computers

MDPI

Article

Knowledge Distillation in Image Classification: The Impact of Datasets

Ange Gabriel Belinga [1,*], Cédric Stéphane Tekouabou Koumetio [2,3,*], Mohamed El Haziti [1,4] and Mohammed El Hassouni [5]

[1] Laboratory of Research in Computer Science and Telecommunications (LRIT), Faculty of Sciences in Rabat, Mohammed V University in Rabat, Rabat 10000, Morocco
[2] Laboratory in Computer Science and Educational Technologies (LITE), Higher Teacher Training College (HTTC), University of Yaoundé 1, Yaounde P.O. Box 47, Cameroon
[3] Department of Computer Science and Educational Technologies (DITE), Higher Teacher Training College (HTTC), University of Yaoundé 1, Yaounde P.O. Box 47, Cameroon
[4] High School of Technology, Mohammed V University in Rabat, Sale 11000, Morocco; elhazitim@gmail.com
[5] FLSH, Mohammed V University in Rabat, 3 Av. Ibn Batouta, Rabat 10090, Morocco; mohamed.elhassouni@flsh.um5.ac.ma
* Correspondence: agbelinga@gmail.com (A.G.B.); ctekouaboukoumetio@gmail.com (C.S.T.K.); Tel.: +212-771-515-963 or +237-697-64-69-78 (C.S.T.K.)

Abstract: As the demand for efficient and lightweight models in image classification grows, knowledge distillation has emerged as a promising technique to transfer expertise from complex teacher models to simpler student models. However, the efficacy of knowledge distillation is intricately linked to the choice of datasets used during training. Datasets are pivotal in shaping a model's learning process, influencing its ability to generalize and discriminate between diverse patterns. While considerable research has independently explored knowledge distillation and image classification, a comprehensive understanding of how different datasets impact knowledge distillation remains a critical gap. This study systematically investigates the impact of diverse datasets on knowledge distillation in image classification. By varying dataset characteristics such as size, domain specificity, and inherent biases, we aim to unravel the nuanced relationship between datasets and the efficacy of knowledge transfer. Our experiments employ a range of datasets to comprehensively explore their impact on the performance gains achieved through knowledge distillation. This study contributes valuable guidance for researchers and practitioners seeking to optimize image classification models through kno-featured applications. By elucidating the intricate interplay between dataset characteristics and knowledge distillation outcomes, our findings empower the community to make informed decisions when selecting datasets, ultimately advancing the field toward more robust and efficient model development.

Keywords: knowledge distillation; dataset selection; transfer learning; knowledge transfer

Citation: Belinga, A.G.; Tekouabou Koumetio, C.S.; El Haziti, M.; El Hassouni, M. Knowledge Distillation in Image Classification: The Impact of Datasets. *Computers* **2024**, *13*, 184. https://doi.org/10.3390/computers13080184

Academic Editors: Jeng-Shyang Pan, Junzo Watada, Vaclav Snasel and Pei Hu

Received: 27 May 2024
Revised: 15 July 2024
Accepted: 18 July 2024
Published: 24 July 2024

1. Introduction

In the ever-evolving landscape of computer vision, image classification is a fundamental and challenging task with many applications [1–4]. In the recent literature, machine learning (ML) models based on deep neural networks (DNNs) have proven to be the most effective for computer vision, particularly image analysis [5–7]. To achieve this efficiency, several DNN architectures have been proposed in the literature, with different processes including knowledge distillation [8–10]. Knowledge distillation in deep neural networks is a crucial process in the ML field [11]. As the demand for more efficient and lightweight models grows, the concept of knowledge distillation (KD) has emerged as a promising avenue to transfer knowledge from complex, high-capacity models (teachers) to simpler, more deployable counterparts (students) [8,12]. This transfer of knowledge from the teacher to the student through a training paradigm typically involves the following steps.

