

Harmonizing knowledge Transfer in Neural Network with Unified Distillation

Yaomin Huang¹, Zaomin Yan¹, Chaomin Shen¹ Faming Fang¹, and Guixu Zhang¹, 

School of Computer Science, East China Normal University
{ymhuang, ZaoMingYan}@stu.ecnu.edu.cn
{cmshen, fmfang, gxzhang}@cs.ecnu.edu.cn

Abstract. Knowledge distillation (KD), known for its ability to transfer knowledge from a cumbersome network (teacher) to a lightweight one (student) without altering the architecture, has been garnering increasing attention. Two primary categories emerge within KD methods: feature-based, focusing on intermediate layers’ features, and logits-based, targeting the final layer’s logits. This paper introduces a novel perspective by leveraging diverse knowledge sources within a unified KD framework. Specifically, we aggregate features from intermediate layers into a comprehensive representation, effectively gathering semantic information from different stages and scales. Subsequently, we predict the distribution parameters from this representation. These steps transform knowledge from the intermediate layers into corresponding distributive forms, thereby allowing for knowledge distillation through a unified distribution constraint at different stages of the network, ensuring the comprehensiveness and coherence of knowledge transfer. Numerous experiments were conducted to validate the effectiveness of the proposed method.

1 Introduction

Despite the considerable efficacy of large-scale models [1, 4, 18, 25] in a broad spectrum of tasks, their substantial size and computational demands present significant challenges. Knowledge Distillation, a widely employed network compression technique, continues to attract substantial attention. The primary objective of knowledge distillation is to facilitate knowledge transfer from a sophisticated, high-capacity teacher network to a more streamlined and efficient student network. Mainstream knowledge distillation methods are primarily classified into two categories: 1) logits-based [3, 11, 30, 34] and 2) feature-based [13, 14, 20]. Logits-based distillation focuses on transferring the teacher’s output classification probability to the student network. This methodology focuses on capturing the class probability distribution, which serves as distribution-level constraints. Conversely, feature-based approaches seek to align the individual feature values or activations at each spatial location between the teacher and student networks. This process can be viewed as imposing pixel-level constraints.

 Corresponding authors.

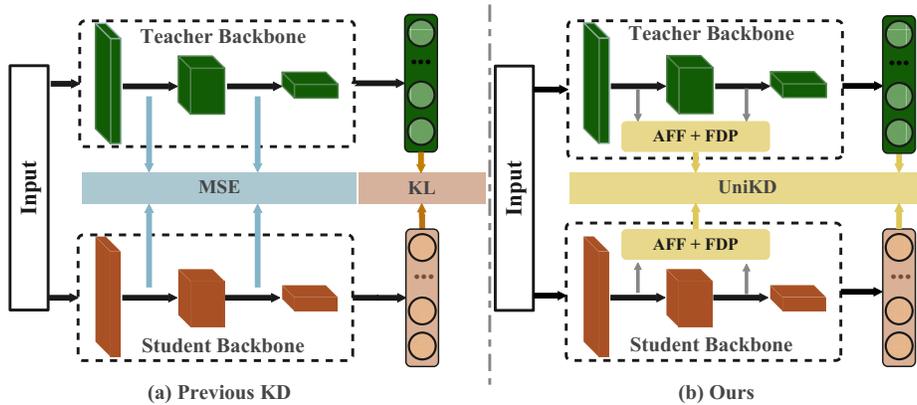


Fig. 1: Comparison of Methodologies in Feature-based knowledge distillation (MSE), Logits-based knowledge distillation (KL), and the Proposed UniKD. UniKD achieves unified knowledge distillation across different network layers, considering information at various levels while circumventing the incoherence from direct combinations.

A critical review of current knowledge distillation methods reveals that these approaches typically center on either a singular type of knowledge [2, 6, 9, 20, 24, 31, 33, 34] or a direct hybridization of two knowledge types [8, 22], overlooking the inconsistencies in knowledge across different layers. This paper aims to facilitate knowledge transfer from the teacher to the student network across various layers using a unified constraint. We will explain the proposed method using three step-by-step problems. First, **why conduct knowledge distillation at different layers?** On the one hand, considering feature capture at different backbone stages is crucial. Shallow layers encode robust local features, bolstering the network’s noise and interference resistance, while deep layers capture abstract semantic information, enhancing adaptability to novel data. On the other hand, the final layer’s logits, which directly relate to the task’s decision-making, contain rich information. Thus, the integration of different stages features and the utilization of logits’ knowledge are necessary. Second, **why unify different types of knowledge distillation methods?** Continuous-valued features are compared in the continuous space using Mean Square Error (MSE), while logits consider the shape of the overall distribution via Kullback-Leibler (KL) divergence rather than just the difference in individual values. These two distillation methods have fundamentally different optimization objectives, which lead to varying effects on the gradients and update rules of the network parameters. Attempting to have the student network receive different types of supervision from the teacher in a single distillation process can result in unclear optimization objectives, making it difficult to achieve the optimal solution. Third, **why unify different knowledge into logits?** Features contain a large amount of information, and each element includes information on the input data. However, this information can be redundant or only indirectly related to the final task. Ensuring

ing strict pixel-level feature alignment between the teacher and student networks can impose overly rigid constraints and result in sub-optimal solutions [21]. In contrast, each dimension of logits directly corresponds to a specific category, and its value reflects the network’s prediction information for that category. Logits carry a higher information density and are directly related to the final decision process.

