_i(F_{S_{C_i}})$, where $A_i(\cdot)$ is the attention model. As a result, attention mechanisms can be expressed as follows:

$$\begin{aligned} \mathcal{F}'_{S_{C_i}} &= A_i^c(F_{S_{C_i}}) \otimes \mathcal{F}_{S_{C_i}}, \\ \mathcal{F}''_{S_{C_i}} &= A_i^s(F'_{S_{C_i}}) \otimes F'_{S_{C_i}}, \end{aligned} \tag{4}$$

where $A_i^c(\cdot)$ and $A_i^s(\cdot)$ are channel and spatial attention models, respectively. Since we have one teacher (F_{T_T}) and three student models (F_{S_C}, F_{S_I}, F_{S_T}) in this study, the proposed attention similarity loss $\mathcal{L}_a$ is expressed as follows:

$$\begin{aligned} \rho_i &= 1 - \langle A_{T_T,i}(F_{T_T,i}), A_{S_C,i}(F_{S_C,i}), A_{S_I,i}(F_{S_I,i}), \\ &\quad A_{S_T,i}(F_{S_T,i})\rangle \\ &= 1 - \frac{A_{T_T,i}(F_{T_T,i})}{\|A_{T_T,i}(F_{T_T,i})\|_2} \cdot \frac{A_{S_C,i}(F_{S_C,i})}{\|A_{S_C,i}(F_{S_C,i})\|_2} \cdot \\ &\quad \frac{A_{S_I,i}(F_{S_I,i})}{\|A_{S_I,i}(F_{S_I,i})\|_2} \cdot \frac{A_{S_T,i}(F_{S_T,i})}{\|A_{S_T,i}(F_{S_T,i})\|_2}, \end{aligned} \tag{5}$$

where ρ_i is the cosine distance between attention maps from the i-th layer of teacher and student models. $\langle\cdot,\cdot\rangle$ represent

the cosine similarity; $\|\cdot\|_2$ denotes L2-norm, and T_T, S_C, S_I, and S_T denote the teacher and student models, respectively. Consequently, the attention-based distillation loss function $\mathcal{L}_{att}$ which averages the cosine distance for channel and spatial attention maps between teacher ($\mathcal{T}_T$) and student models ($\mathcal{S}_C$, $\mathcal{S}_I$, $\mathcal{S}_T$) is derived as follows:

$$\mathcal{L}_{att} = \sum_{i=1}^{N} \frac{(\rho_i^s + \rho_i^c)}{2}, \tag{6}$$

where N is the number of layers utilized for the distillation. Similar to [25], we distilled the attention maps for every Transformer layer from the teacher model with the exception of the last MLP Head layer.

C. Second-stage Distillation strategy

Inspired by the Cannikin Law, we further propose another distillation framework (Second-Stage Distillation) where the better two students will try to help the weakest one once again in terms of accuracy performance. For ease of understanding, after n epochs, we assume $\mathcal{S}_T$ and $\mathcal{S}_C$ achieve higher accuracy, and they were selected as the two stronger student models. In contrast, the $\mathcal{S}_I$, which has the lowest accuracy performance, is assigned as the student model. Note that our proposed second-stage distillation method is conducted as follows after the traditional distillation process in the same epochs. During the training, only the parameters of the weakest student get updated while the two strong students remain frozen. As a result, the final loss of second-stage distillation $\mathcal{L}_{ssd}$ is derived as follows:

$$\mathcal{L}_{ssd} = \mathcal{L}^{S_C \to S_I} + \mathcal{L}^{S_T \to S_I}, \tag{7}$$

where the right arrow at the subscript indicates the KD direction. Eq. (7) can then be expressed in the same format as Eq. (2) as follows:

$$\mathcal{L}_{ssd} = \lambda_4 \mathcal{L}_{CE} + \lambda_5 \mathcal{L}_{KD}^{S_C \to S_I} + \lambda_6 \mathcal{L}_{KD}^{S_T \to S_I}, \tag{8}$$

In this way, the weakest student model $\mathcal{S}_{Lowest}$ is selected based on the prediction score, and complementary knowledge from the other stronger student networks can be dynamically transferred to this student model. Again, we set λ_4 = λ_5 = λ_6 = 1 according to the previous study [21].

