Learning From Human Educational Wisdom: A Student-Centered Knowledge Distillation Method

Shunzhi Yang, Jinfeng Yang, MengChu Zhou, *Fellow, IEEE*, Zhenhua Huang, Wei-Shi Zheng, *Member, IEEE*, Xiong Yang, and Jin Ren

***Abstract*—Existing studies on knowledge distillation typically focus on teacher-centered methods, in which the teacher network is trained according to its own standards before transferring the learned knowledge to a student one. However, due to differences in network structure between the teacher and the student, the knowledge learned by the former may not be desired by the latter. Inspired by human educational wisdom, this paper proposes a Student-Centered Distillation (SCD) method that enables the teacher network to adjust its knowledge transfer according to the student network's needs. We implemented SCD based on various human educational wisdom, e.g., the teacher network identified and learned the knowledge desired by the student network on the validation set, and then transferred it to the latter through the training set. To address the problems of current deficiency knowledge, hard sample learning and knowledge forgetting faced by a student network in the learning process, we introduce and improve Proportional-Integral-Derivative (PID) algorithms from automation fields to make them effective in identifying the current knowledge required by the student network. Furthermore, we propose a curriculum learning-based fuzzy strategy and apply it to the proposed PID control algorithm, such that the student network in SCD can actively pay attention to the learning of challenging samples after with certain knowledge. The overall performance of SCD is verified in multiple tasks by comparing it with state-of-the-art ones. Experimental results show that our student-centered distillation method outperforms existing teacher-centered ones.**

***Index Terms*—Curriculum learning, fuzzy PID, human educational wisdom, knowledge distillation, student-centered.**

Manuscript received 27 March 2023; revised 1 December 2023; accepted 13 January 2024. Date of publication 16 January 2024; date of current version 7 May 2024. This work was supported in part by the National Natural Science Foundation of China under Grants 62076166, 62172166, and 61772366, in part by Guangdong Basic and Applied Basic Research under Grant 2022A1515011380, and in part by Shenzhen Science and Technology Program under Grant 20231126112732001. Recommended for acceptance by V. Morariu. *(Corresponding author: Zhenhua Huang.)*

Shunzhi Yang, Jinfeng Yang, Xiong Yang, and Jin Ren are with the Institute of Applied Artificial Intelligence of the Guangdong-Hong Kong-Macao Greater Bay Area, Shenzhen Polytechnic University, Shenzhen 518055, China (e-mail: yangshunzhi1994@gmail.com; jfyang@szpt.edu.cn; 2018021011@cauc.edu.cn; renjin666@szpu.edu.cn).

MengChu Zhou is with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: zhou@njit.edu).

Zhenhua Huang is with the School of Computer Science, South China Normal University, Guangzhou 510631, China (e-mail: jukiehuang@163.com).

Wei-Shi Zheng is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510275, China (e-mail: wszheng@ieee.org).

This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2024.3354928, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2024.3354928

I. Introduction

KNOWLEDGE distillation (KD) [1] is a machine learning technique similar to human educational wisdom, i.e., it uses a powerful teacher network to guide a weaker student network to learn knowledge. In this paper, the teacher and student networks in KD are abbreviated as teacher and student, respectively, for convenience. KD is a model compression method to develop efficient networks for resource-constrained devices [2], [3], [4]. KD methods involving two distinct networks can be classified into online and offline distillations [3]. Both the online and offline distillation studies have focused on mining knowledge from a teacher and transferring it to a student.

However, existing KD methods typically adopt a teacher-centered approach. In this methodology, the teacher acquires and transfers knowledge based on its own standards, often neglecting the student's specific learning requirements. For example, DML [5] involves two untrained networks learning each other's knowledge independently, without considering each other's needs. FitNets [6] transfers the pre-trained knowledge of a teacher to a student. DML and FitNets are representative online and offline KD methods, respectively. Typically, these methods involve a teacher transmitting its acquired knowledge to a student, without considering the latter's learning situations. Such a way may lead to misaligned knowledge transfer, as a teacher may convey information that is not entirely relevant or beneficial for a student. In contrast, human educational practices emphasize a student-centered strategy [7], [8], where the focus is on aligning teaching content with a student's learning needs. This alignment is crucial because the objective of KD is to develop an efficient student for processing user tasks, rather than enhancing a teacher. Moreover, while teachers are typically more complex and capable of acquiring more knowledge, the primary challenge lies in distilling their knowledge into a form that is accessible and beneficial for a student. Disseminating irrelevant knowledge could lead to inefficient use of a teacher's abilities, potentially reducing the effectiveness of the KD process.

To maximize the efficiency and effectiveness of knowledge transfer in KD, this paper proposes a Student-Centered Distillation (SCD) method, where the teacher aims to learn the knowledge desired by the student. In this paper, the knowledge that a student desires is it lacks. Providing the student with desired knowledge can improve its generalization performance. Different from existing studies, which usually try to make a student imitate the knowledge learned by a complex teacher, both the ideas and the implementation methods of this paper

follow the student-centered human educational wisdom. For example, in human educational activities, teachers use test papers to check students' knowledge mastery before providing targeted instruction. Accordingly, in this paper, a teacher identifies and learns the knowledge desired by a student through the validation set, and then transfers that knowledge to the student on the training set. The validation set only evaluates the student's performance, and does not update its parameters. The validation set contains almost all the knowledge that a student should have to learn, similar to the knowledge that a human student should have to master for admission to a university.

Although all of the knowledge on the validation set should be learned by the student, the teacher is not required to learn and transfer all of it, as some of it may already be mastered by the student. In human educational activities, teachers regularly work on overcoming students' weaker areas of knowledge to deepen their understanding [9]. Thus, in this paper, we follow this teaching experience to enable a teacher to discover and learn a student's relatively weak knowledge. Afterwards, the teacher transfer that knowledge into the student to effectively improve the latter's performance. In SCD, a teacher becomes a facilitator for learning and transferring knowledge that a student does not understand. With the help of a teacher, a student can continuously discover and overcome its relatively weak knowledge. However, the relatively weak knowledge of the student at different epoch is usually different. It requires the teacher not only to discover the weak knowledge of the student in time, but also to prevent the student from experiencing knowledge forgetting. Knowledge forgetting [10] refers to the fact that when a student learns new knowledge, it may forget what it previously learned. For example, at the tth epoch, a student's weakest knowledge is "$\Diamond$", whereas in the $(t+1)$th epoch, it is "$\square$". Unfortunately, the student may forget knowledge "$\Diamond$" after learning knowledge "$\square$". Thus, the student in SCD should maintain past knowledge while absorbing new one to alleviate knowledge forgetting. In addition, a student usually learn almost all of the knowledge in a simple sample through ground-truth labels alone. However, the simple structure of a student makes it difficult to learn rich knowledge contained in a hard sample. Of course, whether a sample is simple or hard is decided by the student, not the teacher. Accordingly, a teacher in SCD should pay special attention to the hard samples that a student consistently fails to master.

From the above analysis, in SCD, a teacher should not only learn and teach the knowledge that a student desires in the present and past, but also reduce knowledge forgetting. To effectively meet these conditions, we introduce the Proportional-Integral-Derivative (PID) control algorithm [11], [12] from the automation discipline. The proportional, integral, and derivative units can represent a student's current error, cumulative error, and error change trend on knowledge, respectively. Therefore, in SCD, the PID control algorithm has the potential to address a student's learning difficulties. Nonetheless, they still have to improve on several problems, to be shown in this paper. To overcome these problems, we design an PID control algorithm to effectively handle challenges that a student's current deficiency knowledge, hard sample learning, and knowledge forgetting. For example, the integral unit is still difficult to distinguish hard samples. The derivative unit is insufficient to reflect the degree of knowledge forgetting effectively.

It's worth noting that each of the proportional, integral, and derivative units in the PID control algorithm can only address a specific aspect of a student's learning difficulties. The relative sizes of these units are indicative of the importance of addressing these issues to enhance student's performance. For example, when a student struggles with hard sample learning, the ratio of the integral unit in the PID control algorithm should be increased. However, the learning state of the student is not constant, but rather constantly evolving. It means that we should dynamically adjust the ratios of proportional, integral, and derivative units to match the student's changing developmental needs. While in the field of automation, human-made fuzzy rules can adaptively control these ratios, they cannot be directly applied to implement student-centered distillation methods, to be shown in this paper. Inspired by the Curriculum Learning (CL) [13], [14] of human educational wisdom, we propose a CL-based fuzzy strategy for adaptively tuning the proportions of proportional, integral, and derivative units. It can make a student more active to explore rich knowledge in hard samples after mastering certain knowledge. More specifically, as student's performance improves, we propose a fuzzy strategy that reduces proportional unit and increases integral and derivative ones.

Therefore, we propose a student-centered distillation method by drawing on the ideas of student-centered and curriculum learning in education, introducing and improving the KD technology in artificial intelligence and the fuzzy PID control algorithm in cybernetics. We combine our CL-based fuzzy strategy and PID control algorithm, abbreviated as CL-FPID. CL-FPID can effectively figure out what knowledge the student desires at each epoch by controlling proportional, integral, and derivative units and their ratios. The complete implementation of this work is available at its Github repository.[1] In a word, this work aims to make the below contributions:

1) We propose a Student-Centered Distillation (SCD) method by mimicking student-centered human educational wisdom, where the teacher adjusts the teaching content according to the student's needs and abilities.
2) We propose a PID control algorithm to handle the problems of current deficiency knowledge, hard sample learning and knowledge forgetting in student learning.
3) We present a CL-based fuzzy strategy that allows a student to actively learn challenging samples after mastering certain knowledge.
4) We show the effectiveness of SCD through extensive experiments on multiple tasks. Experimental results show that the student-centered distillation method outperforms the existing teacher-centered ones.

Section II reviews the related work. Section III presents our proposed SCD method. Section IV provides theoretical analyses of our motivation. Section V compares SCD with its peers on multiple tasks. Section VI concludes the paper.

[1] https://github.com/yangshunzhi1994/SCD

II. Related Work

A. Knowledge Distillation

Deep learning networks has received widespread attention and rapidly developed in many fields for its robustness. However, these networks often require abundant resources, which are scarce in resource-constrained devices such as edge computing and mobile devices. To deploy deep learning networks on resource-constrained devices and meet the requirements of various application scenarios, researchers are exploring efficient methods. One such method is KD [1], which has gained popularity as a model compression technique in the field of deep learning.

Existing KD methods largely focus on three critical areas to improve the performance of a student [15]. The first area is determining what knowledge to transfer from a teacher to a student. In KD, knowledge is an abstract concept, and various aspects of a network, including gradients, outputs, and intermediate features, can all be considered as knowledge. For example, Robust [16] leverages gradient information as a form of knowledge transfer. CRD [17] utilizes contrastive objectives to capture the correlation of feature knowledge and the dependencies among higher-order outputs to acquire the knowledge of a teacher. DKD [18] decouples logits information into knowledge of target and non-target classes. A student in Filter-KD [19] learns the output knowledge of a teacher with label smoothing in early stopping epochs. Jiang et al. [20] use the weighting of ground-truth labels and soft targets predicted by a teacher as a supervised target for a student learning. FKD [21] transfers the knowledge of the inner product of last-layer representations across different sample inputs of a teacher into a student.

The second area is clarifying where the knowledge of a teacher should be transferred into a student. A teacher usually has more network layers than a student. Therefore, it is necessary to extract knowledge from the most representative network layers in a teacher. For example, CRKD [22] transfers a teacher's knowledge on the logits layer to a student. Ge et al. [23], [24] and Massoli et al. [25] indiscriminately and selectively transfer the knowledge at the mimic layer of a teacher with high-resolution samples as input into a student with low-resolution ones as input, respectively. KDEP [26] proposes a non-parametric feature size alignment method in the penultimate layer.

The third area is studying how to transfer knowledge from a teacher into a student. KD is to transfer knowledge from a teacher into a student by minimizing the closeness of their features. Therefore, some researches focus on loss functions to increase KD performance. For instance, Leap [27] minimizes the expected length of the gradient path of a meta-learner to reduce knowledge forgetting of a student. Huang et al. [28] propose a novel rank-based loss function that restricts the critical relations in the student to approximate those in the teacher. In addition to loss functions, some work proposes new distillation methods to achieve knowledge transfer between a teacher and student. For example, Jamal et al. [29] propose a "Lazy" MAML model, in which a teacher guides the learning of a student at intervals. A student in Annealing-KD [30] gets rid of the guidance of a teacher after acquiring certain knowledge. ProKT [31] constrains the path of student optimization by projecting the knowledge of a teacher into a student's parameter space. In SCKD [32], a student learns the feature knowledge of a teacher only if their gradients are similar. DGKD [33] gradually reduce the difference in network output between a teacher and student through multiple teacher assistants.

However, the above methods are all teacher-centered teaching ones, also known as knowledge cramming education [34], in which a teacher directly instills knowledge into a student. In other words, a student in teacher-centered method can only be in a "auditorium", losing the autonomy in the learning process. Different from teacher-centered methods, a student in our student-centered one can actively learn the knowledge it desires.

B. Student-Centered Distillation

One of the key challenges in implementing a student-centered KD method is determining which knowledge is desired by the student before the teacher can learn and transfer it. This implies that the method is an online KD one, where both the teacher and the student need to train from scratch. However, the problem with existing online KD methods [5], [35], [36], [37], [38], [39] is that they usually involve two untrained networks learning each other's knowledge, rather than learning and transferring knowledge based on each other's feedback. In human educational activities, teachers typically learn about students through questionnaire [7], case study reports or exam scores [7], etc. Similarly, this paper uses validation sets to discover the knowledge that the student desires. The validation set is akin to an "examination" that helps to identify the knowledge that the student has not yet fully grasped.

Currently, KD methods that involve student feedback are based on meta-learning, where the teacher relies on the gradient of the student's loss on the validation set to update its parameters [40], [41], [42]. However, due to differences in network architectures between the teacher and student, it can be challenging for the teacher to effectively absorb the gradient information returned by the student. In other words, the gradient of a student's loss on the examination may not be suitable for updating the teacher's parameters. Fortunately, we can figure out where a student's knowledge is relatively weak through an examination. For example, when a student's losses on validation samples A and B are L_A and L_B, respectively, where $L_A < L_B$, it indicates that the student is more interested in learning about sample B than sample A. This means that the teacher should enhance the understanding of the relevant knowledge in sample B and transfer it to the student. Thus, our approach uses the validation set to determine the knowledge desired by the student and allows the teacher to identify and learn the relevant knowledge before transferring it to the student.

III. Proposed Method

First of all, we provide the idea of our CL-FPID algorithm for implementing a student-centred distillation method. What's more, a complete SCD method is introduced in which a teacher learns and teaches the knowledge that a student desires. Once again, we describe how the teacher learns the knowledge that

the student desires through the validation set, including: the PID control algorithm that tackles the issues of current deficiency knowledge, hard sample learning and knowledge forgetting in student learning; the CL-based fuzzy strategy, which enables a student to actively learn the knowledge it desires with the help of a teacher. The PID control algorithm and CL-based fuzzy strategy, i.e., CL-FPID method, are used for a teacher to learn the knowledge that the student desire through a validation set. Finally, the teacher transfers the learned knowledge to the student through the training set.