1. Teacher model training: The first step is to train a large and complex model (the teacher) on a given dataset to achieve high accuracy.
2. Generation of soft targets: The trained teacher model is then used to make predictions on the training data, producing probability distributions (soft targets) over possible classes. These soft targets contain more information than the hard targets (i.e., the actual labels), as they reflect the relative confidence of the teacher model in its predictions. The soft targets can be obtained using a sofmax function

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_i/T)} \tag{1}$$

where q_i is the output probability for class i, z_i is the logit for class i, and T is the temperature parameter.
3. Student model training: The smaller student model is trained using a combination of the original true labels and the soft targets generated by the teacher model. The loss function typically includes a component for standard classification loss and another component for distillation loss, which measures the difference between the student and teacher probability distributions. The Kullback–Leibler (KL) function is usually used for distillation loss. The KL formula is defined as

$$D_{KL}(P \parallel Q) = \sum_{x \in X} P(x) \log(\frac{P(x)}{Q(x)}) \tag{2}$$

where P and Q are probability distributions defined on the same sample space X. The final loss formula is defined as

$$L_{total} = \alpha L_{classification} + (1 - \alpha) L_{distillation} \tag{3}$$

where $L_{classification} = \sum_i y_i log(p_i)$ and $L_{distillation} = KL(q_teacher^T \parallel q_student^T)$.

Indeed, this approach makes it possible to compress and generalize the information learned by complex deep neural networks, facilitating their deployment on resource-limited devices [8,13]. This process of KD not only facilitates model compression but also enhances the generalization capabilities of the student model [10]. The success of KD is inherently tied to the quality and diversity of the datasets used during the training step, as well as the large applications of the KD learning-based processes [1,12,14–19].

The effectiveness of KD in DNN could depend largely on the complexity (quality and quantity, etc.) of the data used. Thus, datasets play a pivotal role in shaping the learning process, influencing the model's ability to discern patterns and generalize to unseen features [15–17]. While extensive research has been conducted on KD and image classification independently, a comprehensive understanding of how various datasets impact the effectiveness of KD remains an open and critical area of investigation. However, although many studies have been published on this method, few have explored in depth how the characteristics and properties of the data would influence this knowledge transfer process. This research gap raises a crucial question: How do data characteristics, such as complexity, diversity, and distribution, impact the efficiency of KD in a deep neural network? Answering this question will enable us to better understand the challenges and opportunities for KD applications related to the use of different data sources, paving the way for more efficient and robust techniques for transfer learning in deep neural networks.

This study seeks to address this gap by systematically examining the impact of different datasets on KD in image classification. As datasets vary in terms of size, domain specificity, and inherent biases, their influence on the transfer of knowledge from teacher to student models warrants meticulous exploration. Through a series of experiments, we aim to unravel the intricate relationship between dataset characteristics and the performance gains achieved through knowledge distillation. In the subsequent sections, we delve into the relevant literature, providing insights into the existing landscape of KD and its appli-

cation in image classification. Following this, we elucidate our methodology, detailing the datasets chosen for experimentation, model architectures, and the KD process. The results and their implications are then discussed, shedding light on the nuanced impact of datasets. Ultimately, this study aims to contribute valuable insights for researchers and practitioners navigating the intersection of knowledge distillation and image classification, offering guidance on optimizing model performance through judicious dataset selection.

Following on from the remainder of this work, Section 2 will discuss previous work on knowledge distillation in deep neural networks. Then, Sections 3 and 4 will describe the proposed research approach and analyze the obtained results, respectively. Finally, Sections 5 and 6 will discuss the results obtained and conclude this work.

2. Related Work

Knowledge distillation (KD) has been widely studied in the literature, and several notable works have contributed to the understanding and development of this technique [20–25]. Since its introduction by Hinton et al. [8], this approach has attracted growing interest in the machine learning research community. Table 1 presents some recents knowledge distillation work in the field of image classification. This table mainly presents the different databases, the architecture of the teacher and student models and the main evaluation metric used to perform KD in image classification task.

2.1. Knowledge Distillation in the Literature

Several works have explored various aspects of knowledge distillation in deep neural networks [26], including teacher and student model architectures, regularization techniques, and optimization methods.

For example, Li et al. proposed a transferred attention method to improve the performance of convolutional neural networks [27], while Yazdanbakhsh et al. studied the application of knowledge distillation in specific domains such as healthcare [19]. However, despite these significant advances, little attention has been paid to the impact of data on this knowledge transfer process.