In summary, we use the original KD loss $\mathcal{L}_{logit}$ and augment it to include the proposed attention similarity loss L_{att} as well as the second-stage distillation loss $\mathcal{L}_{ssd}$, to train the student network and the final loss for the student models is defined as follow:

$$\mathcal{L} = \mathcal{L}_{logit} + \alpha \mathcal{L}_{att} + \beta \mathcal{L}_{ssd}, \tag{9}$$

where α and β are the tunable hyperparameters to balance the loss terms for the student networks.

IV. Experiments

A. Dataset and Implementation

We evaluate the proposed method on two datasets: CIFAR100 [11], and ImageNet [23]. We train all the experiments on four Nvidia GeForce GTX 1080 Ti GPUs using PyTorch. To guarantee a reproducible behavior, all training procedures are initialized with a fixed random seed. For the CIFAR100 dataset, the batch size is 64, and the total number of epochs is 200. The learning rate is initialized as 0.1 and multiplied by 0.1 at epoch 100, and epoch 150 [14]. For ImageNet, the batch size is 32, and the total number of epochs is 300. We use AdamW [17] as the optimizer with a learning rate equal to 0.001 and weight decay equal to 0.05. The token alignment is adopted according to [21]. The final student accuracy result was obtained by averaging three student models similar to [37].

Type	Method	Model	T (%)	S (%)	Gap (%)
CNN	Logit [9]	ResNet101 → ResNet50	77.29	76.59	-0.70
	FitNet [22]			76.45	-0.84
	AT [35]			76.64	-0.65
	RKD [20]			76.71	-0.58
	CRD [28]			76.86	-0.43
	ReviewKD [3]			76.97	**-0.32**
	AFD [31]			76.82	-0.47
Trans.	MINILM [32]	Swin-T→ViT-B	81.20	79.02	-2.18
	DeiT [29]			79.21	-1.99
	CAKD [14]			78.82	-2.38
X-arc.	CAKD [14]	Swin-T→ResNet50	81.20	79.87	-1.33
	Ours	ACMKD	81.20	**80.47**	-0.73

TABLE I: Performance gap comparison between our proposed method and state-of-the-art methods on ImageNet [23]. X-arc. denotes Cross-Architecture and → denotes the KD direction. T represents the teacher model, and S represents the student model. Gap denotes the top-1 accuracy performance gap between the teacher and student model.

B. Performance Comparison

Table I presents the KD results on the ImageNet dataset. It can be seen that the student model (80.47%) in the proposed ACMKD method has higher accuracy performance than any of the other student models in CNN-CNN KD methods (76.45%-76.97%). In order to make a fair comparison, we use the same teacher model (Swin-T) for Transformer-Transformer models. The student model from the proposed ACMKD approach gains a higher performance (0.56%-1.65%) compared with the Transformer-based group. This indicates that existing Transformer-based KD methods fail to take full advantage of the various teacher models, regardless of the fact that they can be adapted to the cross-architecture scenario. In terms of cross-architecture models, the results from the proposed method also surpass all the other cross-architecture KD results by 0.60%. This indicates that the cross-architecture KD proposed in the ACMKD method can obtain higher promotion than the transformer homologous-architecture KD. However, the performance gap (0.73 %) in the proposed ACMKD is still higher than the ones from most of the CNN-based KD methods (0.32%-0.84%). This indicates that the conventional homologous-architecture KD can reduce the performance gap more effectively due to the similar inductive bias [21]. However, it is still worth noting that our proposed ACMKD method has the lowest performance gap compared to any other Transformer-based or cross-architecture KD methods.

Table II presents the KD results on the CIFAR100 dataset. It can be seen that our method presents the best perfor-

Type	Method	Model	T (%)	S (%)	Gap (%)
CNN	Logit [9]	ResNet101 → ResNet50	90.76	88.98	-1.59
	FitNet [22]			88.45	-2.31
	AT [35]			89.04	-1.72
	RKD [20]			89.16	-1.60
	CRD [28]			89.39	-1.37
	ReviewKD [3]			89.91	**-0.85**
	AFD [31]			89.44	-1.32
Trans.	MINILM [32]	Swin-T→ViT-B	92.59	90.22	-2.37
	DeiT [29]	Swin-T→ViT-B		90.67	-1.92
	CAKD [14]	Swin-T→ViT-B		90.96	-1.63
X-arc.	CAKD [14]	Swin-T→ResNet50	92.59	88.06	-2.53
	Ours	ACMKD	92.59	**91.19**	-1.40

TABLE II: Performance gap comparison between our proposed method and state-of-the-art methods on CIFAR100 [11]. X-arc. denotes cross-architecture and → denotes the KD direction. T represents the teacher model, and S represents the student model. Gap denotes the top-1 accuracy performance gap between the teacher and student model.