A. Motivation and Design Ideas

The PID control algorithm is a form of feedback control in industrial applications that minimizes the error between actual output and desired values [43]. The PID control algorithm accomplishes this by generating a feedback signal from a set of proportional, integral, and derivative operations based on error. The proportional, integral, and derivative operations represent errors in present, past, and future states, respectively. Mathematically, it can be expressed as:

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt}, \tag{1}$$

where K_p, K_i, and K_d denote the control gains of proportional, integral, and derivative respectively. $u(t)$ is the feedback signal at time t, which is used to update the current state in order to attain optimal performance. It is worth mentioning that $u(t)$ is used to control the learning weight of a teacher on a sample from a validation set. If the $u(t)$ of a sample is relatively large, the teacher should focus more on the sample learning. $e(t)$ is proportional unit that represents the error at time t, i.e., the current loss of a student on a sample from the validation set. $\int_0^t e(t)dt$ is integral unit of the cumulative error, i.e., the difficulty for a student to grasp the knowledge in a sample from the validation set. A sample is considered hard if its $\int_0^t e(t)dt$ is relatively large. $\frac{de(t)}{dt}$ is derivative unit that implies the trend of the error, i.e., the state of knowledge forgetting by a student on a sample from the validation set. If a sample's $\frac{de(t)}{dt}$ is positive, the student has forgotten that knowledge, and the absolute value of $\frac{de(t)}{dt}$ indicates the degree of forgetting.

Therefore, the PID control algorithm in (1) has the potential to deal with the problems of current deficiency knowledge, hard sample learning and knowledge forgetting in student-centered distillation method. However, it has the following four issues in specific applications:

1) The PID control algorithm is only used to regulate one sample, not multiples. We should compute feedback values for multiple samples to obtain their relative importance.
2) If we treat the integral unit of each sample equally, it is difficult to distinguish which samples are easy and which are hard because their differences are small. We should make large integral units bigger and small ones smaller, enabling the teacher to focus more on samples that are hard for a student.
3) The derivative unit only considers the trend of the two most recent losses, i.e., the change rate between $e(t)$ and $e(t-1)$, which may not adequately reflect the degree of knowledge forgetting of a sample. We should calculate the amount of knowledge forgetting for a sample over more epochs.
4) K_p, K_i, and K_d are three constant parameters with no direct knowledge of the process, resulting in the sub-optimal of overall feedback algorithm. We should dynamically modify the three parameters of PID control algorithm according to the learning state of a student to improve the KD performance.

For the first three issues, we improve the PID control algorithm as:

$$\mathbf{u}(t) = K_p \mathbf{e}(t) + K_i \sigma\left(\int_{t-n-1}^{t} \mathbf{e}(t)dt\right) + K_d \frac{d^n e(t)}{dt^n} \in \mathbb{R}^N, \tag{2}$$

where N is the number of samples in an iteration, and $\mathbf{u}(\mathrm{t})$ and $\mathbf{e}(\mathrm{t})$ respectively are the feedback value and loss size of N samples, which are used to solve the first issue. $\sigma(\cdot)$ is a function to reduce the size of small values and increase the size of large ones to deal with the second issue. For the third concern of capturing long-term knowledge forgetting, we employ an nth order difference, expressed as $\sum_{i=0}^{n}(-1)^i \binom{n}{i} \mathbf{e}(t-i)$. For $n = 1$, this difference emphasizes the change between $\mathbf{e}(t)$ and $\mathbf{e}(t-1)$; at $n = 2$, it captures the variations among $\mathbf{e}(t)$, $\mathbf{e}(t-1)$, and $\mathbf{e}(t-2)$, allowing a holistic view of knowledge loss over epochs.

For the fourth issue, it is challenging for a deep learning algorithm to directly optimize discrete hyperparameters. To overcome this challenge, we introduce a fuzzy PID control algorithm [44] that incorporates human empirical knowledge. The fuzzy PID control algorithm first calculates current error and its change rate, then does inference using fuzzy rules to identify the current values of three parameters K_p, K_i, and K_d. The performance of the fuzzy PID control algorithm is heavily influenced by the quality of fuzzy rules. The existing rules are primarily designed for situations in which the error has both positive and negative values [45], [46], such as speed and direction control. However, the error of a student on a sample can only be positive, rendering the existing fuzzy rules inapplicable to the research in this paper. To enable PID parameters to automatically adapt to the learning progress of a student, we propose a CL-FPID algorithm based on curriculum learning [13], an easy-to-hard training method that mimics human educational wisdom. In human education, it is typical for a student to master certain knowledge at the primary school stage before effectively learning more challenging knowledge in secondary school. Drawing inspiration from this concept, we propose a fuzzy control strategy that allows a student to actively learn challenging samples after mastering certain knowledge.

In this paper, if a student is still difficult to master the knowledge of a sample after multiple training, it is considered that the sample is challenging for the student. The difficulty degree of a sample is reflected by the cumulative errors of the student,

TABLE I
LIST OF THE MAIN SYMBOLS AND NOTATIONS

Notation	Description
$(x_i^{val}, y_i) \in \mathbb{D}_{val}$	The i^{th} sample x_i^{val} from validation set $\mathbb{D}_{val}$ and its truth label y_i.
$(x_i^{train}, y_i) \in \mathbb{D}_{train}$	The i^{th} sample x_i^{train} from training set $\mathbb{D}_{train}$ and its truth label y_i.
$(x_i^{test}, y_i) \in \mathbb{D}_{test}$	The i^{th} sample x_i^{test} from testing set $\mathbb{D}_{test}$ and its truth label y_i.
$f(\theta_T)$	The teacher with parameter θ_T.
$f(\theta_T^p)$	The teacher with parameter θ_T^p before update on validation set.
$f(\theta_T^a)$	The teacher with parameter θ_T^a after update on validation set.
$f(\theta_S)$	The student with parameter θ_S.
w_i	The importance of knowledge in sample x_i^{val} to the current student.
$\bar{w}_i$	The importance of knowledge in sample x_i^{train} to the current student.
L_i	The loss size of a sample x_i^{val} on $f(\theta_T)$.
L_i^t	The loss size of x_i^{val} on $f(\theta_S)$ at its t^{th} epoch.
$\bar{L}_i$	The loss size of a sample x_i^{train} on $f(\theta_T)$.
$\tilde{L}_i$	The loss size of the knowledge distillation of a sample x_i^{train} on $f(\theta_S)$.

i.e., the integral unit. The integrated unit allows a student to focus on the knowledge that is currently most difficult to master. However, in different epochs, the student usually focuses on different challenging samples. For example, the student focuses on learning the knowledge of the hard sample "◊" at the ith epoch, whereas it is "□" at the $(i+1)$th epoch. Unfortunately, the student may forget knowledge "◊" after learning knowledge "□". It is known as knowledge forgetting [10], i.e., when the student is focused on learning hard samples, it is more likely to ignore the learning of other ones. As the student focuses more on learning hard samples, the issue of knowledge forgetting becomes more pronounced. The method proposed in this paper addresses the issue of knowledge forgetting during student learning by increasing the ratio of derivative units. Thus, as student's performance improves, we use fuzzy strategy to reduce proportional unit and increase integral and derivative ones.

It is worth noting that the student informs the teacher of what knowledge it desires instead of updating on the validation set. The teacher learns the knowledge required by the student on the validation set and transfers it into the student in the process of KD. After the teacher has learned the knowledge required by a student on the validation set, it is necessary to identify which training samples correspond to this knowledge. To achieve this, a score for the teacher's knowledge in a training sample is computed by comparing its output before and after being updated on the validation set. It helps analyze the relative importance of knowledge in a training sample to a student in the current state.

B. Overview of the Proposed Method

A summary of the notation to be used in this paper is provided in Table I. The method of the complete training and testing of SCD is illustrated in Fig. 1. Our SCD method is based on the following intuition: A teacher identifies the importance w_i of the knowledge of the sample x_i^{val} to a student by analyzing the current student's performance on the validation set $\mathbb{D}_{val}$.

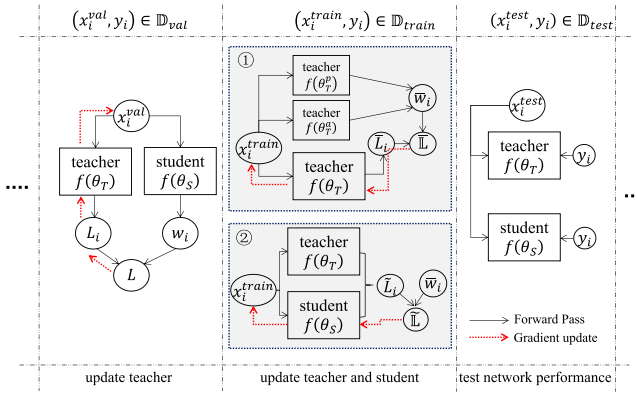


Fig. 1. Proposed SCD method.

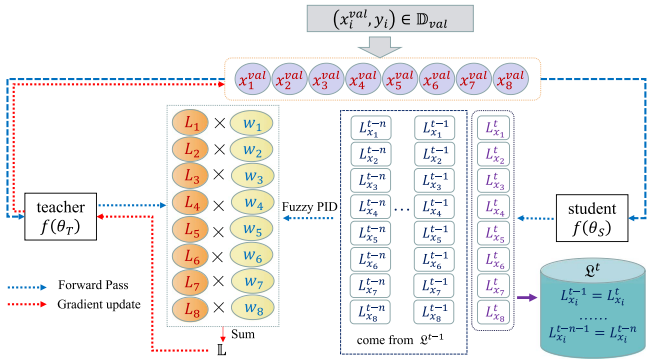


Fig. 2. Method of teacher learns what student desires. $\mathfrak{L}^t$ records the historical performance of the student on each validation sample. We propose the Fuzzy PID method to evaluate the weight of each sample by fully considering the current error, the past accumulated error, and the future trend of the student.

Through w_i, a teacher can learn the knowledge desired by a student on $\mathbb{D}_{val}$, as presented on the left in Fig. 1. A teacher learns knowledge on $\mathbb{D}_{val}$, and then transmits it to the student via the training set $\mathbb{D}_{train}$. However, the samples in $\mathbb{D}_{val}$ and $\mathbb{D}_{train}$ are not the same, and the only connection between them is the teacher. By comparing the output of a sample x_i^{train} on $f(\theta_T^p)$ and $f(\theta_T^a)$, we can derive the importance score $\bar{w}_i$ of its knowledge to the current state student, as depicted in step ① on the middle in Fig. 1. $f(\theta_T^p)$ and $f(\theta_T^a)$ are used to represent the teacher's network parameters before and after the update on the validation set, respectively. $\bar{w}_i$ is used for the teacher to learn knowledge desired by the student and to transfer them to the latter in step ②. Finally, we validate the performance of the teacher and student on the testing set $\mathbb{D}_{test}$, as offered on the right in Fig. 1.

C. Teacher Learns What Student Desires

A teacher learns the knowledge that a student desires through a validation set, as presented in Fig. 2. This figure illustrates the learning process with eight samples as examples. To optimize the parameters of a deep learning network, several iterations (or epochs) are required. In each iteration, we randomly select N_1 samples $\{x_i^{val}, y_i\}_i^{N_1}$ from the validation

set $\mathbb{D}_{val} = \{x_i^{val}, y_i\}_i^{N_V}$ for updating the teacher, where $y_i \in \mathbb{N}_K = \{1, 2, \ldots, K\}$. Here, N_V is the total number of samples in $\mathbb{D}_{val}$, x_i^{val} is the ith validation sample and $i \in \mathbb{N}_{N_V}$, y_i is the corresponding ground truth label and K is the total number of image classes. Let $f(\theta_T)$ denote the teacher with independently trained parameters θ_T at the tth epoch. Take a sample x_i^{val} from the $\{x_i^{val}\}_i^{N_1}$ as the input of $f(\theta_T)$, its output logits $\mathbf{t}_i = f(x_i^{val}; \theta_T) = (\mathbf{t}_{i,1}, \ldots, \mathbf{t}_{i,k}, \ldots, \mathbf{t}_{i,K}) \in \mathbb{R}^K$. For a multi-class classification problem, we use cross-entropy loss to calculate the loss size of teacher $f(\theta_T)$ on sample x_i^{val}:

$$L_i = -\sum_{k=1}^{K} \mathbf{y}_{i,k} \log \mathbf{p}_{i,k}, \tag{3}$$

$$\mathbf{p}_{i,k} = \text{softmax}\left(\mathbf{t}_{i,k}\right) = \frac{\exp\left(\mathbf{t}_{i,k}\right)}{\sum_{i=1}^{k} \exp\left(\mathbf{t}_{i,k}\right)}, \tag{4}$$

where $\mathbf{y}_{i,k}$ and $\mathbf{p}_{i,k}$ respectively represent the true and predicted class probability of the x_i^{val} on the kth category. It is to be noted that $\mathbf{y}_{i,k}$ is the kth value in the one-hot encoding of the true label y_i. Assuming that the value of knowledge in sample x_i^{val} for improving the present student's performance is w_i, the loss size of the teacher in the current iteration is as follows:

$$\mathbb{L} = \sum_{i=1}^{N_1} w_i \times L_i. \tag{5}$$

Then, the teacher is updated on the validation set by taking one gradient descent step with the loss function above, i.e.,

$$\theta_T = \theta_T - \alpha \nabla_{\theta_T} \mathbb{L}, \tag{6}$$

where α is the learning rate of the teacher. We perform multiple iterations so that all validation samples are trained in one epoch. It allows the teacher to learn the knowledge that the student desires through the validation set.

In order for the teacher to accurately learn the knowledge desired by the student, we are required to be able to evaluate w_i effectively, where w_i is obtained from $\mathbf{u}(t)$ in (2). Following that, we introduce in detail the proportional, integral, and derivative units in (2) and the reasoning methods of their hyperparameters K_p, K_i and K_d, as offered in Sections III-C1 and III-C2, respectively.

1) Proposed PID Control Algorithm: Our PID control algorithm focuses on the improvement of the proportional, integral, and derivative units in (2), making them capable of implementing a student-centered distillation method. For a sample x_i^{val} from $\{x_i^{val}\}_i^{N_1}$, we use L_i^t to denote the size of its cross-entropy loss on the student $f(\theta_S)$, where t denotes the current epoch and $t \geq 0$. Similarly, the loss size of the sample x_i^{val} at the $(t-n)$th epoch is denoted by L_i^{t-n}. In experiments, we found that when $n = 2$, the derivative term is more effective in responding to the degree of knowledge forgetting, to be shown in this paper. Therefore, our PID control algorithm can be specifically formulated as follows:

$$\mathbf{u}(t) = K_p \mathbf{e}(t) + K_i \sigma\left(\frac{\mathbf{e}(t) + \mathbf{e}(t-1) + \mathbf{e}(t-2)}{3}\right) + K_d \times (\mathbf{e}(t) - 2\mathbf{e}(t-1) + \mathbf{e}(t-2)) \in \mathbb{R}^{N_1}, \tag{7}$$

$$\mathbf{e}(t) = \left[\{L_i^t\}_i^{N_1}\right] \in \mathbb{R}^{N_1}, \tag{8}$$

where $(\mathbf{e}(t) - 2\mathbf{e}(t-1) + \mathbf{e}(t-2))$ is derived from $\frac{d^2\mathbf{e}(t)}{dt^2}$. $\mathbf{e}(t)$ contains the loss size of each sample in an iteration. $\sigma(\cdot)$ is a function that makes small values smaller and larger ones larger, so that the feedback values of hard and easy samples can be clearly distinguished. Concretely, $\sigma(\cdot)$ is defined as follows:

$$\sigma\left(\mathbf{x}\right) = \tanh(\mathbf{x}) = \left(e^{\mathbf{a}} - e^{-\mathbf{a}}\right) / \left(e^{\mathbf{a}} + e^{-\mathbf{a}}\right) \in \mathbb{R}^{N_1}, \tag{9}$$

$$\mathbf{a} = \frac{2}{\max(\mathbf{x}) - \min(\mathbf{x})} \times (\mathbf{x} - \min(\mathbf{x})) - 1 \in \mathbb{R}^{N_1}, \tag{10}$$

where $\mathbf{x}$ represent the mean of the cumulative error of the samples $\{x_i^{val}\}_i^{N_1}$, i.e., $\mathbf{x} = (\mathbf{e}(t) + \mathbf{e}(t-1) + \mathbf{e}(t-2))/3 \in \mathbb{R}^{N_1}$. $\mathbf{a}$ is the normalized form of $\mathbf{x}$ and any value in $\mathbf{a}$ belongs to $[-1, 1]$, which helps differentiate the values in $\mathbf{x}$ after the $\tanh$ function.