Building upon the previous analysis, we introduce the **Unified Knowledge Distillation (UniKD)**, which ensures harmonized knowledge transfer across different layers, thus facilitating an integrated and coherent transfer from the teacher to the student network. Specifically, the Adaptive Features Fusion (AFF) module is employed to derive an integrated representation from intermediate layers. AFF can retain critical information across different scales while simplifying the computation process. Following this, we explore the knowledge distribution at intermediate layers by utilizing the Feature Distribution Prediction (FDP) module to estimate distribution parameters. This way, knowledge distillation within the intermediate layers can be conducted by implementing distribution-level constraints. Upon completing these steps, UniKD enables a uniform and coherent knowledge distillation process through the network’s various layers. In summary, the main contributions of this work include:

- We comprehensively analyze the challenges in current knowledge distillation methods and introduce UniKD. This framework ensures uniform knowledge distillation across various network layers, thereby ensuring the integrity and coherence of the knowledge transfer.
- The AFF module aggregates features from intermediate layers, which retains multi-scale information while avoiding unnecessary information transfer and further utilizes the FDP to harmonize the knowledge across different layers.
- The superiority of our method is demonstrated by extensive comparative experiments with teacher-student pairs across various datasets for different tasks.

2 Related Work

2.1 Logits-based Knowledge Distillation

Knowledge Distillation, first introduced in [9], aims to distill knowledge from a large, cumbersome model into a smaller, faster one without significant generalization performance loss. It encourages the student model to mimic the teacher’s outputs by minimizing the divergence between the teacher’s and student’s predictions. Following this, a significant amount of work has focused on the research of logits-based knowledge distillation. DML [33] introduces a mutual-learning paradigm that enables an ensemble of students to learn collaboratively and teach each other throughout the training process. WSL [6] explores the reasons behind the functionality of soft labels by comparing the bias-variance of direct training without distillation and training with distillation. DKD [34] and USKD [30] decouple the predicted class distributed to further extract information from the

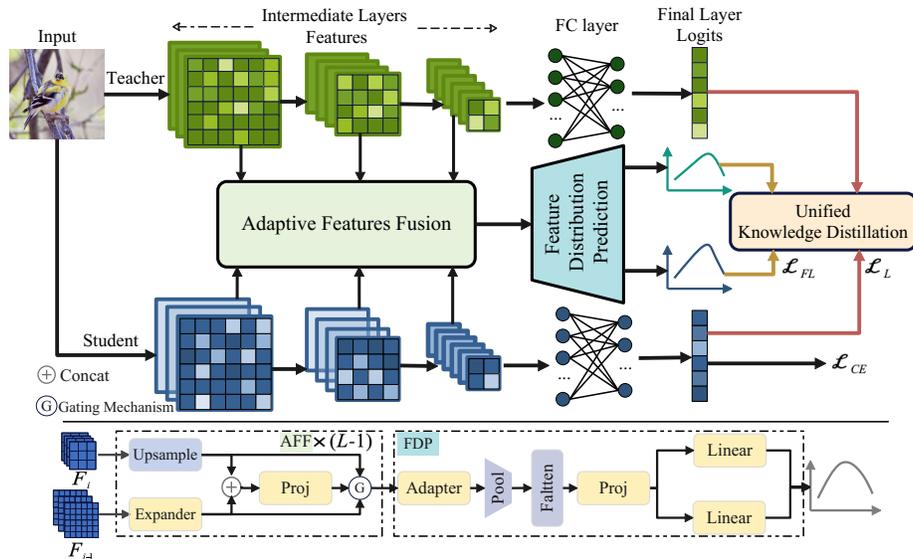


Fig. 2: The overall framework of the proposed UniKD. Our approach primarily comprises two modules. Initially, the features from the intermediate layers are aggregated using the stacked Adaptive Features Fusion (AFF) modules, followed by the derivation of the corresponding feature distribution through the Feature Distribution Prediction module (FDP). Consequently, our **Unified Knowledge Distillation (UniKD)** facilitates knowledge distillation in both the intermediate layer’s features distribution and the final layer’s probability distribution under a unified constraint.

logits. MLKD [11] aligns logits not only at the instance level but also at the batch and class levels. [32] achieves self-distillation by converting the final logits and the features from intermediate layers into logits using a fully connected layer.

2.2 Feature-based Knowledge Distillation

Unlike distillation on logits, feature-based methods were first introduced in FitNets [20], which transfers knowledge from intermediate features. Inspired by this, various methods have been proposed to match the features. AT [31] focuses on transferring the attention of feature maps to the students. OFD [7] operates by collaboratively considering the adaptive layer, distillation feature, and distance function. CRD [24] aims to maximize the lower bound of mutual information between the teacher and student by utilizing the approach of contrastive learning. KR [2] enhances distillation performance through the review of existing knowledge. Additionally, some methods [17, 26] concentrate on distilling the correlation to convey the teacher’s knowledge.