mance of the student model. Similar to the settings of the ImageNet dataset, we compare the proposed ACMKD with two homogeneous-architecture methods as well as one cross-architecture method. The student models from our ACMKD framework outperform all student models in all three KD groups. This result confirms the proposed ACMKD method can encourage the student models associated with heterogeneous architectures to learn both local spatial features from CNN/INN and complementary global features reported in the previous study [14]. As a result, the student models retain the highest accuracy performance among all methods.Similarly, the performance gap in CNN-based KD methods has the lowest accuracy difference (0.85%) compared with the one in the proposed ACMKD method. This reduction indicated that the CNN-based network still has competitive accuracy performance compared to Transformer-based models in medium-scale datasets [29], [32].

C. Ablation Study

To evaluate the effect of each component in the proposed loss function, further experiments on the ImageNet dataset [12] are investigated. Specially, we compare the ACMKD with two student's baselines: 1) a student's baseline model, which learns directly from the pre-trained teacher model without the second-stage distillation process (W/O SSD); 2) a student baseline model which contains the second-stage distillation process while lacking the proposed attention-based similarity mechanism (W/O ASM). As shown in Table III, the student model from the proposed ACMKD model outperforms the student's baseline model by 0.19%-0.36%, providing that the attention-based similarity mechanism and the second-stage distillation are able to reduce the performance gap and thus improve the accuracy performance of the student models. In addition, the second-stage distillation loss L_{ssd} contributes about 0.17% more to accuracy improvement as compared to attention similarity loss L_{att}, which sheds light on the importance of helping the lowest student models when various student networks exist in KD process. In order to evaluate the effect of the proposed second-stage distillation, further attention map visualization

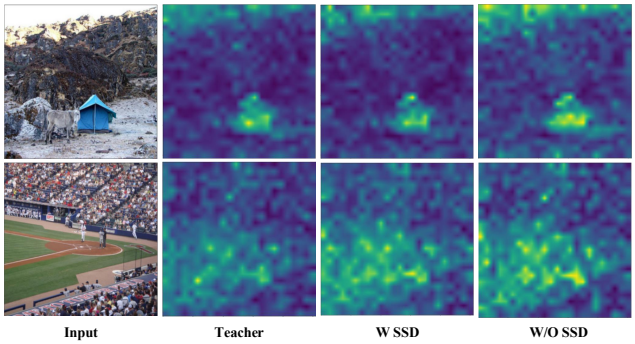

Fig. 2: Comparison results of attention map visualizations with and without proposed second-stage distillation (SSD). W represents with, and W/O denotes without.

Type	ASM	SSD	Param (M)	Top-1 Accuracy (%)
Teacher			28	81.20
Student		✓	25	80.28
	✓		22	80.11
	✓	✓	28	80.47
Baseline (ResNet50)			19	76.41
Baseline (RedNet50)			11	78.56
Baseline (Swin-T)			20	79.02

TABLE III: Performance comparison of each loss term on ImageNet [23]. ASM represents the proposed attention-based similarity mechanism and SSD represents the proposed second-stage distillation. A check mark ✓ represents a loss term of the specified type presented.

experiments are conducted. As shown in Fig.2, the attention map results from student baselines (W SSD) have a more similar attention map to the teacher model. In contrast, the student baseline without SSD has a larger different attention map compared to the teacher model. This validates that complementary knowledge from other stronger students can effectively guide the weakest student on where to mimic the teacher's performance, ultimately facilitating a more effective learning process.

V. Conclusion

In this study, we present an adaptive cross-architecture mutual knowledge distillation (ACMKD) framework. It aims to narrow the performance gap between teacher and student models by leveraging various inductive biases within mutual KD learning. We employ a Transformer-based model as the teacher, which inherently lacks certain inductive bias. We also propose an attention similarity mechanism and a second-stage distillation to further close this performance gap. We conduct evaluations of the ACMKD method on ImageNet and CIFAR100 datasets, demonstrating that diverse models offer unique data perspectives to student models, ultimately enhancing their accuracy in the mutual KD learning process.

VI. Acknowledgements

This research is supported by NSF SCH-2123749 and SCH-2123521 Collaborative Research. This article solely reflects the opinions and conclusions of its authors and not the funding agents.

REFERENCES

[1] G. Aguilar, Y. Ling, Y. Zhang, B. Yao, X. Fan, and C. Guo. Knowledge distillation from internal representations. In *AAAI*, volume 34, pages 7350–7357, 2020.