It should be noted that L_i^t, L_i^{t-1} and L_i^{t-2} are the loss sizes of the same sample at the tth, $(t-1)$th and $(t-2)$th epochs, respectively. The establishment of (7) occurs when $t \geq 3$. When $t < 3$, the following formula is used to determine $\mathbf{u}(t)$:

$$\mathbf{u}(t) = K_p \mathbf{e}(t), \quad s.t. \quad t = 1. \tag{11}$$

$$\mathbf{u}(t) = K_p \mathbf{e}(t) + K_i \sigma\left(\frac{\mathbf{e}(t) + \mathbf{e}(t-1)}{2}\right) + K_d \left(\mathbf{e}(t) - \mathbf{e}(t-1)\right), \; s.t. \; t = 2. \tag{12}$$

For hyperparameters K_p, K_i and K_d, our CL-based fuzzy strategy dynamically tunes them, as detailed in Section III-C2. After obtaining the feedback value $\mathbf{u}(t)$, we use the softmax function to calculate the importance w_i of the knowledge of a sample x_i^{val} in $\{x_i^{val}\}_i^{N_1}$ to the current student:

$$w_i = \text{softmax}\left(\mathbf{u}_i(t)\right) = \frac{\exp\left(\mathbf{u}_i(t)\right)}{\sum_{i=1}^{N_1} \exp\left(\mathbf{u}_i(t)\right)}, \tag{13}$$

where $\mathbf{u}_i(t)$ is the ith value in $\mathbf{u}(t)$, i.e., the feedback value of the sample x_i^{val}. It is worth noting that the student does not update on the sample x_i^{val}, but feeds back to the teacher that the importance of the sample's knowledge to the student in the current state is w_i.

2) Our CL-Based Fuzzy Strategy: Our CL-based fuzzy strategy dynamically tunes the parameters of K_p, K_i and K_d according to the learning state of the student, i.e., L_i^t, to optimize $\mathbf{u}(t)$ in real time. Fuzzy self-tuning PID controllers are calculated in the domain of fuzzy sets. Therefore, we first quantify the average performance of the student on the most recent test, i.e., the average loss of all validation samples in $\mathbf{e}(t-1)$, before determining the parameters of K_p, K_i and K_d for the current state. The student's performance at $(t-1)$th epoch can be stated as follows:

$$\bar{e} = \frac{1}{N_V} \sum_{i=1}^{N_V} L_i^{t-1}. \tag{14}$$

In automation disciplines, a fuzzy set usually consists of a membership function with seven linguistic variables (NB, NM,

NS, ZO, PS, PM, PB), which represent negative big, negative medium, negative small, zero, positive small, positive medium, and positive big, respectively [47]. We followed this time-tested practical experience to divide the variable-range of $\bar{e}$ into seven grades. Specifically, let the student's maximum loss size on the validation set is L_m, we divide L_m into a membership function $\mathbf{m}$ with seven linguistic variables on average. Therefore, the membership function of $\bar{e}$ is:

$$m_1 = \begin{cases} 1 & \bar{e} < \frac{1}{7}L_m \\ \left(\frac{2}{7}L_m - \bar{e}\right) / \frac{1}{7}L_m & \frac{1}{7}L_m \le \bar{e} < \frac{2}{7}L_m \\ 0 & \text{else} \end{cases} \quad (15)$$

$$m_{i \atop i\in\{2,3,4,5,6\}} = \begin{cases} \left(\bar{e} - \frac{i-1}{7}L_m\right) / \frac{1}{7}L_m & \frac{i-1}{7}L_m \le \bar{e} < \frac{i}{7}L_m \\ \left(\frac{i+1}{7}L_m - \bar{e}\right) / \frac{1}{7}L_m & \frac{i}{7}L_m < \bar{e} \le \frac{i+1}{7}L_m \\ 0 & \text{else} \end{cases} \quad (16)$$

$$m_7 = \begin{cases} \left(\bar{e} - \frac{6}{7}L_m\right) / \frac{1}{7}L_m & \frac{6}{7}L_m < \bar{e} \le L_m \\ 1 & \bar{e} > L_m \\ 0 & \text{else} \end{cases} \quad (17)$$

$$\mathbf{m} = [m_1, m_2, m_3, m_4, m_5, m_6, m_7] \in \mathbb{R}^7. \quad (18)$$

The membership function $\mathbf{m}$ is used to quantify the performance of a student. Specifically, the greater the value of m_1, the better the performance of the student; the higher the value of m_7, the worse the performance of the student.

Second, we set the variable range $\mathbf{kp}$ of K_p according to the same interval Δp, and the median value of $\mathbf{kp}$ is as the initial value K_{p0} of K_p, i.e.,

$$\mathbf{kp} = [0, \Delta p, 2\Delta p, 3\Delta p, 4\Delta p, 5\Delta p, 6\Delta p] \in \mathbb{R}^7, \quad (19)$$

$$K_{p0} = 3\Delta p, \quad (20)$$

where Δp is a hyperparameter that controls the range of K_p values.

Third, we perform defuzzification which uses the centroid to derive the exact K_p value:

$$K_p = \sum_{i=0}^{7} \mathbf{kp}_i \times \mathbf{m}_i. \quad (21)$$

Finally, we make the following tunes to K_i and K_d in light of K_p:

$$K_i = K_{i0} + (K_{p0} - K_p), \quad (22)$$

$$K_d = K_{d0} + (K_{p0} - K_p), \quad (23)$$

where K_{i0} and K_{d0} are hyperparameters, and both K_i and K_d are greater than or equal to 0. (22) and (23) are our proposed CL-based strategies. In general, we first input $\bar{e}$ into the membership function to calculate $\mathbf{m}$, and then infer the value of K_p according to the preset $\mathbf{kp}$ and (21). The higher the $\bar{e}$ value, the higher the K_p value, and the lower the K_i and K_d values. At the same time, the smaller the value of $\bar{e}$, the smaller the value of K_p, and the larger the value of K_i and K_d. The dynamic tune process of K_p, K_i and K_d is primarily inspired by the curriculum learning of human educational wisdom, so that the student can pay more attention to the integral, and derivative units after mastering certain knowledge.

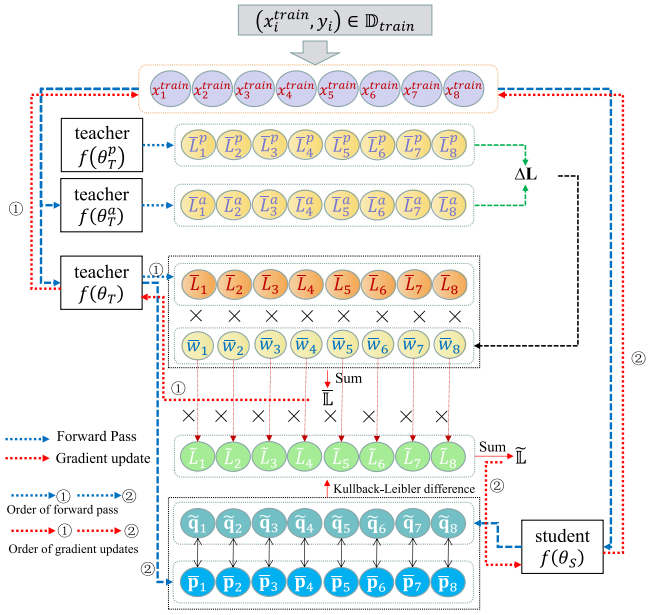


Fig. 3. Method of teacher teaches what student desires. We evaluate the importance $\bar{w}_i$ of a training sample for the current student by measuring the difference between its outputs on $f(\theta_T^p)$ and $f(\theta_T^a)$. $\bar{w}_i$ is used to guide the teacher $f(\theta_T)$ to learn the knowledge that the student desires, and transfer it to the latter through distillation training.

D. Teacher Teaches What Student Desires

A teacher teaches the knowledge that a student desires through a training set, as presented in Fig. 3. Eight samples are used to demonstrate the teaching process in this figure. Let x_i^{train} represents the ith sample on the training set $\mathbb{D}_{train}$, and $f(\theta_T^p)$ and $f(\theta_T^a)$ denote the parameters before and after the nearest update of the teacher $f(\theta_T)$ on the validation set $\mathbb{D}_{val}$, respectively. We identify the knowledge learned by $f(\theta_T)$ on the validation set by the difference Δl_i of the output of sample x_i^{train} on $f(\theta_T^p)$ and $f(\theta_T^a)$:

$$\Delta l_i = \bar{L}_i^p - \bar{L}_i^a, \quad (24)$$

where $\bar{L}_i^p$ and $\bar{L}_i^a$ are the loss sizes of x_i^{train} on $f(\theta_T^p)$ and $f(\theta_T^a)$, respectively. It is worth noting that the value of Δl_i can be positive or negative. The larger the value of Δl_i, the more knowledge that $f(\theta_T)$ has learned in a sample x_i^{train} through $\mathbb{D}_{val}$. The teacher can use Δl_i to focus on learning the knowledge that the student desires on the training set.

In each iteration, we randomly select N_2 samples $\{x_i^{train}, y_i\}_i^{N_2}$ from the training set $\mathbb{D}_{train} = \{x_i^{train}, y_i\}_i^{N_T}$ for updating the teacher and student, where N_T is the total number of samples in $\mathbb{D}_{train}$. Thus, the importance score of the knowledge in the $\{x_i^{train}, y_i\}_i^{N_2}$ samples to the current student can be expressed as:

$$\Delta\mathbf{L} = [\Delta l_1, \ldots, \Delta l_i, \ldots, \Delta l_{N_2}] \in \mathbb{R}^{N_2}. \quad (25)$$

Then, we use a softmax function to calculate the relative importance $\bar{w}_i$ of the knowledge in a sample x_i^{train} from $\{x_i^{train}, y_i\}_i^{N_2}$ for the current student:

$$\bar{w}_i = \text{softmax}\left(\gamma \Delta \mathbf{L}_i\right) = \frac{\exp\left(\gamma \Delta \mathbf{L}_i\right)}{\sum_{i=1}^{N_2} \exp\left(\gamma \Delta \mathbf{L}_i\right)}, \tag{26}$$

where $\Delta \mathbf{L}_i$ is the ith value in $\Delta \mathbf{L}$ and γ is a hyperparameter. γ is used to tune the weight of the relative importance of the samples. When γ is large, the differences in $\{\bar{w}_i\}_i^{N_2}$ among N_2 samples are big; when it is small, the differences are tiny.

$\bar{w}_i$ is used to update the teacher. In the same way as we did in (3) and (4), we use cross-entropy to determine the loss of the teacher on a training sample. Let $\bar{L}_i$ denote the loss size of a sample x_i^{train} on the teacher $f(\theta_T)$, the loss of $f(\theta_T)$ in the current iteration is:

$$\bar{\mathbb{L}} = \sum_{i=1}^{N_2} \bar{w}_i \times \bar{L}_i. \tag{27}$$

Then, the teacher is updated on the training set by taking one gradient descent step with the loss function above, i.e.,

$$\theta_T = \theta_T - \beta \nabla_{\theta_T} \bar{\mathbb{L}}, \tag{28}$$

where β is the learning rate of the teacher.

After the teacher learns the knowledge that the student desires on the samples $\{x_i^{train}, y_i\}_i^{N_2}$, the former transfers that knowledge into the latter. We propagate the trained teacher $f(\theta_T)$ forward once again, and set the output logits of the sample x_i^{train} as $\bar{\mathbf{t}}_i = f(x_i^{train}; \theta_T) = (\bar{\mathbf{t}}_{i,1}, \ldots, \bar{\mathbf{t}}_{i,k}, \ldots, \bar{\mathbf{t}}_{i,K}) \in \mathbb{R}^K$. Simultaneously, let the output logits of the student $f(\theta_S)$ in the sample x_i^{train} be $\tilde{\mathbf{s}}_i = f(x_i^{train}; \theta_S) = (\tilde{\mathbf{s}}_{i,1}, \ldots, \tilde{\mathbf{s}}_{i,k}, \ldots, \tilde{\mathbf{s}}_{i,K}) \in \mathbb{R}^K$. We use "Kullback-Leibler difference" as an indicator and try to minimize the output distribution between teacher and student:

$$\tilde{L}_i = \tau^2 \sum_{k=1}^{K} \bar{\mathbf{p}}\left(\bar{\mathbf{t}}_{i,k}, \tau\right) \log_2 \frac{\bar{\mathbf{p}}\left(\bar{\mathbf{t}}_{i,k}, \tau\right)}{\tilde{\mathbf{q}}\left(\tilde{\mathbf{s}}_{i,k}, \tau\right)}, \tag{29}$$

$$\bar{\mathbf{p}}\left(\bar{\mathbf{t}}_{i,k}, \tau\right) = \frac{\exp\left(\bar{\mathbf{t}}_{i,k}/\tau\right)}{\sum_{k=1}^{K} \exp\left(\bar{\mathbf{t}}_{i,k}/\tau\right)}, \tag{30}$$

$$\tilde{\mathbf{q}}\left(\tilde{\mathbf{s}}_{i,k}, \tau\right) = \frac{\exp\left(\tilde{\mathbf{s}}_{i,k}/\tau\right)}{\sum_{k=1}^{K} \exp\left(\tilde{\mathbf{s}}_{i,k}/\tau\right)}, \tag{31}$$

where $\bar{\mathbf{p}}(\bar{\mathbf{t}}_{i,k}, \tau)$ and $\tilde{\mathbf{q}}(\tilde{\mathbf{s}}_{i,k}, \tau)$ are the "dark knowledge" of the kth class in $\bar{\mathbf{t}}_i$ and $\tilde{\mathbf{s}}_i$, respectively, and τ is a higher temperature factor. Thus, the distillation loss of the student in SCD on the current iteration is expressed as:

$$\tilde{\mathbb{L}} = \sum_{i=1}^{N_2} \bar{w}_i \times \tilde{L}_i. \tag{32}$$

We can also use the ground-truth labels to train the student, i.e., $\tilde{\mathbb{L}} = \tilde{\mathbb{L}} - \sum_{k=1}^{K} \mathbf{y}_{i,k} \log \tilde{\mathbf{q}}(\tilde{\mathbf{s}}_{i,k}, \tau = 1)$, where $\mathbf{y}_{i,k}$ is the true label of the kth category. Then, the student is updated on the training set by taking one gradient descent step with the loss function above, i.e.,

$$\theta_S = \theta_S - \beta \nabla_{\theta_S} \tilde{\mathbb{L}}, \tag{33}$$

where β is the learning rate of the student. The learning rate is the same in both (28) and (33) to ensure a consistent pace of learning between teacher and student. We perform multiple iterations so that each training example in an epoch can be used to update the teacher and student.