Previous KD approaches focused on only one of these types of knowledge (features or logits). Some studies, despite incorporating two knowledge types [10,

22], need to address the distinctions between features and logits. [27] converts features into corresponding logits using the head, disregarding the differences in knowledge across different layers. This paper proposes UniKD, a solution that enables consistent knowledge transfer across different layers through unified knowledge.

3 Unified Knowledge Distillation

This section begins with a review of existing distillation methods, analyzing their limitations. Subsequently, we present our method, detailing the specific modules it comprises. Figure 2 provides a comprehensive overview of this framework.

3.1 Preliminaries

Logits-based KD was first proposed in [9], where the student learns the logits knowledge from the teacher through KL divergence. In contrast to direct supervised learning through ground truth, logits-based KD using soft labels can elicit more ‘dark knowledge’ [9], thereby enhancing the performance of the student network without altering its architecture. The logits distillation losses are represented as follows:

$$\mathcal{L}_L = KL(p^t || p^s) = \sum_{j=1}^C p_j^t(\tau) \log\left(\frac{p_j^t(\tau)}{p_j^s(\tau)}\right), \quad (1)$$

P_j denotes the class probability derived from the logits $z \in \mathbb{R}^C$ after undergoing Softmax. τ is the temperature scaling hyper-parameter which enables the production of different probability distributions:

$$P_j(\tau) = \frac{e^{z_j/\tau}}{\sum_{c=1}^C e^{z_c/\tau}}. \quad (2)$$

Feature-based KD transfers the knowledge of intermediate layers features from the teacher to the student. Generally, we can formulate such distillation methods as:

$$\mathcal{L}_F = \frac{1}{2} \sum_{i=1}^n \|\mathcal{F}_i^t - w(\mathcal{F}_i^s)\|^2, \quad (3)$$

where \mathcal{F}_i^t and \mathcal{F}_i^s are the corresponding features of the teacher and student at i -th layer. $w(\cdot)$ is a mapping function that aligns the dimensions of features from the student to the teacher.

From the above distillation process, it can be seen that logits-based knowledge distillation focuses on constraining the overall knowledge distribution. In contrast, feature-based knowledge distillation imposes pixel-level constraints on the student network. These two methods have distinct optimization objectives

during the distillation process, and directly integrating them can lead to sub-optimal solutions. In this study, we aim to achieve comprehensive knowledge transfer by conducting knowledge distillation on the intermediate layer features and the final layer logits. To address this issue, we introduce UniKD, a meticulously designed framework that adjusts the intermediate layer features for a thorough and consistent knowledge transfer, using the same constraints employed for the final layer’s logits.

3.2 Unified Knowledge Transfer

In this paper, we convert the intermediate layers features at the pixel level into a distribution form of type logits, facilitating the distillation of knowledge at the distribution level from the intermediate layers. This primarily considers the inherent differences in representation between the teacher and student networks due to the gap in model size, making it difficult for the student network to understand excessive pixel-level constraints. In contrast, logits provide a higher-level abstract representation that captures the overall characteristics of the input data rather than just the local pixel-level details. Therefore, by aligning the logits distributions, the student network can better learn the knowledge from the teacher network, enhancing its performance.

Adaptive Features Fusion Module Directly transforming features from each layer to their corresponding distributions is computationally expensive and challenging. Therefore, we first fuse features from different layers to obtain a feature that incorporates multi-scale information. This approach ensures that the resultant feature representation captures both fine-grained details from lower layers and high-level semantics from upper layers. By integrating multi-scale information, the model can effectively process diverse features with reduced computational overhead. We propose the Adaptive Features Fusion (AFF) module, which aggregates feature information from different layers. We denote the output of each intermediate layer’s last residual blocks as $\{\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_L\}$. For \mathcal{F}_{i-1} , its channels are increased through an expander and then concatenated with the result after upsampling of \mathcal{F}_i . Subsequently, both information is fused through a gate mechanism, and the integrated intermediate representation \mathcal{F}_R are output through an top-down method. Formally,

$$\begin{aligned}\mathcal{F}_R &= \mathbf{g} \odot \mathbf{E}(\mathcal{F}_{i-1}) + (1 - \mathbf{g}) \odot \mathbf{Up}(\mathcal{F}_i), \\ \mathbf{g} &= f(\mathbf{E}(\mathcal{F}_{i-1}) \oplus \mathbf{Up}(\mathcal{F}_i)),\end{aligned}\tag{4}$$

where $i = 2, 3, \dots, L$, \mathbf{E} represents a 1×1 convolution to adapt to the number of channels at the $(i - 1)$ -th layer, \mathbf{Up} denotes an upsampling operation to match the resolution of the $(i - 1)$ -th layer, f stands for a gating function implemented through a 3×3 convolution, \odot indicates point-wise multiplication, and \oplus represents the concatenation operator. By adaptively determining the importance of adjacent layer features through g , this approach ensures that the resultant \mathcal{F}_R retains the relatively significant components across different layers while eliminating redundant information.