Table 1. Summary of recent literature on knowledge distillation in image classification. EM = evaluation metric.

-		About Data		Methods			EM
Ref	**Year**	**Type**	**Dataset**	**Type**	**Teacher**	**Student**	**Acc**
LightCyan [8]	2015	article	JFT,MNIST	Images	DNN	DNN	✓
[28]	2019	article	PPMI, Willow, UIUC-Sport	Images	AlexNet	AlexNet	✓
LightCyan [29]	2021	article		Images	DNN	DNN	✓
[30]	2017	Conf	CIFAR100	Images	VGG13, ResNet32x4	VGG13, ResNet32x4	✓
LightCyan [31]	2018	Conf	CIFAR100	Images	ResNet34	VGG9	✓
[32]	2020	Conf	CIFAR10, CIFAR100	Images	ResNet26	ResNet8&14	✓
LightCyan [20]	2014	article	CIFAR10, CIFAR100, SVHN, MNIST, AFLW	Images		FiNet	✓
[33]	2021	article	CIFAR100	Images	ResNet20	ResNet8	✓
LightCyan [34]	2024	article	CIFAR100, ImageNet	Images	ResNet152	ResNet18	✓
[35]	2023	article	CIFAR100	Images	ResNet50& ResNet34	ResNet18	✓
LightCyan [36]	2023	article	CIFAR100	Images	ResNet18	ResNet18	✓

The authors demonstrated the effectiveness of the distillation on various tasks and highlighted its potential for model compression. The FitNets paper [20] proposed a specific form of knowledge distillation called FitNets, where a student network is guided not only by the output probabilities of a teacher network but also by intermediate representations

(or hints). This work aimed to improve the transfer of information in the training process. Ref. [27] introduces attention transfer as a form of knowledge distillation. It focuses on transferring attention maps from a teacher to a student network to improve the student's performance. Attention transfer has proven effective in enhancing the generalization capabilities of the student model. To address the limitations of traditional knowledge distillation, ref. [31] introduces Jacobian matching, a novel method that aims to transfer not only the output probabilities but also the derivatives of the teacher model's predictions. This approach provides a more comprehensive form of knowledge transfer. Ref. [30] explores the benefits of knowledge distillation beyond model compression. The authors show that the knowledge distillation process not only compresses models but also accelerates the optimization process, enabling faster convergence during training. Ref. [32] introduces the concept of a "teacher assistant" by proposing an extension to traditional knowledge distillation. The teacher assistant helps bridge the performance gap between the teacher and the student, leading to enhanced knowledge transfer.

2.2. Role of Datasets for Model Training by KD

The impact of datasets on model training has been a longstanding focus in machine learning research. Datasets serve as the foundation upon which models learn to recognize and classify patterns, making their composition and characteristics crucial determinants of model performance. Studies by refs. [37,38] emphasize the importance of diverse datasets in fostering robust image recognition systems, highlighting how exposure to a wide range of scenarios aids in generalization. In the context of image classification, biases present in datasets have been identified as potential challenges, leading to models that may not generalize well across different domains [37]. Addressing these biases and ensuring dataset diversity are pivotal considerations in the pursuit of building models that can perform reliably across various real-world scenarios.

2.3. Research Gap and Motivation

While the individual importance of KD and dataset characteristics in image classification has been adequately explored, a comprehensive examination of how different datasets impact the success of KD remains a notable gap in the literature. Synthesizing the existing literature, we recognize the intertwined nature of knowledge distillation and dataset influence on image classification models. Furthermore, the literature review confirms the preliminary observation that several works have studied knowledge distillation in neural networks [8,28,29,31,32,36]. However, the majority of these studies have used not only a single dataset (CIFAR10, CIFAR100, MNIST, ImageNet, etc.) [20,30–32,34] but also, more often than not, residual network architectures (ResNet) [30,32–35]. Moreover, knowledge acquisition is relative to the context, which is nothing other than the data, whereas the existing studies often focus on benchmark datasets without thoroughly investigating the nuances introduced by varying dataset characteristics.