TABLE II
MAIN SYMBOLS USED IN THEORETICAL ANALYSIS

Notation	Description
T_1	A teacher in existing work.
T_2	A teacher in our work.
A	Difficult datasets for the student to learn.
B	Datasets that are easy for the student to grasp.
$C_A(T)$	The volume of knowledge learned by teacher T on A.
$L(T, X)$	Loss function used by teacher T on dataset X.

IV. THEORETICAL ANALYSIS

This section proves the importance of a teacher learning the knowledge that the student desires. We assume that in a dataset, the data subset B consists of the samples that the student can learn effectively by labeling, and the data subset A consists of the samples that are difficult to learn effectively. The teacher T has a limited size and can only learn some sample knowledge. Therefore, we suggest that the teacher focus on learning the data subset A to use its capacity well and improve the student's performance. We call the teacher in the existing work T_1 and the teacher in our approach T_2, both with the same size. The difference is that T_1 does not use feedback to determine which samples of knowledge the student desires, and learns from both data subsets B and A, while T_2 learns only from data subset A. To prove that T_2 learns more or equal knowledge from data subset A than T_1, i.e., to prove that

$$C_A\left(T_2\right) \geq C_A\left(T_1\right).$$

Table II shows a detailed description of the symbols in this section.

We assume that the teacher's loss function $L(T, A)$ on A is a convex function with respect to θ_T, and that there exists an optimal solution θ_T^* such that $L(T, A)$ is minimized. Then, we can use gradient descent to update the teacher's parameters and write their parameter updating rules for each step:

- Parameter update rules for T_1:

$$\theta_T^{(t+1)} = \theta_T^{(t)} - \alpha \nabla L\left(T_1, A\right) - \beta \nabla L\left(T_1, B\right).$$

- Parameter update rules for T_2:

$$\theta_T^{(t+1)} = \theta_T^{(t)} - \alpha \nabla L\left(T_2, A\right).$$

To show that our method can make the teacher parameter θ_T converge to θ_T^*, or at least be closer to θ_T^* than existing methods, we utilize the following property:

- If $L(T, A)$ is a convex function on θ_T, then it has a unique global minimum θ_T^* and satisfies

$$\nabla L(T, A)|_{\theta_T = \theta_T^*} = 0.$$

- If $L(T, A)$ and $L(T, B)$ are both convex functions with respect to θ_T, then their linear combination is also a convex

function with respect to θ_T and satisfies

$$\nabla(L(T,A)+L(T,B))=\nabla L(T,A)+\nabla L(T,B).$$

Based on these properties, we can obtain the following derivation:

- For T_1, if it converges to some parameter value $\hat{\theta}_T$, then it must satisfy

$$\nabla L\left(T_1,A\right)|_{\theta_T=\hat{\theta}_T}+\nabla L\left(T_1,B\right)|_{\theta_T=\hat{\theta}_T}=0.$$

- For T_2, if it converges to some parameter value $\tilde{\theta}_T$, then it must satisfy

$$\nabla L\left(T_2,A\right)|_{\theta_T=\tilde{\theta}_T}=0.$$

As $L(T,A)$ is a convex function with respect to θ_T and there exists a unique global minimum θ_T^*, then

$$L(T,A)|_{\theta_T=\hat{\theta}_T}\geq L(T,A)|_{\theta_T=\tilde{\theta}_T}\geq L(T,A)|_{\theta_T=\theta_T^*}.$$

Since $L(T,B)$ is a convex function on θ_T and there is a unique global minimum (not necessarily θ_T^*), then we have

$$L(T,B)|_{\theta_T=\hat{\theta}_T}\geq L(T,B)|_{\theta_T=\tilde{\theta}_T}\geq L(T,B)|_{\min}.$$

Combining the above two inequalities, we get

$$L\left(T_1,A\right)|_{\theta_T=\hat{\theta}_T}+L\left(T_1,B\right)|_{\theta_T=\hat{\theta}_T}\geq L\left(T_2,A\right)|_{\theta_T=\tilde{\theta}_T}+L\left(T_2,B\right)|_{\min}.$$

Assuming that A and B contain different or opposite knowledge, it can be assumed that $L(T_2,B)|_{\min}$ is a larger value because T_2 is not trained on B, so it has a low level of knowledge about B. Then, we can further obtain

$$L\left(T_1,A\right)|_{\theta_T=\hat{\theta}_T}+L\left(T_1,B\right)|_{\theta_T=\hat{\theta}_T}\geq L\left(T_2,A\right)|_{\theta_T=\tilde{\theta}_T}+L\left(T_2,B\right)|_{\min}\gg L\left(T_2,A\right)|_{\theta_T=\tilde{\theta}_T}.$$

This implies that the total loss of T_1 on A and B is much larger than the loss of T_2 on A. In other words, the performance of T_1 on A is much lower than that of T_2 on A. Since we assume that the teacher's loss function $L(T,A)$ on A is convex with respect to the teacher's parameter θ_T, and that there is an optimal solution θ_T^* that minimizes $L(T,A)$, then we can assume that the teacher's performance on A is positively correlated with the amount of knowledge the teacher learned on A, i.e., T_2 has more knowledge than T_1. This means that

$$L(T,A)\downarrow\Rightarrow C_A(T)\uparrow.$$

Therefore, we can conclude that

$$L\left(T_1,A\right)\gg L\left(T_2,A\right)\Rightarrow C_A\left(T_1\right)\ll C_A\left(T_2\right).$$

V. Experiments

Three sets of experiments are conducted to the performance of the proposed SCD with the existing KD methods. The first set is the cross-resolution experiment, wherein the teacher and the student use high- and low-resolution samples as the input, respectively. It assesses the performance of the proposed approach by generating low-resolution datasets from multiple existing high-resolution ones. The second set, referred to as the same-resolution experiment, used the same samples as input for both teacher and student. It evaluates the effectiveness of the proposed method on various real-world computer vision datasets. The third set is the different-architecture experiment, wherein we test our method on various teacher-student architectures, following the existing KD literature. Finally, we carry out various ablation studies and conduct a thorough qualitative assessment to effectively highlight the proposed SCD method. The details of the datasets and experimental implementations are presented in the Supplementary File.

A. Experimental Results for Cross-Resolution Tasks

Low-resolution (LR) object recognition is a technology with important application value in different scenarios, such as video surveillance systems, remote sensing image analysis, etc. LR images have lower resolution and less information content than normal images, leading to poor feature extraction and classification performance. Moreover, LR object recognition also faces many challenges, especially in recognizing LR targets captured from a far distance on resource-constrained edge computing devices in real time [55]. To improve the recognition performance of LR targets, KD is an effective solution, which can use the "privileged information" provided by high-resolution (HR) samples and teacher to guide the learning of student [55]. Privileged information refers to the rich details contained in HR samples, such as clear textures, explanations, comments, and comparisons. This subsection validates the advantages of SCD on cross-resolution tasks. Specifically, the teacher and the student use the same sample as input, but the former's image size is larger than the latter's. We provide the performance of SCD and its peers on multiple LR object recognition tasks and a task at different resolutions, including very low resolution, as presented in Tables III and IV, respectively. We evaluate performance using three metrics: accuracy rate (A), unweighted average recall (U), and F1 score (F1). A is the top-1 accuracy rate, which means the percentage of times that the model predicts the correct class with the highest probability. U is the average accuracy of each class across all categories, for unbalanced data. F1 is a weighted average of precision and recall.

According to the experimental results in Tables III and IV, our method outperforms all other methods in the A. In most cases, U and F1 are optimal. Because of the unbalanced data distribution, our method is occasionally suboptimal in terms of U and F1. In this paper, we save the network parameters of each method at the highest A in the experiments and then calculate their corresponding U and F1. Although U and F1 of our method are not the best in a few cases, they are quite near to the best results in the testing. It convincingly proves the robustness of our SCD method. It also means that the teacher of other popular methods have more room for improvement in learning and transferring the knowledge that a student desires. Furthermore, Table IV shows that the lower the resolution of the samples input into the student, the worse the student's performance in general.

As indicated in Table II of Supplementary File and Table III, there are three distinctions among all KD approaches. The first one is that the student in some studies has learned other knowledge from a teacher in addition to soft targets. For example, Robust [16] minimizes the difference between the gradients

TABLE III
EXPERIMENTAL RESULTS ON LOW-RESOLUTION TASKS

Work	RAF-DB [48]			Oxford-IIIT Pet [49]			FairFace [50]			Food-101 [51]			Places-Extra69 [52]			Logo-2K+ [53]			IMDB-WIKI [54]		
	A	U	F1	A	U	F1	A	U	F1	A	U	F1	A	U	F1	A	U	F1	A	U	F1
Ge et al. [23]	85.98%	76.44%	78.06%	55.30%	55.24%	54.84%	67.75%	67.20%	67.28%	65.22%	65.22%	65.07%	54.71%	52.33%	49.90%	52.15%	39.19%	41.39%	46.60%	38.11%	40.51%
Ge et al. [24]	85.72%	75.73%	78.70%	60.97%	60.94%	60.72%	69.14%	68.65%	68.80%	62.21%	62.21%	61.87%	53.35%	51.03%	48.69%	54.40%	43.34%	46.02%	45.55%	38.11%	39.87%
MPL [40]	85.89%	77.48%	79.00%	22.27%	22.44%	16.76%	68.30%	67.96%	68.14%	67.69%	67.69%	67.55%	53.53%	51.20%	46.86%	51.64%	39.69%	42.37%	46.49%	37.64%	39.83%
EEM [14]	85.95%	76.33%	78.40%	50.67%	50.61%	50.50%	68.16%	67.65%	67.83%	60.72%	60.72%	60.48%	53.91%	51.57%	49.29%	53.24%	42.50%	44.34%	45.68%	37.58%	39.73%
CRKD [22]	85.82%	76.60%	78.37%	49.25%	49.24%	49.17%	68.45%	67.82%	68.04%	61.35%	61.35%	61.04%	53.80%	51.46%	48.99%	53.17%	42.02%	44.63%	45.88%	37.95%	40.01%
Robust [16]	83.57%	74.57%	75.97%	47.40%	47.39%	47.07%	58.65%	57.68%	57.37%	55.84%	55.84%	55.39%	53.77%	51.43%	49.14%	48.51%	35.03%	37.88%	44.86%	35.14%	37.18%
CRD [17]	84.75%	75.37%	76.97%	47.89%	47.84%	47.57%	67.81%	67.08%	67.24%	56.91%	56.91%	56.73%	51.09%	48.87%	46.59%	50.71%	40.11%	41.79%	46.20%	37.07%	39.36%
DKD [18]	85.75%	76.36%	77.91%	58.98%	58.94%	58.87%	69.35%	68.91%	69.10%	66.59%	66.59%	66.33%	54.85%	52.46%	**50.17%**	51.61%	37.61%	41.49%	46.75%	38.87%	40.92%
Filter-KD [19]	84.97%	76.94%	77.37%	44.18%	44.16%	43.87%	67.37%	66.98%	66.81%	63.16%	63.16%	63.09%	52.06%	49.80%	44.95%	51.92%	37.90%	41.84%	41.90%	31.83%	31.49%
Jiang et al. [20]	85.79%	76.51%	78.17%	49.93%	49.91%	49.59%	68.99%	68.39%	68.59%	64.00%	64.00%	63.71%	54.30%	51.94%	49.29%	54.32%	43.63%	45.84%	46.78%	38.48%	40.38%
FKD [21]	86.31%	77.89%	79.16%	49.20%	49.14%	49.49%	68.87%	68.32%	68.33%	40.06%	40.06%	39.05%	45.44%	43.46%	39.58%	44.88%	29.86%	31.67%	46.21%	37.82%	40.14%
Massoli et al. [25]	85.07%	75.23%	76.87%	54.51%	54.47%	54.12%	67.32%	66.67%	66.76%	60.83%	60.83%	60.49%	52.62%	50.33%	47.87%	54.06%	43.28%	45.54%	45.47%	35.71%	38.18%
KDEP [26]	85.89%	77.65%	79.16%	61.27%	61.20%	61.11%	68.74%	68.16%	68.44%	64.45%	64.45%	64.26%	51.47%	49.23%	46.44%	53.63%	43.36%	45.27%	46.41%	**39.85%**	**41.53%**
Leap [27]	85.24%	74.73%	77.01%	19.68%	19.84%	14.00%	68.74%	68.24%	68.34%	67.17%	67.17%	66.95%	53.09%	50.78%	46.68%	51.08%	38.42%	41.30%	46.38%	38.75%	41.02%
Huang et al. [28]	86.08%	77.44%	79.09%	49.41%	49.36%	49.16%	67.69%	67.47%	67.43%	61.72%	61.72%	61.38%	53.26%	50.94%	48.52%	53.57%	42.59%	45.00%	45.28%	36.85%	38.67%
"Lazy" MAML [29]	84.49%	74.91%	76.65%	42.19%	42.15%	41.54%	66.70%	65.94%	66.04%	59.38%	59.38%	58.96%	52.52%	50.23%	47.40%	52.87%	41.89%	43.83%	44.76%	36.69%	38.45%
Annealing-KD [30]	85.27%	76.19%	78.01%	48.52%	48.47%	48.30%	67.49%	66.93%	66.83%	62.63%	62.63%	62.48%	53.32%	51.00%	48.67%	54.54%	**43.81%**	46.32%	45.55%	37.52%	39.04%
ProKT [31]	86.18%	76.64%	78.62%	55.30%	55.26%	54.80%	69.43%	69.07%	69.02%	68.05%	68.05%	67.86%	50.97%	48.75%	45.00%	50.83%	38.91%	41.57%	46.63%	39.28%	40.85%
SCKD [32]	85.30%	75.22%	77.70%	50.38%	50.33%	49.75%	67.29%	66.94%	66.94%	61.20%	61.20%	61.04%	52.80%	50.51%	48.22%	53.90%	43.07%	45.21%	45.21%	37.11%	38.97%
DGKD [33]	85.98%	75.49%	78.04%	58.30%	58.23%	57.88%	69.13%	68.38%	68.78%	61.42%	61.42%	61.13%	53.58%	51.25%	48.62%	54.61%	43.24%	46.20%	46.52%	38.20%	40.38%
IKD [41]	85.33%	75.90%	77.79%	46.28%	46.23%	45.75%	67.26%	66.70%	66.70%	61.52%	61.52%	61.34%	49.17%	47.03%	44.17%	54.12%	43.52%	45.73%	44.23%	34.33%	35.35%
MetaDistil [42]	84.42%	75.81%	77.57%	42.57%	42.60%	42.50%	68.17%	67.65%	67.88%	61.29%	61.29%	60.99%	53.12%	50.81%	48.23%	53.57%	43.14%	45.06%	45.98%	36.87%	39.13%
SCD	**86.96%**	**78.66%**	**80.54%**	**63.21%**	**63.13%**	**62.83%**	**69.50%**	**69.18%**	**69.14%**	**68.26%**	**68.26%**	**68.01%**	**55.36%**	**52.96%**	49.96%	**54.87%**	43.60%	**46.69%**	**46.85%**	38.82%	41.04%

Best results are bold-faced.