Feature Distribution Prediction Module Feature-based Knowledge distillation inherently steers the student’s intermediate feature towards convergence with the teacher’s, suggesting a shared underlying distribution for student and teacher features. In this work, we align the features of the teacher and student networks at the intermediate layers to a multivariate Gaussian distribution. The Gaussian assumption allows us to model the distribution of features with fewer parameters, capturing the essential characteristics of the data while discarding redundant information. Thus, this approach not only facilitates computation by leveraging the properties of Gaussian distributions, which simplify mathematical operations and statistical analysis, but has also been empirically proven to be effective in improving the performance of the student network.

For the multivariate Gaussian distributions corresponding to the features output by the student network, $q_\phi = \mathcal{N}(\mu_s, \Sigma_s)$, and those output by the teacher network, $q_\theta = \mathcal{N}(\mu_t, \Sigma_t)$, the KL divergence between them can be determined with an explicit analytical solution. μ_s and μ_t represent the mean vectors of the distributions corresponding to student and teacher features, respectively, while Σ_s and Σ_t denote the corresponding covariance matrices. The analytical solution as follows:

$$\mathcal{L}_{FL}(q_\phi \| p_\theta) = \frac{1}{2} \left(\text{tr}(\Sigma_t^{-1} \Sigma_s) + (\mu_t - \mu_s)^\top \Sigma_t^{-1} (\mu_t - \mu_s) - k + \ln \left(\frac{|\Sigma_t|}{|\Sigma_s|} \right) \right), \quad (5)$$

where $\text{tr}(\cdot)$ denotes the trace of a matrix, $|\cdot|$ represents the determinant of the matrix, and k is the dimensionality of the features. A detailed proof of the process for obtaining the explicit analytical solutions is provided in the Supplementary Materials.

Eq. 5 involves the computation of the inverse matrix, the determinant, as well as the trace of the matrix, which is not only time-consuming but also numerically unstable, particularly in high-dimensional spaces. Here, we simplify the corresponding covariance matrices to diagonal matrices, specifically $\Sigma_s = \text{diag}(\sigma_{s1}^2, \sigma_{s2}^2, \dots, \sigma_{sk}^2)$ and $\Sigma_t = \text{diag}(\sigma_{t1}^2, \sigma_{t2}^2, \dots, \sigma_{tk}^2)$. Consequently, Eq. 5 can be simplified as:

$$\mathcal{L}_{FL}(q_\phi \| p_\theta) = \frac{1}{2} \sum_{i=1}^k \left(\frac{\sigma_{si}^2}{\sigma_{ti}^2} + \frac{(\mu_{ti} - \mu_{si})^2}{\sigma_{ti}^2} - 1 + \ln \left(\frac{\sigma_{ti}^2}{\sigma_{si}^2} \right) \right), \quad (6)$$

where σ_{si}^2 and σ_{ti}^2 represent the variance of the student and teacher features in the i th dimension, respectively, while μ_{si} and μ_{ti} denote the corresponding means.

Based on the aforementioned analysis, we only need to predict the corresponding mean values μ_s , μ_t and variances σ_s^2 , σ_t^2 of the student and teacher network features. Regarding the \mathcal{F}_R obtained through the Attention Feature Fusion (AFF) method, we deploy a Feature Distribution Prediction (FDP) module with shared parameters to predict the parameters of the multivariate Gaussian distributions corresponding to the fused features of both the student \mathcal{F}_R^s and the teacher \mathcal{F}_R^t . Specifically, for \mathcal{F}_R^s and \mathcal{F}_R^t , we apply pooling and flattening

Table 1: Results on the CIFAR-100 validation. Teachers and students are in homogeneous architectures. Top-1 accuracy is adopted as the evaluation metric and original performance is reported respectively. NKD* denotes NKD [23] with DKD [34].

Model	Teacher Student	ResNet56	ResNet110	ResNet32×4	WRN-40-2	WRN-40-2	VGG13
		ResNet20	ResNet32	ResNet8 ×4	WRN-16-2	WRN-40-1	VGG8
Baseline	Teacher	72.34	74.31	79.42	75.61	75.61	74.64
	Student	69.06	71.14	72.50	73.26	72.98	70.36
Feature	FitNet [20]	69.21	71.06	73.50	73.58	72.24	71.02
	CRD [24]	71.16	73.48	75.51	75.48	74.14	73.94
	AT [31]	70.55	72.31	73.44	74.08	72.77	71.43
	KR [2]	71.89	73.89	75.63	76.12	75.09	74.84
Logits	KD [9]	70.66	73.08	73.33	74.92	73.54	71.98
	DKD [34]	71.97	74.11	76.32	76.24	74.81	74.68
	NKD* [23]	72.32	74.29	77.01	76.39	74.89	74.81
Ours		73.01	75.36	78.18	77.49	75.35	75.76

operations, followed by a projection to align the dimensions with the task label, and then predict the mean values μ_s, μ_t and variances σ_s^2, σ_t^2 . This process implements the distillation of feature knowledge as formulated in Eq. 6.