[2] D. Chen, J.-P. Mei, C. Wang, Y. Feng, and C. Chen. Online knowledge distillation with diverse peers. In *AAAI*, volume 34, pages 3430–3437, 2020.

[3] P. Chen, S. Liu, H. Zhao, and J. Jia. Distilling knowledge via knowledge review. In *CVPR*, pages 5008–5017, 2021.

[4] J. H. Cho and B. Hariharan. On the efficacy of knowledge distillation. In *ICCV*, pages 4794–4802, 2019.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[6] S. d'Ascoli, H. Touvron, M. L. Leavitt, A. S. Morcos, G. Biroli, and L. Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *ICML*, pages 2286–2296. PMLR, 2021.

[7] M. Gao, Y. Shen, Q. Li, and C. C. Loy. Residual knowledge distillation. *arXiv preprint arXiv:2002.09168*, 2020.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[9] G. Hinton, O. Vinyals, J. Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

[10] X. Jin, B. Peng, Y. Wu, Y. Liu, J. Liu, D. Liang, J. Yan, and X. Hu. Knowledge distillation via route constrained optimization. In *ICCV*, pages 1345–1354, 2019.

[11] A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[13] D. Li, J. Hu, C. Wang, X. Li, Q. She, L. Zhu, T. Zhang, and Q. Chen. Involution: Inverting the inherence of convolution for visual recognition. In *CVPR*, pages 12321–12330, 2021.

[14] Y. Liu, J. Cao, B. Li, W. Hu, J. Ding, and L. Li. Cross-architecture knowledge distillation. *arXiv preprint arXiv:2207.05273*, 2022.

[15] Y. Liu, W. Zhang, and J. Wang. Adaptive multi-teacher multi-level knowledge distillation. *Neurocomputing*, 415:106–113, 2020.

[16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pages 10012–10022, 2021.

[17] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[18] A. Malinin, B. Mlodozeniec, and M. Gales. Ensemble distribution distillation. *arXiv preprint arXiv:1905.00076*, 2019.

[19] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *AAAI*, volume 34, pages 5191–5198, 2020.

[20] W. Park, D. Kim, Y. Lu, and M. Cho. Relational knowledge distillation. In *CVPR*, pages 3967–3976, 2019.

[21] S. Ren, Z. Gao, T. Hua, Z. Xue, Y. Tian, S. He, and H. Zhao. Co-advise: Cross inductive bias distillation. In *CVPR*, pages 16773–16782, 2022.

[22] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[23] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

[24] B. B. Sau and V. N. Balasubramanian. Deep model compression: Distilling knowledge from noisy teachers. *arXiv preprint arXiv:1610.09650*, 2016.

[25] S. Shin, J. Lee, J. Lee, Y. Yu, and K. Lee. Teaching where to look: Attention similarity knowledge distillation for low resolution face recognition. In *ECCV*, pages 631–647. Springer, 2022.

[26] W. Son, J. Na, J. Choi, and W. Hwang. Densely guided knowledge distillation using multiple teacher assistants. In *ICCV*, pages 9395–9404, 2021.

[27] X. Tan, Y. Ren, D. He, T. Qin, Z. Zhao, and T.-Y. Liu. Multilingual neural machine translation with knowledge distillation. *arXiv preprint arXiv:1902.10461*, 2019.

[28] Y. Tian, D. Krishnan, and P. Isola. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

[29] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, pages 10347–10357. PMLR, 2021.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017.

[31] K. Wang, X. Gao, Y. Zhao, X. Li, D. Dou, and C.-Z. Xu. Pay attention to features, transfer learn faster cnns. In *ICLR*, 2019.

[32] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *NeurIPS*, 33:5776–5788, 2020.

[33] Y. Xu, Q. Zhang, J. Zhang, and D. Tao. Vitae: Vision transformer advanced by exploring intrinsic inductive bias. *NeurIPS*, 34:28522–28535, 2021.

[34] S. You, C. Xu, C. Xu, and D. Tao. Learning from multiple teacher networks. In *KDD*, pages 1285–1294, 2017.

[35] S. Zagoruyko and N. Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

[36] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *ICCV*, pages 3713–3722, 2019.

[37] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu. Deep mutual learning. In *CVPR*, pages 4320–4328, 2018.

[38] X. Zhu, S. Gong, et al. Knowledge distillation by on-the-fly native ensemble. *NeurIPS*, 31, 2018.