TABLE IV
EXPERIMENTAL RESULTS ON DIFFERENT LOW-RESOLUTION CELEBA [56] DATASET

Work	44×44			32×32			24×24			16×16			8×8		
	A	U	F1	A	U	F1	A	U	F1	A	U	F1	A	U	F1
Ge et al. [23]	93.12%	65.99%	67.55%	92.54%	65.39%	68.53%	91.84%	61.77%	64.53%	91.01%	63.76%	67.37%	84.92%	47.67%	52.03%
Ge et al. [24]	92.99%	67.75%	68.53%	93.09%	67.13%	69.56%	92.42%	61.36%	65.19%	91.73%	63.10%	65.52%	85.57%	45.09%	50.21%
MPL [40]	92.58%	68.32%	67.89%	93.15%	63.93%	67.61%	92.63%	61.07%	63.53%	91.68%	59.67%	63.14%	85.50%	47.66%	51.94%
EEM [14]	93.39%	68.60%	70.15%	92.71%	66.74%	69.44%	92.12%	64.15%	66.62%	91.05%	62.96%	66.58%	84.93%	47.27%	52.05%
CRKD [22]	93.35%	62.65%	66.59%	92.74%	66.10%	71.19%	92.01%	61.84%	64.14%	91.37%	62.12%	65.04%	85.00%	47.70%	51.64%
Robust [16]	93.64%	68.99%	72.09%	92.96%	67.06%	70.38%	92.32%	64.07%	67.61%	91.40%	61.20%	64.84%	85.42%	47.11%	52.11%
CRD [17]	91.82%	63.99%	64.82%	92.62%	65.89%	67.89%	91.93%	61.16%	65.30%	91.05%	60.35%	63.92%	85.10%	44.20%	48.70%
DKD [18]	93.36%	66.16%	69.14%	93.17%	67.04%	70.93%	92.60%	64.41%	67.04%	91.67%	63.20%	66.32%	85.29%	47.27%	51.07%
Filter-KD [19]	92.95%	56.79%	55.90%	93.40%	64.48%	67.91%	92.66%	60.20%	64.30%	91.89%	61.15%	64.23%	85.66%	45.23%	49.96%
Jiang et al. [20]	93.47%	68.65%	70.21%	93.02%	67.37%	69.80%	92.41%	64.51%	67.56%	91.51%	63.66%	66.59%	85.08%	47.50%	52.27%
FKD [21]	93.39%	67.48%	69.55%	93.41%	67.63%	71.72%	92.13%	62.93%	66.00%	91.88%	61.69%	64.49%	85.02%	44.06%	47.88%
Massoli et al. [25]	93.11%	68.48%	69.23%	92.60%	64.64%	67.88%	91.80%	63.34%	65.77%	91.11%	56.84%	60.78%	84.90%	45.13%	49.79%
KDEP [26]	93.13%	66.41%	68.89%	93.00%	64.95%	67.45%	92.35%	64.24%	65.87%	91.51%	64.09%	65.49%	84.71%	43.19%	47.76%
Leap [27]	92.37%	60.68%	63.74%	93.07%	68.10%	70.70%	92.71%	61.16%	64.78%	91.79%	62.20%	65.92%	85.53%	45.24%	50.00%
Huang et al. [28]	93.20%	66.69%	69.76%	92.57%	64.81%	68.14%	92.06%	63.74%	65.67%	90.96%	61.99%	65.12%	84.95%	46.76%	51.32%
"Lazy" MAML [29]	93.13%	**69.54%**	70.60%	92.43%	66.87%	69.58%	91.83%	62.36%	65.27%	90.82%	60.96%	65.07%	84.86%	45.10%	50.30%
Annealing-KD [30]	93.10%	64.25%	67.92%	92.64%	65.61%	69.09%	91.90%	62.40%	65.96%	91.00%	61.65%	65.11%	84.88%	46.55%	51.64%
ProKT [31]	93.43%	63.12%	66.51%	93.30%	68.37%	71.42%	92.59%	61.81%	65.48%	91.80%	62.37%	65.74%	85.36%	44.98%	49.47%
SCKD [32]	93.34%	67.01%	69.31%	92.41%	61.55%	65.90%	91.97%	62.90%	66.47%	91.01%	58.43%	62.02%	84.89%	46.09%	50.50%
DGKD [33]	93.63%	67.80%	70.00%	92.93%	65.81%	68.94%	92.29%	64.40%	66.57%	91.26%	62.63%	63.91%	85.02%	46.93%	51.91%
IKD [41]	93.54%	68.58%	71.75%	92.60%	67.50%	70.62%	92.08%	60.46%	64.20%	90.98%	62.99%	65.33%	84.92%	47.57%	51.75%
MetaDistil [42]	93.35%	64.15%	66.34%	92.66%	63.65%	68.17%	91.89%	**64.61%**	**68.15%**	91.23%	62.88%	65.77%	84.74%	45.90%	50.56%
SCD	**93.87%**	68.47%	**72.30%**	**93.42%**	**68.50%**	**72.06%**	**92.72%**	63.58%	66.23%	**91.93%**	**64.26%**	**67.59%**	**85.83%**	**47.71%**	**52.64%**

Best results are bold-faced.

of the student and teacher. The student learns the gradient knowledge of the teacher, which enhances its adversarial robustness. However, learning knowledge in hard samples is tough for the student but relatively simple for the teacher. Therefore, when face with hard samples, the teacher learns as easy samples and returns less knowledge into the student. The students in CRD [17] and KDEP [26] imitate the knowledge of a teacher in the mimic and penultimate layers, respectively. Generally speaking, the more knowledge a student gains from a teacher, the better the student's performance [55]. Thus, many research tend to explore the various kinds of knowledge existing in the teacher, such as relational knowledge [2], [3]. However, such knowledge may not be what the student desires, making it harder to increase their performance. It can be seen from the experimental results that even if the students in the peers acquires more knowledge, their performances are still poor.

The second one is that the number of training samples available to the student is not the same in all studies. In cases where a validation set is not required, we use all of the training set to update the student. However, for methods that require a validation set, such as our SCD, we partition the training set into separate training and validation ones. A validation set is required for four methods: our SCD, MPL [40], Leap [27] and "Lazy" MAML [29]. Among them, Leap [27] updates the student on the validation set, while the remaining three do not. Generally, a larger number of training samples enables the student to acquire more knowledge. Therefore, the student in SCD does not have much knowledge to learn compared to most of its peers. Nevertheless, despite this disadvantage, the student in SCD still performs well.

The third one is that the student in some methods only learns from the knowledge provided by its teacher, without self-learning from the ground-truth labels, i.e., MPL [40], Filter-KD [19], ProKT [31] and SCD. In these four studies, none of the ground-truth labels are exposed to the student. Therefore, the student's performance is entirely dependent on the teacher's dark knowledge, which consists of two elements: the quality of the dark knowledge generated by the teacher and the extent to

which it is absorbed by the student. Some studies [19], [61] have shown that the closer the dark knowledge output by a teacher is to the ground-truth dark knowledge, the better the performance of its student. However, none of these studies consider whether the student can effectively absorb the knowledge offered by the teacher. The knowledge learned by the teacher may be too obscure for the student due to their significant capacity gap, making it challenging to improve the student's performance. For example, the teacher in MPL [40] and our SCD achieve accuracies of 87.16% and 86.51%, respectively, on the RAF-DB dataset. However, the performance of their students is reversed, with the SCD's student performing better than the MPL's student. The reason for this phenomenon is that the knowledge learned by the teacher in the existing work may not be what the student desires. Thus, we explain the observations of Cho et al. [62] and our previous work [55] from the perspective of human teaching experiences, i.e., a more competent teacher does not necessarily produce a better student. Although the quality of dark knowledge is critical for improving student's performance, its quality should be assessed by the student, not the teacher.

Overall, our experimental results show that a teacher should learn and impart knowledge that the student desires. The validation set serves as a "examination sheet" in a student-centered method, containing almost all of the knowledge that the student has to master. It can highlight the knowledge that the student desires and help the teacher to learn it in a focused manner. Therefore, with the help of the examination sheets, a teacher in our student-centered distillation method can focus more on the knowledge that the student desires, enabling the latter to study more effectively. Furthermore, pupils' absorption of knowledge is a step-by-step process in human teaching. For this purpose, we propose CL-FPID that can adapt to the student's learning process, making it easier to not only accept knowledge but also to learn more actively, thereby improving performance.

In addition, we explain the reason why a good teacher may not be able to teach a better student, i.e., the quality of a teacher should be evaluated by the student. Knowledge in a teacher may be useless if it is unnecessary or obscure for the student. This occurrence is also common in human educational efforts. For example, a professor with ordinary research ability may produce better undergraduates than one with outstanding research skills. This phenomena could be caused by two factors: 1) the former has relatively strong teaching ability, allowing the student to successfully absorb the knowledge it has gained; 2) the latter is not good at teaching, making it difficult for the student to absorb its knowledge effectively. Thus, it is not true that a higher-performing teacher must be able to teach a better-performing student.

B. Experimental Results for Same-Resolution Tasks

In addition to transferring rich privileged information from a teacher with a HR sample as input to a student with a LR sample as input, KD can transfer knowledge from a teacher with strong learning ability to a student with weak learning ability. The sample resolution for the teacher and student is required to be different in the former and the same in the latter. Therefore, we can term them as cross-resolution and same-resolution tasks, respectively. Experiments on same- and cross-resolution tasks have different setups and motivations. In terms of experimental setup, experiments on cross-resolution samples require the downsampling of HR samples as the input to a student, whereas experiments on same-resolution samples do not. Experiments on cross-resolution samples are typically carried out to improve a student's performance in processing LR object recognition, while experiments on same-resolution samples are not necessarily used for that purpose. In this subsection, we verify the performance of the proposed SCD method on the same-resolution tasks for object recognition, object detection, and object verification, respectively.

1) Experimental Results on Object Recognition: We validate the performance of the proposed SCD on general object recognition tasks with CIFAR-100 [57], CINIC-10 [58], SVHN [59] and Tiny-ImageNet [60] datasets. All four datasets are popular datasets and their widely used to evaluate the performance of a KD method. The experimental results are presented in Table V. Similar to cross-resolution tasks, a fraction of the samples in the training set is used as the validation set in these experiments. Specifically, we select 5,000 training samples on each dataset for the teacher to identify and learn the knowledge desired by the student. As a result, the student in our approach can only learn the knowledge transferred by the teacher from a relatively small number of samples compared to most of its peers. Nevertheless, SCD can also perform well on the general object recognition tasks.

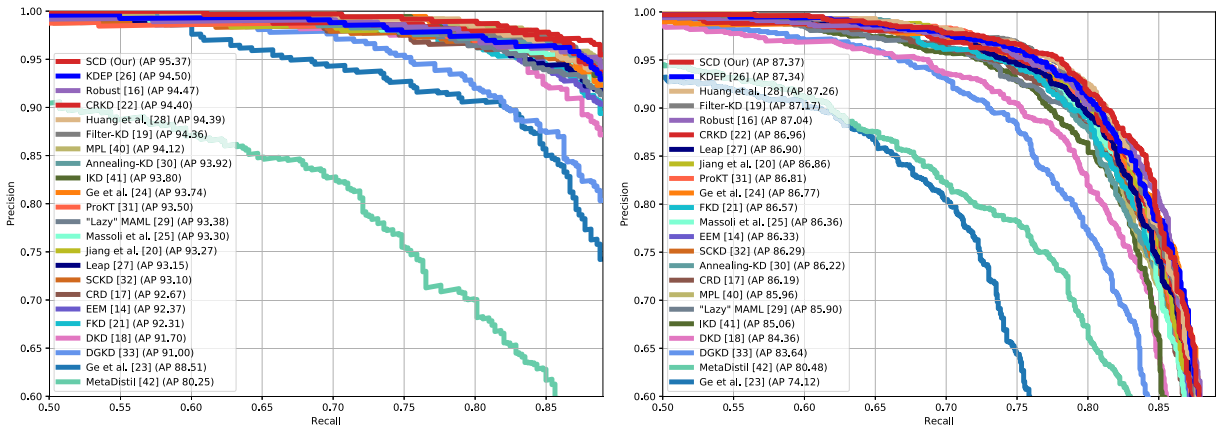
2) Experimental Results on Object Detection: We follow the experimental setup outlined by Zhang et al.'s [63] to validate the performance of the proposed SCD method on an object detection task. Specifically, each knowledge distillation method is trained on WIDER FACE [64]'s training set, while their performance is evaluated on AFW [65] and PASCAL [66] datasets. This evaluation method not only effectively tests the generalization performance of each method, but also validates their performance in real-world scenarios. Our evaluation metric for object detection is Intersection over Union (IoU), where an object is considered detected if the IoU is at least 0.5 [67]. In line with typical testing practices, we evaluated the performance of our proposed SCD and its peers at IoU $= 0.5$, as indicated in Fig. 4. Experimental results demonstrate that our student-centered distillation method outperforms existing teacher-centered methods for object detection task.

3) Experimental Results on Object Verification: This subsection aims to assess the effectiveness of SCD and other existing distillation methods in face verification tasks, which involve determining whether two faces belong to the same person. Each distillation method is trained using an IJB-C dataset [68]. The experiment performs object recognition during training, and object verification during testing. The former enables each distillation method to extract facial features, while the latter uses the extracted features to determine whether two unknown faces belong to the same individual. This approach is ideal for testing the overall performance of each method, similar to the object detection task.