3.3 Overall Objective Loss

We employ the AFF method to merge features from various layers, subsequently converting them into their respective distributions using the FDP. This method effectively standardizes the process of distilling knowledge from both the intermediate layers and the final layer logits. Our approach ensures a consistent and efficient knowledge transfer by utilizing KL divergence, as detailed in Eq. 6 and 1 respectively. This not only enhances the learning capacity of the student model by leveraging the rich feature information from the teacher model but also simplifies the overall distillation process. In summary, by integrating Eq. 1 and Eq. 6, the total training objective for the student model is:

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \alpha\mathcal{L}_{FL} + \beta\mathcal{L}_L, \quad (7)$$

where \mathcal{L}_{CE} is the standard task training loss for the student, α, β is corresponding weights. From \mathcal{L}_{total} , it can be observed that, apart from the standard supervision loss of the student network, all other losses are implemented through KL divergence. Consequently, this approach permits knowledge distillation within a comparatively unified paradigm.

4 Experiments

This section first provides a brief introduction to our primary experimental settings. Subsequently, we evaluate the performance of our method and compare

Table 2: Results on the CIFAR-100 validation. Teachers and students are in heterogeneous architectures. Top-1 accuracy is adopted as the evaluation metric and original performance is reported respectively. NKD* denotes NKD [23] with DKD [34].

Model	Teacher Student	ResNet32×4 ShuffleNet-V1	WRN-40-2 ShuffleNet-V1	VGG13 MobileNet-V2	ResNet50 MobileNet-V2	ResNet32×4 ShuffleNet-V2
Baseline	Teacher	79.42	75.61	74.64	79.34	79.42
	Student	70.50	70.50	64.60	64.60	71.82
Feature	FitNet [20]	73.54	73.73	64.14	63.16	73.54
	CRD [24]	75.11	72.21	69.73	69.11	75.65
	AT [31]	71.73	73.32	69.40	68.58	73.40
	KR [2]	77.45	77.14	70.37	69.89	77.78
Logits	KD [9]	74.07	74.83	67.37	67.35	74.45
	DKD [34]	76.45	76.70	69.71	70.35	77.07
	NKD* [23]	77.11	77.12	69.98	70.45	77.37
Ours		77.89	78.10	71.02	72.10	79.20

it with previous KD methods. Ultimately, we conduct ablation experiments and visual demonstrations to illustrate the efficacy of our method.

4.1 Experimental Settings

Datasets. (1) CIFAR-100 [12] comprises a well-known image classification dataset, with 60,000 32×32 images in total (50,000 for training and 10,000 for validation) from 100 categories; (2) ImageNet [5] is a large-scale classification dataset that consists of 1,000 classes, with nearly 1.3 million training images and 50,000 images for validation; (3) MS-COCO [16], a challenging object detection dataset, comprises over 200,000 labeled images across 80 categories, including 118,000 training images and 5,000 validation images.

Implementation Details. We perform distillation experiments on various neural network architectures, including within homogeneous architecture for teacher-student pairs (e.g., ResNet110 and ResNet32) and heterogeneous architectures (e.g., VGG13 and MobileNet-V2). For CIFAR-100, we used 1 NVIDIA 3090 GPU, with a batch size of 64 and a learning rate of 0.05. For ImageNet, we utilized 4 NVIDIA A100 GPUs, with a batch size of 512 and a learning rate set to 0.2. Finally, we also used 4 GPUs for related experiments on MS-COCO, where the batch size was set to 8, and the learning rate was set to 0.01.

4.2 Main Results

Results on CIFAR-100. We evaluate our method on CIFAR-100 and compare it with previous methods. Table 1 displays the distillation results between the teacher-student pair with a homogeneous architecture. The results indicate that our method outperforms others in facilitating a more efficient and effective transfer of knowledge from the teacher to the student model. This advantage is evident

Table 3: Top-1 and Top-5 accuracy (%) of student networks on ImageNet validation set.

ResNet34 as the teacher, ResNet18 as the student									
	Teacher	Student	AT [31]	OFD [7]	CRD [24]	KR [2]	KD [9]	DKD [34]	Ours
Top-1	73.71	69.75	70.69	70.81	71.17	71.61	70.66	71.70	72.83
Top-5	91.42	89.07	90.01	89.98	90.51	90.51	89.88	90.41	91.38
ResNet50 as the teacher, MobileNetV1 as the student									
Top-1	76.16	68.87	69.56	71.25	72.56	72.56	68.58	72.05	73.25
Top-5	92.86	88.76	89.33	90.34	91.00	91.00	88.98	91.05	91.72

across all the different pairs, highlighting the robustness and versatility of our approach.

In addition to the experiments on similar architecture pairs, we have also conducted comparative experiments on teacher-student pairs that display significant structural differences. The results of these experiments are presented in Table 2. Similar to the observations made in the homogeneous architecture scenario, we noticed a substantial performance improvement. This result suggests that our method is not only effective in homogeneous architecture scenarios but also in heterogeneous architecture. This further underscores the adaptability and effectiveness of our approach in a wide range of scenarios.

Results on ImageNet. In order to showcase the wide-ranging effectiveness of our methodology, we engaged in a series of comparative analyses using a notably intricate dataset, ImageNet. We report both Top-1 and Top-5 accuracy in Table 3. We also conducted comparative experiments on homogeneous architectures (ResNet34 and ResNet18) and heterogeneous architectures (ResNet50 and MobileNet). The results in the table show that our method has improved by 0.93% and 0.24% compared to the previous SOTA methods on Top-1 accuracy.

Results on MS-COCO. We extensively experimented with object detection tasks. Object detection is more complex than classification as it requires not only categorizing objects but also accurately determining their bounding boxes. We employed the widely recognized MS-COCO 2017 dataset for method evaluation for these experiments. Detectron2 [29], a popular open-source framework, served as our benchmark. We adopted Faster-RCNN [19] with FPN [15] as the backbone architecture. Performance was assessed on the MS-COCO 2017 validation set using metrics such as AP , AP_{50} , and AP_{75} , with results presented in Table 4. The results indicate that our method holds a distinct advantage over approaches utilizing a single type of knowledge. Compared with straightforward hybrids of two knowledge types, UnikD consistently outperforms in most scenarios. However, employing constraints at the distribution level might yield less significant improvements for dense prediction tasks such as object detection than classification tasks. This may be primarily because our method focuses on constraints at

Table 4: Results on MS-COCO. We take Faster-RCNN [19] with FPN [15] as the backbone, and use AP on different settings to evaluate results.

		ResNet101 & ResNet18			ResNet101& ResNet50			ResNet50 & MobileNet2		
		AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75
Method	Teacher	42.04	62.48	45.88	42.04	62.48	45.88	40.22	61.02	43.81
	Student	33.26	53.61	35.26	37.93	58.84	41.05	29.47	48.87	30.90
Feature	FitNet [20]	34.13	54.16	36.71	38.76	59.62	41.80	30.20	49.80	31.69
	FGFI [28]	35.44	55.51	38.17	39.44	60.27	43.04	31.16	50.68	32.92
	KR [2]	36.75	56.72	34.00	40.36	60.97	44.08	33.71	53.15	36.13
Logits	KD [9]	33.97	54.66	36.62	38.35	59.41	41.71	30.13	50.28	31.35
	DKD [34]	35.05	56.60	37.54	39.25	60.90	42.73	32.34	53.77	34.01
	MLKD [11]	36.03	57.28	38.51	40.15	61.67	44.57	33.83	54.01	35.22
Hybrid	FitNet+KD	34.32	54.52	37.01	39.03	59.97	42.23	30.18	50.27	31.59
	KR+DKD	37.01	57.53	39.85	40.65	61.51	44.44	34.35	54.89	36.61
Ours		37.24	57.86	39.79	40.76	61.72	44.61	34.62	55.14	36.59

the overall distribution level, making such dense prediction tasks less noticeable.

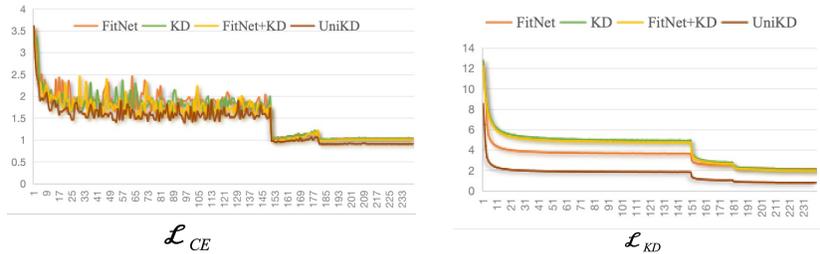
4.3 Analyses

Comparison with Hybrid KD Methods. In order to better illustrate the core element of our method, which is the ability to coordinate different types of knowledge at different layers of a network and thus ensure comprehensive and coherent knowledge transfer, we compared our approach with directly hybrid two types of KD methods. We evaluated four different logits-based and feature-based simple hybrid methods and FLG [10], which incorporates distillation via logits, features, and gradients but does not unify varying knowledge types. The results in Table 5 indicate that combining the two types of distillation methods can significantly enhance the performance of the student network, underscoring the importance of integrating knowledge from different layers. However, compared to the proposed UniKD, which conducts comprehensive and continuous knowledge distillation using a unified type of knowledge, the performance gap remains substantial. By representing the knowledge of each layer of the network in a unified manner, our method can mitigate the inconsistency between different types of knowledge and enhance the distillation performance. This further highlights the necessity of unifying different types of knowledge.

Convergence Analysis. Figure 3 illustrates the convergence behaviors of different distillation methods. The left side shows the variation of the task loss \mathcal{L}_{CE} . It is observable that a single type of method is relatively stable, yet the outcome is sub-optimal. Although a direct hybrid of the two methods results in a loss reduction, the training process exhibits instability, particularly in the early layers of network training. In contrast, our approach ensures stable training and the

Table 5: Comparing the distillation methods directly combining logits and features, experiments were conducted in six different teacher-student pairs in CIFAR100.