TABLE V
EXPERIMENTAL RESULTS ON CIFAR-100, CINIC-10, SVHN AND TINY-IMAGENET DATASETS

Work	CIFAR-100 [57]			CINIC-10 [58]			SVHN [59]			Tiny-ImageNet [60]		
	A	U	F1	A	U	F1	A	U	F1	A	U	F1
Ge et al. [23]	72.95%	72.95%	72.84%	82.64%	82.64%	82.63%	96.51%	96.37%	96.35%	59.72%	59.72%	59.51%
Ge et al. [24]	73.21%	73.21%	73.13%	82.56%	82.56%	82.57%	96.72%	96.69%	96.58%	59.16%	59.16%	58.93%
MPL [40]	70.43%	70.43%	70.35%	82.69%	82.69%	82.63%	96.26%	96.10%	96.06%	57.22%	57.22%	56.98%
EEM [14]	71.47%	71.47%	71.38%	82.25%	82.25%	82.24%	96.66%	96.51%	96.49%	58.09%	58.09%	57.82%
CRKD [22]	71.44%	71.44%	71.36%	82.18%	82.18%	82.12%	96.55%	96.45%	96.40%	57.72%	57.72%	57.52%
Robust [16]	70.14%	70.14%	70.15%	73.30%	73.30%	73.17%	94.60%	94.24%	94.20%	56.83%	56.83%	56.63%
CRD [17]	71.24%	71.24%	71.17%	81.92%	81.92%	81.87%	96.40%	96.23%	96.19%	57.48%	57.48%	57.25%
DKD [18]	74.52%	74.52%	74.38%	74.68%	74.68%	74.69%	96.71%	96.50%	96.56%	56.61%	56.61%	56.18%
Filter-KD [19]	71.27%	71.27%	71.17%	77.12%	77.12%	77.09%	96.24%	95.94%	96.04%	57.04%	57.04%	56.79%
Jiang et al. [20]	71.42%	71.42%	71.30%	82.29%	82.29%	82.29%	96.77%	96.64%	96.60%	59.38%	59.38%	59.14%
FKD [21]	71.21%	71.21%	71.13%	81.51%	81.51%	81.49%	96.65%	96.52%	96.50%	57.39%	57.39%	57.12%
Massoli et al. [25]	71.61%	71.61%	71.46%	81.96%	81.96%	81.89%	96.38%	96.20%	96.22%	57.76%	57.76%	57.58%
KDEP [26]	73.47%	73.47%	73.37%	83.20%	83.20%	83.22%	96.77%	**96.77%**	96.61%	61.26%	61.26%	61.05%
Leap [27]	70.71%	70.71%	70.58%	81.98%	81.98%	81.96%	96.34%	96.18%	96.21%	57.46%	57.46%	57.24%
Huang et al. [28]	71.66%	71.66%	71.64%	81.96%	81.96%	81.91%	96.42%	96.33%	96.27%	57.48%	57.48%	57.20%
"Lazy" MAML [29]	70.04%	70.04%	69.94%	81.57%	81.57%	81.60%	96.20%	95.99%	96.00%	57.44%	57.44%	57.26%
Annealing-KD [30]	70.90%	70.90%	70.87%	81.52%	81.52%	81.55%	96.32%	96.13%	96.14%	57.17%	57.17%	56.83%
ProKT [31]	73.22%	73.22%	73.22%	82.11%	82.11%	82.10%	96.68%	96.58%	96.53%	61.16%	61.16%	61.01%
SCKD [32]	70.85%	70.85%	70.78%	81.76%	81.76%	81.74%	96.35%	96.08%	96.18%	56.66%	56.66%	56.52%
DGKD [33]	71.66%	71.66%	71.65%	83.38%	83.38%	83.36%	96.71%	96.57%	96.53%	57.72%	57.72%	57.41%
IKD [41]	70.44%	70.44%	70.36%	82.18%	82.18%	82.18%	96.48%	96.38%	96.32%	57.14%	57.14%	56.95%
MetaDistil [42]	70.22%	70.22%	70.24%	81.73%	81.73%	81.73%	96.55%	96.38%	96.40%	56.92%	56.92%	56.76%
SCD	**74.61%**	**74.61%**	**74.52%**	**83.49%**	**83.49%**	**83.48%**	**96.81%**	96.66%	**96.63%**	**61.44%**	**61.44%**	**61.20%**

Best results are bold-faced.



(a) AFW dataset (b) PASCAL dataset

Fig. 4. Precision-recall curves on object detection tasks. AP = average precision.

We tested each distillation method's performance on the IJB-C dataset at every epoch, saving the optimal network parameters. We then evaluated the methods' performance on the LFW [69], SCface [70] and TinyFace [71] datasets using these saved parameters. The performance of our SCD with its peers on the object verification tasks of IJB-C, LFW, SCface and TinyFace is presented in Table VI and indicate that our methods outperform its 22 peers.

C. Experimental Results for Different Teacher-Student Architectures

In this subsection, we evaluate the performance of our proposed SCD method under the teacher and the student have different network architectures. SCD is an online KD method that only utilizes the teacher's knowledge at the output layer, which is different from most existing methods that use intermediate layers or additional losses. We compare our method with MetaDistil [42], a state-of-the-art online KD method that uses meta-learning to optimize the distillation loss based on the

TABLE VI
EXPERIMENTAL RESULTS ON IJB-C, LFW, SCFACE, AND TINYFACE DATASETS

Work	IJB-C [68]	LFW [69]	SCface [70]	TinyFace [71]
Ge et al. [23]	92.92 ± 4.36%	77.83 ± 1.77%	48.20 ± 20.39%	79.76 ± 13.14%
Ge et al. [24]	93.50 ± 4.12%	77.92 ± 1.14%	50.45 ± 21.22%	77.97 ± 13.00%
MPL [40]	92.11 ± 5.01%	74.08 ± 2.20%	42.46 ± 17.46%	78.45 ± 13.59%
EEM [14]	92.99 ± 4.32%	78.48 ± 1.09%	42.86 ± 18.94%	78.54 ± 13.13%
CRKD [22]	92.83 ± 4.39%	77.45 ± 1.44%	44.00 ± 17.54%	78.14 ± 12.63%
Robust [16]	93.63 ± 4.01%	78.72 ± 1.69%	49.20 ± 17.98%	78.84 ± 12.85%
CRD [17]	92.96 ± 4.57%	78.02 ± 2.20%	47.17 ± 20.90%	78.35 ± 13.12%
DKD [18]	93.19 ± 4.06%	76.70 ± 1.63%	47.74 ± 19.69%	79.23 ± 12.01%
Filter-KD [19]	92.51 ± 4.92%	77.75 ± 1.53%	43.73 ± 20.66%	78.52 ± 13.12%
Jiang et al. [20]	92.71 ± 4.83%	75.73 ± 1.72%	49.03 ± 26.80%	78.28 ± 14.24%
FKD [21]	92.67 ± 4.88%	77.05 ± 1.65%	52.20 ± 24.87%	79.16 ± 11.51%
Massoli et al. [25]	93.39 ± 4.22%	77.80 ± 1.85%	45.34 ± 18.09%	78.60 ± 13.70%
KDEP [26]	93.23 ± 4.20%	78.17 ± 1.92%	46.20 ± 21.14%	79.63 ± 11.81%
Leap [27]	91.77 ± 4.88%	72.50 ± 1.51%	46.76 ± 23.14%	79.35 ± 11.93%
Huang et al. [28]	89.02 ± 8.84%	67.03 ± 1.60%	46.79 ± 21.67%	76.16 ± 15.50%
"Lazy" MAML [29]	92.43 ± 4.66%	74.62 ± 2.42%	50.20 ± 19.78%	79.49 ± 13.08%
Annealing-KD [30]	92.92 ± 3.89%	78.45 ± 1.36%	48.43 ± 20.56%	79.12 ± 11.16%
ProKT [31]	93.16 ± 4.07%	77.13 ± 1.51%	47.81 ± 19.77%	78.21 ± 12.93%
SCKD [32]	92.88 ± 4.30%	77.28 ± 1.54%	46.59 ± 17.97%	79.20 ± 12.59%
DGKD [33]	92.80 ± 4.76%	77.133 ± 1.72%	47.90 ± .22.40%	78.71 ± 12.79%
IKD [41]	91.89 ± 4.76%	75.87 ± 1.51%	45.26 ± 19.15%	76.60 ± 14.39%
MetaDistil [42]	92.91 ± 4.44%	75.82 ± 1.87%	52.74 ± 20.23%	78.27 ± 13.54%
SCD	**93.81** ± 3.88%	**78.83** ± 1.59%	**53.06** ± 21.27%	**79.84** ± 11.85%

The table displays the mean and standard deviations of test accuracy obtained from 10-fold cross-validation (best results are bold-faced).

TABLE VII
EXPERIMENTAL RESULTS ON CIFAR-100 DATASET WITH DIFFERENT TEACHER-STUDENT ARCHITECTURES

Feedback	Teacher	ResNet-56	ResNet-110	ResNet-110	VGG-13	ResNet-50
	Accuracy	72.34%	74.31%	74.31%	74.64%	79.34%
	Student	ResNet-20	ResNet-20	ResNet-32	VGG-8	VGG-8
	Accuracy	69.06%	69.06%	71.14%	70.36%	70.36%
No	KD	70.66%	70.67%	73.08%	72.98%	73.81%
	FitNet	69.21%	68.99%	71.06%	71.02%	70.69%
	AT	70.55%	70.22%	72.31%	71.43%	71.84%
	SP	69.67%	70.04%	72.69%	72.68%	73.34%
	CC	69.63%	69.48%	71.48%	70.71%	70.25%
	VID	70.38%	70.16%	72.61%	71.23%	70.30%
	RKD	69.61%	69.25%	71.82%	71.48%	71.50%
	PKT	70.34%	70.25%	72.61%	72.88%	73.01%
	AB	69.47%	69.53%	70.98%	70.94%	70.65%
	FT	69.84%	70.22%	72.37%	70.58%	70.29%
	ProKT	70.98%	70.74%	72.95%	73.03%	73.90%
	CRD	71.16%	71.46%	73.48%	73.94%	74.30%
Yes	MetaDistil	71.25%	71.40%	73.35%	73.65%	74.42%
	SCD	**71.41%**	**71.92%**	**73.66%**	**74.31%**	**74.94%**
	Δ	+0.16%	+0.52%	+0.31%	+0.66%	+0.52%

For the details of the baselines, please refer to MetaDistil [42].
Best results are bold-faced.

TABLE VIII
EXPERIMENTAL RESULTS ON IMAGENET DATASET WITH DIFFERENT TEACHER-STUDENT ARCHITECTURES

Teacher	ResNet-18	ResNet-50	ResNet-50
Accuracy	69.95%	76.28%	76.28%
Student	ShuffleV2	ShuffleV2	ResNet-18
Accuracy	64.25%	64.25%	69.95%
DML [5]	65.35%	65.34%	71.03%
KDCL [35]	64.58%	64.49%	70.34%
AFD [37]	64.72%	65.42%	70.85%
PCL [36]	65.29%	63.59%	70.08%
CKD-MKT [38]	64.52%	65.43%	71.02%
L-MCL [39]	66.06%	66.44%	71.69%
SCD	**67.47%**	**68.69%**	**71.79%**

For the details of the baselines, please refer to L-MCL [39].

TABLE IX
TEST ACCURACY (%) OF STUDENTS ON MS-COCO2017 DATESET

T&S Model	Methods	mAP	AP50	AP75	APl	APm	APs
Teacher	R101-FPN(T)	42.04	62.48	45.88	54.60	45.55	25.22
Student	R18-FPN(S)	33.26	53.61	35.26	43.16	35.68	18.96
R101& R18	Leap [27]	33.80	54.78	35.91	44.29	35.92	19.04
	MPL [40]	33.89	54.93	**36.28**	44.53	36.11	**19.79**
	IKD [41]	33.81	54.64	36.06	43.80	36.12	18.95
	MetaDistil [42]	33.84	54.72	36.10	44.04	36.10	18.95
	SCD	**34.02**	**55.30**	36.22	**44.62**	**36.43**	19.39
Teacher	R101-FPN(T)	42.04	62.48	45.88	54.60	45.55	25.22
Student	R50-FPN(S)	37.93	58.84	41.05	49.10	41.14	22.44
R101& R50	Leap [27]	38.10	59.66	41.24	49.34	41.17	22.46
	MPL [40]	38.12	59.68	41.25	49.75	41.23	22.25
	IKD [41]	38.26	59.65	41.50	49.28	41.60	22.21
	MetaDistil [42]	38.23	59.50	41.30	49.51	**41.80**	22.33
	SCD	**38.31**	**59.70**	**41.61**	**49.94**	41.54	**22.73**
Teacher	R50-FPN(T)	40.22	61.02	43.81	51.98	43.53	24.16
Student	MV2-FPN(S)	29.47	48.87	30.90	38.86	30.77	16.33
R50& MV2	Leap [27]	30.93	51.88	32.42	40.64	32.74	17.55
	MPL [40]	30.91	51.83	32.26	40.54	**33.06**	17.51
	IKD [41]	30.58	51.05	31.63	39.71	32.62	17.14
	MetaDistil [42]	30.61	51.08	32.38	40.28	32.44	**17.69**
	SCD	**31.11**	**52.06**	**32.68**	**41.19**	32.87	17.48

The best results in each experiment are emphasized in bold. Resnet110, resnet50, resnet18, and mobilenetv2 are denoted as r101, r50, r18, and mv2, respectively. T&S is teacher and student

student's feedback. MetaDistil is the most relevant method to ours, as it also only uses the knowledge at the output layer and is an online distillation method. We also compare our method with L-MCL [39], another state-of-the-art online KD method that uses contrastive learning to enhance the knowledge distillation process. L-MCL is chosen as a baseline for ImageNet, because MetaDistil did not report results on this dataset. We conduct experiments on three datasets: CIFAR-100, ImageNet and COCO. The experimental results are shown in Tables VII, VIII and IX, respectively. We observe that using a larger teacher can improve the student's performance, but our method can still generate a better student than existing methods under the same teacher-student architecture. This indicates that our method can effectively leverage a teacher's knowledge.

D. Ablation Studies

This subsection investigates the influence of hyperparameters on student's performance, evaluates the effectiveness of each module in the proposed approach, and provides the training cost associated with implementing the proposed method.

1) Effect of Parameters on Student's Performance: In this paper, K_p, K_i, and K_d control the size of the proportional, integral, and derivative units in the improved algorithm, respectively. They are calculated from Δp, K_{i0}, K_{d0} based on the student's performance. τ is the temperature and is used to control the smoothness of the knowledge transferred from a teacher to a student. γ is used to control the relative importance of the knowledge to be learned by the teacher. L_m is the magnitude of the cross-entropy loss of a validation sample. The number of linguistic variables in the membership function $\mathbf{m}$ controls the fine-grained level of the student performance's grades. We select the CIFAR-100 dataset in the same-resolution task to validate the effect of those hyperparameters on student's performance, as shown in Table X. We verify the effect of the value of a parameter on the student's performance when the other parameters are fixed. Experimental results demonstrated that the optimal student's performance on the same-resolution task of CIFAR-100 dataset was achieved with the following parameters: $\Delta p = 0.04$, $K_{i0} = 0.0$, $K_{d0} = 0.1$, $\tau = 20$, $\gamma = 0.06$, $L_m = 2.1$, $n = 2$, and seven linguistic variables.

2) Effect of the Proposed CL-FPID on Student's Performance: To further understand the importance of each component in SCD, we perform the following multiple ablation studies:

- CNN-RIS: the performance of student without KD.
- CNN-RIS-KD: the performance of a student using the KD of Hinton et al. [1].
- CNN-RIS-PID: the performance of a student on KD under the PID algorithm in automation.
- CNN-RIS-PID (Our): the performance of a student on KD under our improved PID algorithm.
- CNN-RIS-CL-FPID: the performance of a student on KD under our CL-FPID algorithm, i.e., SCD method.

Their experimental results are shown in Table XI. Experimental results show that both our proposed PID control algorithm and CL-based fuzzy strategy are beneficial to improve the performance of a student. Moreover, the performance between CNN-RIS-PID and CNN-RIS-PID (Our) confirms the PID algorithm from the automation field has limited improvement on the student's performance. In this paper, the PID algorithms are

TABLE X
EFFECT OF PARAMETER ON STUDENT'S PERFORMANCE

(a) Effect of Δp on student's performance, where $K_{i0} = 0.1$, $K_{d0} = 0.1$, $\gamma = 0.3$, $L_m = 2.1$, $\tau = 20$, $n = 2$, and seven linguistic variables.

Δp	0.1	0.3	0.2	0.02	0.04	0.06	0.08	0.05
A	73.06%	69.24%	69.55%	73.43%	**73.90%**	73.82%	73.47%	73.71%

(b) Effect of K_{i0} on student's performance, where $\Delta p = 0.04$, $K_{d0} = 0.1$, $\gamma = 0.3$, $L_m = 2.1$, $\tau = 20$, $n = 2$, and seven linguistic variables.

K_{i0}	0.1	0.3	0.2	0.00	0.02	0.04	0.06	0.08
A	73.90%	73.47%	73.64%	**73.99%**	73.62%	73.65%	73.28%	73.91%

(c) Effect of K_{d0} on student's performance, where $\Delta p = 0.04$, $K_{i0} = 0.0$, $\gamma = 0.3$, $L_m = 2.1$, $\tau = 20$, $n = 2$, and seven linguistic variables.