Model	Teacher Student	ResNet56	ResNet110	ResNet32×4	WRN-40-2	WRN-40-2	VGG13
		ResNet20	ResNet32	ResNet8 ×4	WRN-16-2	WRN-40-1	VGG8
Baseline	Teacher	72.34	74.31	79.42	75.61	75.61	74.64
	Student	69.06	71.14	72.50	73.26	71.98	70.36
Hybrid	KD [9]+FitNet [20]	70.09	73.08	75.19	74.32	72.68	72.61
	KD [9] + CRD [24]	71.63	73.75	75.46	75.64	74.38	74.29
	DKD [34] + KR [2]	71.98	74.23	76.45	76.19	75.52	75.33
	MLKD [11] + KR [2]	72.83	74.52	78.01	77.54	76.21	75.69
	FLG [10]	71.55	73.48	76.66	75.74	74.29	74.28
Ours		73.01	75.36	78.18	77.49	76.35	75.76

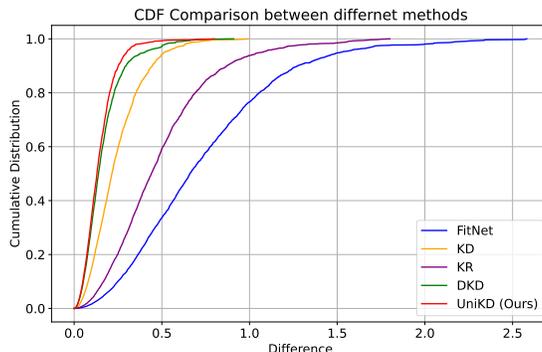
**Fig. 3:** A comparative analysis of the convergence of test task loss \mathcal{L}_{CE} and knowledge distillation loss \mathcal{L}_{KD} across various types of distillation methods. All experiments were conducted on the CIFAR100 dataset utilizing a ResNet32×4-ResNet8×4 teacher-student pair architecture.

most favorable results. Additionally, the right side illustrates the variations in distillation loss during the training process. It can be observed that the proposed UniKD allows the student network to more closely approximate the teacher network, thereby learning more of the teacher’s knowledge.

Ablation Study. To validate the effectiveness of each module in our method, we analyzed their contributions individually. Table 6 shows distillation results for both homogeneous (i.e., ResNet110 and ResNet32) and heterogeneous architectures (i.e., WRN-40-2 and ShuffleNet-V1), underlining our method’s efficacy. The results confirm that our carefully crafted modules significantly enhance the student model’s performance. In our initial approach, we apply the Feature Distribution Processor (FDP) to the output features of the last intermediate layer to unify different types of knowledge, significantly enhancing the performance of the student model. The results indicate that our method achieves substantial improvements. However, due to the lack of integration of information from different layers, the utilization of the teacher’s knowledge still needs to be improved. Therefore, we incorporate the Adaptive Feature Fusion (AFF) module to inte-

Table 6: Ablation study for different modules in UniKD.

Module	ResNet110 & ResNet32			WRN-40-2 & SN-V1		
FDP	✗	✓	✓	✗	✓	✓
AFF	✗	✗	✓	✗	✗	✓
Acc.	73.08	74.42	75.36	75.21	77.10	78.10

**Fig. 4:** CDF curve of the differences in logits output by teacher and student in correspondence with various methods.

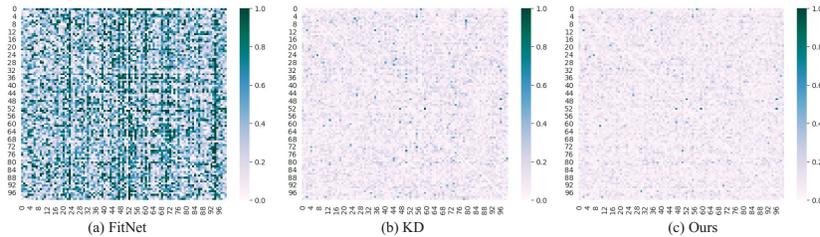
grate intermediate features from different layers, achieving a comprehensive and unified knowledge distillation. The results demonstrate the effectiveness of the AFF module.

Comparison with Teacher Model. The results in Table 7 show that while most methods improve upon the original student model, they often do not outperform the teacher model. However, with our UniKD, the distilled student model not only significantly outperforms its original version but also exceeds the teacher model in most cases, achieving an average improvement of 0.57% across six teacher-student pairs. The results highlight UniKD’s efficacy in capturing essential knowledge from the teacher network and minimizing redundant information transfer, leading to a lightweight student model that surpasses the more cumbersome teacher model.

Visualizations. We analyzed the Cumulative Distribution Function (CDF) curves of different methods, where the x -axis indicates the difference magnitude between teacher and student model outputs, and the y -axis shows the proportion of differences not exceeding each x -axis value. As Figure 4 demonstrates, our method’s CDF curve swiftly rises to 1, which suggests a high degree of similarity between the outputs of the student and teacher models, implying effective learning and enhanced performance of the student model.

Table 7: We calculated the average metrics the deviations from both the student and the teacher of different teacher-student pairs in Table 1.