K_{d0}	0.1	0.3	0.2	0.00	0.02	0.04	0.06	0.08
A	**73.99%**	73.07%	73.76%	73.87%	73.56%	73.77%	73.48%	73.62%

(d) Effect of γ on student's performance, where $\Delta p = 0.04$, $K_{i0} = 0.0$, $K_{d0} = 0.1$, $L_m = 2.1$, $\tau = 20$, $n = 2$, and seven linguistic variables.

γ	0.1	0.3	0.2	0.02	0.04	0.06	0.08	0.05
A	74.56%	73.99%	74.14%	74.05%	74.35%	**74.61%**	74.32%	74.40%

(e) Effect of L_m on student's performance, where $\Delta p = 0.04$, $K_{i0} = 0.0$, $K_{d0} = 0.1$, $\gamma = 0.06$, $\tau = 20$, $n = 2$, and seven linguistic variables.

L_m	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3.0
A	73.97%	74.17%	74.14%	74.20%	**74.61%**	74.16%	74.08%	73.57%

(f) Effect of τ on student's performance, where $\Delta p = 0.04$, $K_{i0} = 0.0$, $K_{d0} = 0.1$, $\gamma = 0.06$, $L_m = 2.1$, $n = 2$, and seven linguistic variables.

γ	1	2	4	8	12	16	20	24
A	71.44%	72.67%	73.76%	74.39%	74.29%	74.40%	**74.61%**	73.91%

(g) Effect of the number of linguistic variables on student's performance, where $\Delta p = 0.04$, $K_{i0} = 0.0$, $K_{d0} = 0.1$, $\gamma = 0.06$, $L_m = 2.1$, $\tau = 20$, $n = 2$.

Number	3	5	7	9	11	13	15	17
A	74.27%	74.25%	**74.61%**	74.14%	74.39%	74.15%	73.59%	73.70%

(h) Effect of n on student's performance, where $\Delta p = 0.04$, $K_{i0} = 0.0$, $K_{d0} = 0.1$, $\gamma = 0.06$, $L_m = 2.1$, $\tau = 20$, and seven linguistic variables.

γ	1	2	3	4	5	6	7	8
A	73.89%	**74.61%**	69.72%	69.70%	70.18%	65.35%	68.32%	63.33%

A indicates the accuracy rate (best results are bold-faced).

improved to make them better suited to our proposed student-centered distillation method.

3) Effect of Teacher Learning on Student Performance: To validate the significance of teachers learning knowledge that aligns with students' desires, we conduct an ablation study on four datasets: CIFAR-100, CINIC-10, SVHN and Tiny-ImageNet. We use these datasets to represent the knowledge that the student desires or does not desire, as shown in Table XII. The red font indicates the knowledge that the student does not want, while the blue font indicates the knowledge that the student wants. For example, in the second row and second column of Table XII, CIFAR-100 is the desired knowledge, and the teacher only learns on CIFAR-100. In contrast, in the second row and third column of Table XII, CIFAR-100 is still the desired knowledge, but the teacher learns on both CIFAR-100 and CINIC-10. Our comparisons of teacher learning across different dataset combinations reveal that when a teacher learn knowledge not desired by a student, it not only diverges from effective knowledge transfer but also impairs the student's performance. In contrast, our SCD method, by focusing on aligning teaching content with student needs, significantly enhances student performance by avoiding the pitfalls of irrelevant knowledge transfer.

4) Training Costs for SCD versus Existing Studies: In this subsection, we provide experiments of SCD with its peers on training costs, as presented in Fig. 5. The reported results for distillation methods that require pre-training of teachers include the respective training costs for both the teacher and student. The figure illustrates the time (in minutes) required for a single epoch of training on the CIFAR-100 training set. All distillation experiments are conducted in the same environment, utilizing an Intel Xeon W-2255@3.70 GHz CPU and a GeForce RTX 3080 GPU. Experimental results demonstrate the efficiency of the proposed method compared to the majority of its peers. The efficiency of SCD, can be attributed to its online distillation approach that eliminates the need for pre-training a teacher. Consequently, the training cost of SCD is significantly lower than that of offline distillation [72]. In particular, SCD, along with the "Lazy" MAML [29], MPL [40], Leap [27], ProKT [31] and IKD [41] methods, which also do not require pre-trained teachers, exhibit lower training costs than other methods that necessitate pre-trained teachers.

E. Qualitative Analysis

This subsection employs feature visual analysis to demonstrate how our SCD method improves student's performance.

1) Visualize the Change Process of w_i: In this paper, we represent the knowledge desired by the student as w_i, where a higher value of w_i indicates a greater desire to learn the knowledge from the corresponding sample. Thus, we illustrate the evolution of w_i for a sample across multiple epochs and provide an explanation of the computation process in Fig. 6. We use the proposed CL-FPID algorithm to evaluate the relative importance w_i of the knowledge from 10 randomly selected validation samples, based on their current loss size L_i and the current state of the student. We provide the changes in w_i values for those 10 validation samples on the 5th and 105th epochs, respectively. Experimental results demonstrate that our proposed method enables a student to focus more on hard sample learning in later stages of training. For example, on the 5th epoch, the knowledge of the 2^{nd}, 6th, 7th, 8th and 10th samples is more valuable to the student than the other ones. However, on the 105th epoch, the student places a higher priority on the knowledge gained from the 5th and 8th samples.

2) Visualize the Working Mechanism of the Proposed CL-Based Fuzzy Strategy: The fuzzy algorithm developed in this paper aims to enable a student to adopt an "easy to difficult" learning approach. To better present it, we added a Fig. 7 to illustrate how K_p, K_i and K_d change with student's performance. We first present the trends in the average loss size of a student on all validation samples in CIFAR-100, CINIC-10, SVHN, and Tiny-ImageNet across epochs, as detailed in Fig. 7(a). Then, we visualize the changes of K_p, K_i and K_d with epoch on CIFAR-100, SVHN and Tiny-ImageNet datasets, respectively, as provided in Fig. 7(b), (c), and (d). Experimental results show

TABLE XI
ABLATION STUDIES OF SCD METHOD

Work	RAF-DB			Oxford-IIIT Pet			FairFace			Food-101		
	A	U	F1	A	U	F1	A	U	F1	A	U	F1
CNN-RIS	85.95%	75.68%	77.97%	50.56%	50.52%	50.20%	67.54%	67.00%	67.13%	54.69%	54.69%	54.57%
CNN-RIS-KD	86.25%	77.58%	79.54%	53.23%	53.18%	52.97%	68.96%	68.46%	68.61%	54.25%	54.25%	54.06%
CNN-RIS-PID	86.15%	77.74%	79.53%	49.44%	49.43%	49.45%	66.78%	66.03%	66.39%	60.04%	60.04%	59.51%
CNN-RIS-PID (Our)	86.54%	77.53%	79.64%	55.06%	55.04%	54.65%	69.07%	68.65%	68.55%	61.73%	61.73%	61.13%
CNN-RIS-CL-FPID	**86.96%**	**78.66%**	**80.54%**	**63.21%**	**63.13%**	**62.83%**	**69.50%**	**69.18%**	**69.14%**	**68.26%**	**68.26%**	**68.01%**

Work	Places-Extra69			CelebA			Logo-2K+			IMDB-WIKI		
	A	U	F1	A	U	F1	A	U	F1	A	U	F1
CNN-RIS	49.38%	47.23%	44.83%	92.48%	66.78%	68.63%	50.08%	39.18%	40.95%	45.20%	37.36%	39.24%
CNN-RIS-KD	49.48%	47.33%	44.43%	92.53%	67.96%	69.66%	50.03%	38.03%	40.42%	46.00%	37.86%	39.79%
CNN-RIS-PID	45.39%	43.42%	38.48%	86.40%	49.42%	50.02%	49.35%	35.32%	38.49%	46.48%	36.50%	39.12%
CNN-RIS-PID (Our)	53.17%	50.86%	48.28%	93.78%	67.08%	70.18%	53.26%	42.51%	44.86%	46.77%	37.99%	40.33%
CNN-RIS-CL-FPID	**55.36%**	**52.96%**	**49.96%**	**93.87%**	**68.47%**	**72.30%**	**54.87%**	**43.60%**	**46.69%**	**46.85%**	**38.82%**	**41.04%**

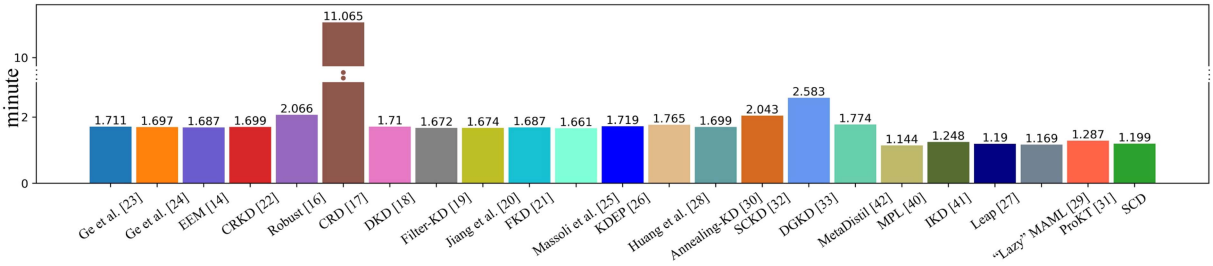


Fig. 5. Number of minutes required for SCD and its peers to train an epoch on the CIFAR-100 training set.

TABLE XII
IMPACT OF TEACHER LEARNING OF STUDENT-UNDESIRED KNOWLEDGE ON STUDENT PERFORMANCE

	CIFAR-100	CINIC-10	SVHN	Tiny-ImageNet
CIFAR-100	**74.61%**	71.05%	72.17%	71.32%
CINIC-10	81.79%	**83.49%**	82.00%	81.95%
SVHN	96.27%	96.44%	**96.81%**	96.25%
Tiny-ImageNet	57.03%	57.47%	57.66%	**61.44%**

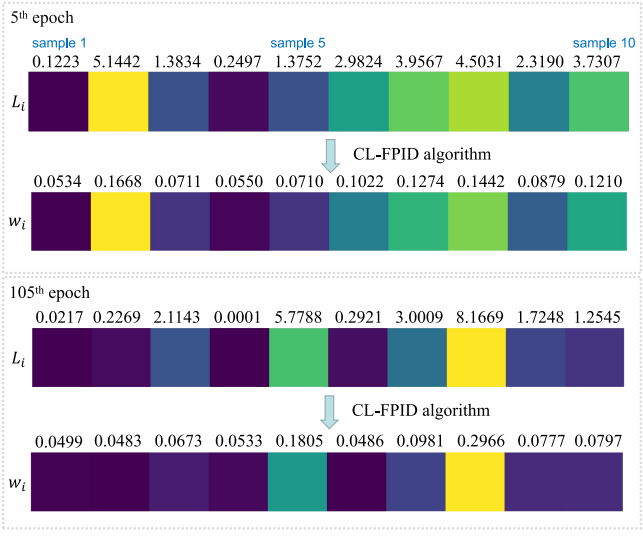
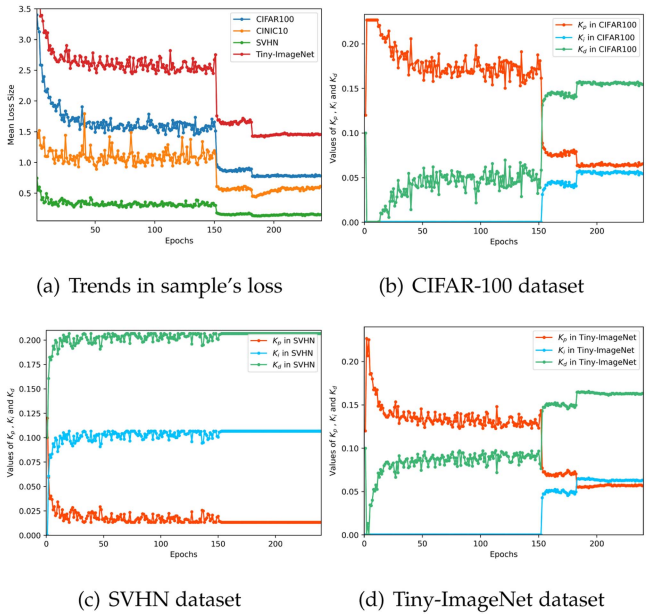


Fig. 6. Visualization of w_i on different epochs.

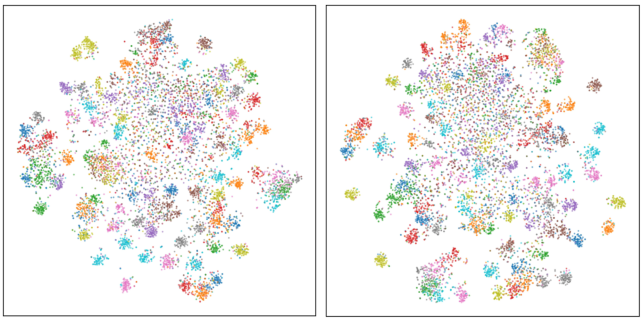


(a) Trends in sample's loss (b) CIFAR-100 dataset

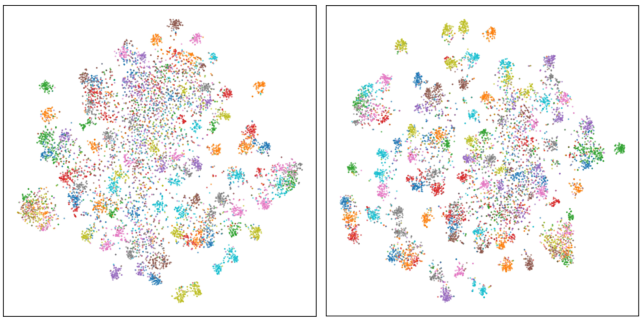
(c) SVHN dataset (d) Tiny-ImageNet dataset

Fig. 7. Visualization the working mechanism of SCD.

that the proposed CL-based fuzzy strategy adaptively adjusts the ratio of proportional, integral, and derivative units according to the student's performance. For example, when the performance of the student is poor, K_p has a higher value, while K_i and K_d have lower values. As the performance of the student improves, K_p decreases, while K_i and K_d increase. Furthermore, for easily identifiable datasets, such as SVHN, the values of K_i and K_d are typically higher than in more challenging datasets, such as Tiny-ImageNet. This is because there are fewer hard samples in the former compared to the latter, which allows for a more focused effort on mining knowledge from those challenging samples.



(a) MPL [40] (b) IKD [41]



(c) MetaDistil [42] (d) SCD (Our)

Fig. 8. Visualization of student's performance using T-SNE.

3) Visualize the Student's Features Generated by Proposed Method and Existing Studies: Section II-B describes three existing studies [40], [41], [42] that use the validation set to evaluate student's performance, similar to our proposed method. However, because a student and teacher have distinct network structures, updating the latter with the gradients of the former is impractical. To clearly prove that the gradient information of the student could not be used to update the teacher, we used T-SNE [73] to visualize the features in the student generated by the proposed method and the work [40], [41], [42], as detailed in Fig. 8. The figure uses CIFAR-100 test set with 10,000 samples to show the performance of a distillation method on the mimic layer. Experimental results indicate that the student's gradient is not suitable for updating a teacher. We can only identify the knowledge desired by a student through validation sets, for informing a teacher to learn and transfer that knowledge to the student.