	Feature				Logits			Hybrid	Ours
	FitNet	CRD	AT	KR	KD	DKD	NKD	DKD+KR	UniKD
Avg	71.77	71.73	72.43	74.58	73.09	74.69	75.09	74.95	75.89
Δ_{stu}	0.39	0.35	1.05	3.2	1.71	3.31	3.40	3.57	4.31
Δ_{tea}	-3.55	-3.59	-2.89	-0.74	-2.23	-0.63	-0.37	-0.37	0.54

**Fig. 5:** Difference of correlation matrices of student and teacher logits.

We also visualize the differences in the correlation matrices between student and teacher logits, as illustrated in Figure 5. UniKD enables the student to produce logits that are more similar to those of the teacher compared to FitNet and KD, thereby achieving superior distillation performance.

5 Conclusion

We analyzed that different network layers contain different types of knowledge, thus necessitating knowledge distillation at various layers. However, using different types of constraint methods can lead to unclear optimization objectives. Therefore, it is essential to unify the different types of knowledge. Based on this, We present a novel KD framework harmonizing various knowledge sources. It adapts intermediate knowledge to align with final-layer logits, providing a uniform knowledge form and enabling efficient KD throughout the network stages. We tested it extensively with various teacher-student pairs on different tasks. Our method outdid both logits and feature-based KD. It successfully acquired sufficient knowledge from the teacher network in various analytical experiments.

Acknowledgements

This work was supported by the National Key R & D Program of China (2022ZD0161800), and the National Natural Science Foundation of China (62271203).

References

1. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. *Advances in neural information processing systems* **33**, 1877–1901 (2020)
2. Chen, P., Liu, S., Zhao, H., Jia, J.: Distilling knowledge via knowledge review. In: *Proceedings of the IEEE/CVF Conference on computer Vision and Pattern Recognition*. pp. 5008–5017 (2021)
3. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 4794–4802 (2019)
4. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., et al.: Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311* (2022)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: *2009 IEEE conference on computer vision and pattern recognition*. pp. 248–255. *Ieee* (2009)
6. Helong, Z., Liangchen, S., Jiajie, C., Ye, Z., Guoli, W., Junsong, Y., Zhang, Q.: Rethinking soft labels for knowledge distillation: a bias-variance tradeoff perspective. In: *International Conference on Learning Representations (ICLR)* (2021)
7. Heo, B., Kim, J., Yun, S., Park, H., Kwak, N., Choi, J.Y.: A comprehensive overhaul of feature distillation. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 1921–1930 (2019)
8. Heo, B., Lee, M., Yun, S., Choi, J.Y.: Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. pp. 3779–3787. AAAI Press (2019). <https://doi.org/10.1609/aaai.v33i01.33013779>, <https://doi.org/10.1609/aaai.v33i01.33013779>
9. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015)
10. Hsu, Y.C., Smith, J., Shen, Y., Kira, Z., Jin, H.: A closer look at knowledge distillation with features, logits, and gradients. *arXiv preprint arXiv:2203.10163* (2022)
11. Jin, Y., Wang, J., Lin, D.: Multi-level logit distillation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 24276–24285 (2023)
12. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
13. Li, J., Guo, Z., Li, H., Han, S., Baek, J.w., Yang, M., Yang, R., Suh, S.: Rethinking feature-based knowledge distillation for face recognition. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 20156–20165 (2023)
14. Lin, S., Xie, H., Wang, B., Yu, K., Chang, X., Liang, X., Wang, G.: Knowledge distillation via the target-aware transformer. In: *Proceedings of the IEEE/CVF Conference on computer Vision and Pattern Recognition*. pp. 10915–10924 (2022)
15. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 2117–2125 (2017)

16. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
17. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3967–3976 (2019)
18. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* **21**(1), 5485–5551 (2020)
19. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* **28** (2015)
20. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
21. Shu, C., Liu, Y., Gao, J., Yan, Z., Shen, C.: Channel-wise knowledge distillation for dense prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5311–5320 (2021)
22. Song, J., Chen, Y., Ye, J., Song, M.: Spot-adaptive knowledge distillation. *IEEE Trans. Image Process.* **31**, 3359–3370 (2022)
23. Sun, S., Ren, W., Li, J., Wang, R., Cao, X.: Logit standardization in knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15731–15740 (2024)
24. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation. arXiv preprint arXiv:1910.10699 (2019)
25. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
26. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1365–1374 (2019)
27. Wang, J., Chen, Y., Zheng, Z., Li, X., Cheng, M.M., Hou, Q.: Crosskd: Cross-head knowledge distillation for dense object detection. arXiv preprint arXiv:2306.11369 (2023)
28. Wang, T., Yuan, L., Zhang, X., Feng, J.: Distilling object detectors with fine-grained feature imitation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4933–4942 (2019)
29. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. <https://github.com/facebookresearch/detectron2> (2019)
30. Yang, Z., Zeng, A., Li, Z., Zhang, T., Yuan, C., Li, Y.: From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. arXiv preprint arXiv:2303.13005 (2023)
31. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. arXiv preprint arXiv:1612.03928 (2016)
32. Zhang, L., Song, J., Gao, A., Chen, J., Bao, C., Ma, K.: Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2019)
33. Zhang, Y., Xiang, T., Hospedales, T.M., Lu, H.: Deep mutual learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)

34. Zhao, B., Cui, Q., Song, R., Qiu, Y., Liang, J.: Decoupled knowledge distillation. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 11953–11962 (2022)