VI. Conclusion

Knowledge Distillation is a machine learning approach that emulates the human teaching and learning process, utilizing a high-performing teacher to guide the training of a student. Current methodologies predominantly favor a teacher-centered approach, which might relay redundant or undesired knowledge to the student. This paper introduces a student-centered knowledge distillation strategy grounded in the human educational wisdom [74]. Specifically, the teacher first pinpoints the areas where the student might falter using a validation set and then systematically bolsters these areas through the training set. By doing so, we harness the full capacity of the teacher, not just to relay its existing knowledge, but to elevate the student's performance. Extensive experiments validate that our method surpasses prevailing teacher-centered techniques.

In the realm of artificial intelligence, the behaviors of animals, particularly their intelligent traits, have been the subject of meticulous study. This has given rise to a vast array of applications, especially within swarm intelligence [75]. Yet, the complexity of human intelligent behavior offers even deeper insights. Though human intelligence shares some parallels with swarm intelligence, it also branches into profound emotional, creative, and advanced cognitive facets. Thus, to truly advance the frontier of artificial intelligence, it is pivotal to delve into and emulate the intelligence of human behavior. This paper highlights the potential for innovation at the intersection of engineering and the humanities by emulating human intelligent behavior.

References

[1] G. Hinton et al., "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.

[2] L. Wang and K.-J. Yoon, "Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3048–3068, Jun. 2022.

[3] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *Int. J. Comput. Vis.*, vol. 129, no. 6, pp. 1789–1819, 2021.

[4] Z. Bao, S. Yang, Z. Huang, M. Zhou, and Y. Chen, "A lightweight block with information flow enhancement for convolutional neural networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 3570–3584, Aug. 2023.

[5] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4320–4328.

[6] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–13.

[7] U. Abdigapbarova and N. Zhiyenbayeva, "Organization of student-centered learning within the professional training of a future teacher in a digital environment," *Educ. Inf. Technol.*, vol. 28, no. 1, pp. 647–661, 2023.

[8] X. Zhang, B. Zhang, and F. Zhang, "Student-centered case-based teaching and online–offline case discussion in postgraduate courses of computer science," *Int. J. Educ. Technol. High. Educ.*, vol. 20, no. 1, 2023, Art. no. 6.

[9] G. Xu et al., "Evaluating the effectiveness of a new student-centred laboratory training strategy in clinical biochemistry teaching," *BMC Med. Educ.*, vol. 23, no. 1, Art. no. 391, 2023.

[10] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2935–2947, Dec. 2018.

[11] H. Shao et al., "ControlVAE: Tuning, analytical properties, and performance analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 9285–9297, Dec. 2022.

[12] H. Wu, X. Luo, M. Zhou, M. J. Rawa, K. Sedraoui, and A. Albeshri, "A pid-incorporated latent factorization of tensors approach to dynamically weighted directed network analysis," *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 3, pp. 533–546, Mar. 2022.

[13] X. Wang, Y. Chen, and W. Zhu, "A survey on curriculum learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 4555–4576, Sep. 2022.

[14] H. Li, N. Wang, X. Ding, X. Yang, and X. Gao, "Adaptively learning facial expression representation via CF labels and distillation," *IEEE Trans. Image Process.*, vol. 30, pp. 2016–2028, 2021.

[15] S. Yang, L. Xu, M. Zhou, X. Yang, J. Yang, and Z. Huang, "Skill-transferring knowledge distillation method," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 11, pp. 6487–6502, Nov. 2023.

[16] T. Guo, C. Xu, S. He, B. Shi, C. Xu, and D. Tao, "Robust student network learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 7, pp. 2455–2468, Jul. 2020.

[17] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–19.

[18] B. Zhao, Q. Cui, R. Song, Y. Qiu, and J. Liang, "Decoupled knowledge distillation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 11953–11962.

[19] Y. Ren, S. Guo, and D. J. Sutherland, "Better supervisory signals by observing learning paths," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–24.

[20] H. Jiang, H. Narasimhan, D. Bahri, A. Cotter, and A. Rostamizadeh, "Churn reduction via distillation," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–53.

[21] B. He and M. Ozay, "Feature kernel distillation," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–38.

[22] Z. Feng, J. Lai, and X. Xie, "Resolution-aware knowledge distillation for efficient inference," *IEEE Trans. Image Process.*, vol. 30, pp. 6985–6996, 2021.

[23] S. Ge, S. Zhao, C. Li, and J. Li, "Low-resolution face recognition in the wild via selective knowledge distillation," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 2051–2062, Apr. 2019.

[24] S. Ge, S. Zhao, C. Li, Y. Zhang, and J. Li, "Efficient low-resolution face recognition via bridge distillation," *IEEE Trans. Image Process.*, vol. 29, pp. 6898–6908, 2020.

[25] F. V. Massoli, G. Amato, and F. Falchi, "Cross-resolution learning for face recognition," *Image Vis. Comput.*, vol. 99, 2020, Art. no. 103927.

[26] R. He, S. Sun, J. Yang, S. Bai, and X. Qi, "Knowledge distillation as efficient pre-training: Faster convergence, higher data-efficiency, and better transferability," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–13.

[27] S. Flennerhag, P. G. Moreno, N. D. Lawrence, and A. Damianou, "Transferring knowledge across learning processes," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–23.

[28] Y. Huang, J. Wu, X. Xu, and S. Ding, "Evaluation-oriented knowledge distillation for deep face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 18740–18749.

[29] M. A. Jamal, L. Wang, and B. Gong, "A lazy approach to long-horizon gradient-based meta-learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 6577–6586.

[30] A. Jafari, M. Rezagholizadeh, P. Sharma, and A. Ghodsi, "Annealing knowledge distillation," in *Proc. Eur. Assocn. Comput. Linguist.*, 2021, pp. 2493–2504.

[31] W. Shi, Y. Song, H. Zhou, B. Li, and L. Li, "Follow your path: A progressive method for knowledge distillation," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2021, pp. 596–611.

[32] Y. Zhu and Y. Wang, "Student customized knowledge distillation: Bridging the gap between student and teacher," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 5057–5066.

[33] W. Son, J. Na, J. Choi, and W. Hwang, "Densely guided knowledge distillation using multiple teacher assistants," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 9395–9404.

[34] X. He and Y. Hong, "The effect of augmented reality on the memorization in history and humanities education," in *Proc. Int. Conf. Appl. Hum. Factors Ergonom.*, 2020, pp. 769–776.

[35] Q. Guo et al., "Online knowledge distillation via collaborative learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11020–11029.

[36] G. Wu and S. Gong, "Peer collaborative learning for online knowledge distillation," in *Proc. Assoc. Adv. Artif. Intell.*, 2021, pp. 10302–10 310.

[37] I. Chung, S. Park, J. Kim, and N. Kwak, "Feature-map-level online adversarial knowledge distillation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 2006–2015.

[38] J. Gou, L. Sun, B. Yu, L. Du, K. Ramamohanarao, and D. Tao, "Collaborative knowledge distillation via multiknowledge transfer," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2022.3212733.

[39] C. Yang, Z. An, H. Zhou, F. Zhuang, Y. Xu, and Q. Zhang, "Online knowledge distillation via mutual contrastive learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 10212–10227, Aug. 2023.

[40] H. Pham, Z. Dai, Q. Xie, and Q. V. Le, "Meta pseudo labels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11557–11568.

[41] Y. Liu, T. Sun, X. Qiu, and X. Huang, "Learning to teach with student feedback," 2021, *arXiv:2109.04641*.

[42] W. Zhou, C. Xu, and J. McAuley, "BERT learns to teach: Knowledge distillation with meta learning," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2022, pp. 7037–7049.

[43] R. Ma, S. Li, B. Zhang, and H. Hu, "Meta PID attention network for flexible and efficient real-world noisy image denoising," *IEEE Trans. Image Process.*, vol. 31, pp. 2053–2066, 2022.

[44] P. Du, Y. Liu, W. Chen, S. Zhang, and J. Deng, "Fast and precise control for the vibration amplitude of an ultrasonic transducer based on fuzzy PID control," *IEEE Trans. Ultrason. Ferroelect. Freq. Control*, vol. 68, no. 8, pp. 2766–2774, Aug. 2021.

[45] Y. Tian, Z. Cao, D. Hu, X. Gao, L. Xu, and W. Yang, "A fuzzy PID-controlled iterative Calderon's method for binary distribution in electrical capacitance tomography," *IEEE Trans. Instrum. Meas.*, vol. 70, 2021, Art. no. 4502911.

[46] P. K. Ray et al., "A hybrid firefly-swarm optimized fractional order interval type-2 fuzzy PID-PSS for transient stability improvement," *IEEE Trans. Ind. Appl.*, vol. 55, no. 6, pp. 6486–6498, Nov./Dec. 2019.

[47] X. Jin, K. Chen, Y. Zhao, J. Ji, and P. Jing, "Simulation of hydraulic transplanting robot control system based on fuzzy PID controller," *Measurement*, vol. 164, 2020, Art. no. 108023.

[48] S. Li and W. Deng, "Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 356–370, Jan. 2019.

[49] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar, "Cats and dogs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3498–3505.

[50] K. Karkkainen and J. Joo, "FairFace: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation," in *Proc. IEEE/CVF Winter Conf. App. Comput. Vis.*, 2021, pp. 1548–1558.

[51] L. Bossard, M. Guillaumin, and L. V. Gool, "Food-101–mining discriminative components with random forests," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 446–461.

[52] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, Jun. 2018.

[53] J. Wang et al., "Logo-2k : A large-scale logo dataset for scalable logo classification," in *Proc. Assoc. Adv. Artif. Intell.*, vol. 34, no. 04, 2020, pp. 6194–6201.

[54] R. Rothe, R. Timofte, and L. Van Gool, "Deep expectation of real and apparent age from a single image without facial landmarks," *Int. J. Comput. Vis.*, vol. 126, no. 2, pp. 144–157, 2018.

[55] Z. Huang, S. Yang, M. C. Zhou, Z. Li, Z. Gong, and Y. Chen, "Feature map distillation of thin nets for low-resolution object recognition," *IEEE Trans. Image Process.*, vol. 31, pp. 1364–1379, 2022.

[56] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3730–3738.

[57] A. Krizhevsky et al., "Learning multiple layers of features from tiny images," 2009.

[58] L. N. Darlow, E. J. Crowley, A. Antoniou, and A. J. Storkey, "Cinic-10 is not ImageNet or CIFAR-10," 2018, *arXiv:1810.03505*.

[59] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*2011, 2011, pp. 1–9.

[60] Y. Le and X. Yang, "Tiny ImageNet visual recognition challenge," *CS 231 N*, vol. 7, no. 7, 2015, Art. no. 3.

[61] A. K. Menon, A. S. Rawat, S. Reddi, S. Kim, and S. Kumar, "A statistical perspective on distillation," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 7632–7642.

[62] J. H. Cho and B. Hariharan, "On the efficacy of knowledge distillation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 4794–4802.

[63] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "FaceBoxes: A CPU real-time face detector with high accuracy," in *Proc. IEEE Int. Joint Conf. Biometrics*, 2017, pp. 1–9.

[64] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "Wider face: A face detection benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5525–5533.

[65] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2879–2886.

[66] J. Yan, X. Zhang, Z. Lei, and S. Z. Li, "Face detection by structural models," *Image Vis. Comput.*, vol. 32, no. 10, pp. 790–799, 2014.

[67] S. Yang, P. Luo, C. C. Loy, and X. Tang, "Faceness-net: Face detection through deep facial part responses," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1845–1859, Aug. 2018.

[68] B. Maze et al., "IARPA Janus benchmark-C: Face dataset and protocol," in *Proc. Int. Conf. Biometrics*, 2018, pp. 158–165.

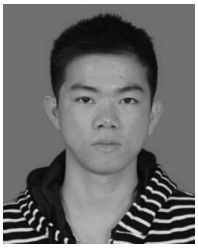
[69] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Proc. Workshop Faces Real-Life'Images: Detection Alignment Recognit.*, 2008, pp. 1–15.

[70] M. Grgic, K. Delac, and S. Grgic, "Scface–surveillance cameras face database," *Multimedia Tools Appl.*, vol. 51, pp. 863–879, 2011.

[71] Z. Cheng, X. Zhu, and S. Gong, "Low-resolution face recognition," in *Proc. Asian Conf. Comput. Vis.*, 2019, pp. 605–621.

[72] X. Zhu et al., "Knowledge distillation by on-the-fly native ensemble," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–11.

[73] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.
[74] C. C. Johnson, J. B. Walton, L. Strickler, and J. B. Elliott, "Online teaching in k-12 education in the United States: A systematic review," *Rev. Educ. Res.*, vol. 93, no. 3, pp. 353–411, 2023.
[75] J. Tang, H. Duan, and S. Lao, "Swarm intelligence algorithms for multiple unmanned aerial vehicles collaboration: A comprehensive review," *Artif. Intell. Rev.*, vol. 56, no. 5, pp. 4295–4327, 2023.



Shunzhi Yang received the senior college degree from Shenzhen Polytechnic University, in 2015, the BS degree from Hanshan Normal University, in 2017, the MS degree in 2020, and the PhD degree in 2023, all from South China Normal University. He is now an assistant researcher in Shenzhen Polytechnic University. His research focuses on computer vision, knowledge distillation and low-resolution object recognition.

Jinfeng Yang received the BE degree in mechanics from Zhengzhou University of Light Industry, in 1994, the MSc degree in hydrokinetics from Zhengzhou University, in 2001, and the PhD degree in patter recognition and intelligent systems from National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Science, China, in 2005. He has published more than 100 research papers in the areas of computer vision and pattern recognition.

MengChu Zhou (Fellow, IEEE) received the PhD degree from Rensselaer Polytechnic Institute, Troy, NY, in 1990 and then joined New Jersey Institute of Technology where he is now a distinguished professor. His interests are Petri nets, automation, Internet of Things, and Big Data. He has more than 900 publications including 12 books, 600+ journal papers (500+ in IEEE transactions), 29 patents and 29 book-chapters. He is Fellow of IFAC, AAAS, CAA and NAI.

Zhenhua Huang received the PhD degree in the computer science from Fudan University, Shanghai, China, in 2008. He is currently a professor in the School of Computer Science with South China Normal University, Guangzhou, China. His research interests mainly include deep learning, Internet of Things, recommendation system, data mining, knowledge discovery, and Big Data.

Wei-Shi Zheng (Member, IEEE) received the PhD degree in applied mathematics from Sun Yat-sen University, in 2008. He is now a full professor with Sun Yat-sen University. His research interests include person/object association and activity understanding in visual surveillance, and the related large-scale machine learning algorithm. He has ever joined Microsoft Research Asia Young Faculty Visiting Programme. He has ever served as area chairs of CVPR, ICCV, BMVC, and IJCAI.

Xiong Yang received the BS degree in electronic information engineering from Civil Aviation University of China, in 2018, and the MSc degree from Civil Aviation University of China, in 2021. He is a research assistant with the Institute of Applied Artificial Intelligence of the Guangdong-Hong Kong-Macao Greater Bay Area, Shenzhen Polytechnic University. His research interests include machine learning, computer vision, and data mining.

Jin Ren received the BS degree from Central South University, in 2013, and the PhD degree in mechanical engineering from Central South University, in 2020. He is currently working with Institute of Applied Artificial Intelligence of the Guangdong-Hong Kong-Macao Greater Bay Area, Shenzhen Polytechnic University. His research interests include speech and language processing, semantic perception, and knowledge distillation.