This study aims to bridge this gap by systematically exploring the relationship between dataset properties and the efficacy of knowledge distillation. Successful knowledge transfer relies not only on the distillation techniques but also on the inherent properties of the datasets used during training. In the subsequent sections, we detail our methodology, experimentally addressing this critical gap and shedding light on how different datasets impact the performance of knowledge-distilled models.

3. Research Method

The research approach adopted in this paper aims to highlight the impact of data complexity on knowledge distillation in deep convolutional neural networks. To better illustrate this approach, we have represented its operating process on the diagram in Figure 1, which gives a better overview of the different steps of the followed method.

From this illustration, the first step in our approach is to select the databases most commonly used in the literature (see analysis in Section 2.2), which will enable us to carry

out our study, as detailed in the following Section 3.1. Once the databases have been selected, the next step is to choose the architectures of the teaching and learning neural networks with which to test our approach. Once the architectures have been chosen, this stage, which we detail in Section 3.2, ends with training the parent model and an instance of the student model from scratch on all the experiments' datasets. Then, the third stage of our experiment consists of simulating the trained student models through knowledge distillation according to two configurations, namely response-based distillation (RKD) and intermediate-based distillation (IKD), which we explain in Section 3.3. Finally, the fourth and last stage of our study consists of comparing the results and seeing the effect of different databases on knowledge distillation.

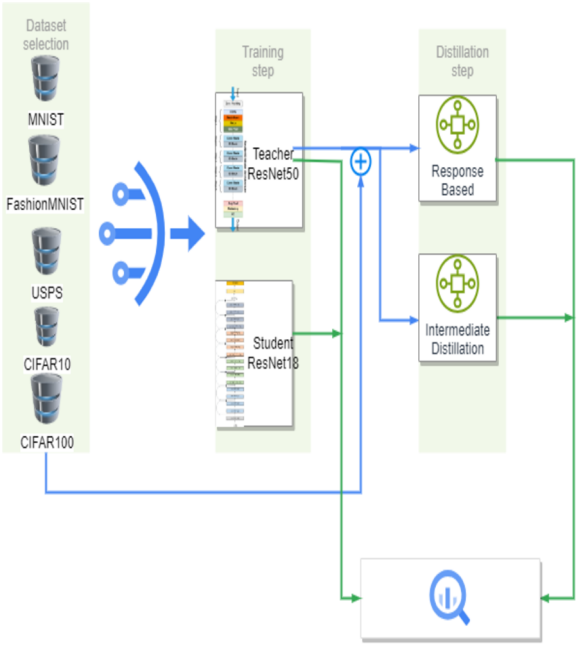


Figure 1. Flowchart of the proposed approach to highlight the impact of the dataset on knowledge distillation in DNN.

3.1. Datasets Selection

To comprehensively investigate the impact of datasets on knowledge distillation in image classification, a diverse set of datasets is curated. The selection criteria include considerations of size, domain specificity, and potential biases. Well-established benchmark datasets, such as CIFAR-10, CIFAR-100, and MNIST as shown in Table 1, form the core of our study, providing a foundation for cross-dataset comparisons.

Dataset Description and Complexity Classification

To highlight the impact of datasets on the distillation of knowledge learned by deep neural networks, we tested teacher and student network architectures on the most popular datasets in the scientific machine learning literature. For this purpose, we used 5 different data sets, including MNIST [6], FashionMNIST [7], UPS [11], CIFAR10 and CIFAR100 [5], which are summarized in Table 2 and described in turn in the rest of this section.

Each dataset was selected to represent different characteristics and complexities, ensuring a comprehensive evaluation of the distillation process. The classification of the level of data complexity in this article is based on a combined analysis of the dataset's characteristics (dimensionality, class diversity, data volume, variability and domain specificity) and the performances obtained in the literature [39,40]. Below are descriptions of the datasets mentioned in the literature review for knowledge distillation in image classification:

Table 2. The key statistics for each dataset.

Dataset	Image Sizes	Nb Classes	Nb-Images	Complexity Level
CIFAR-10	32 × 32	10	60,000	Moderate to high
CIFAR-100	32 × 32	100	60,000	High
USPS	16 × 16	10	9298	Moderate
MNIST	28 × 28	10	60,000	Low
Fashion MNIST	28 × 28	10	60,000	Low to moderate

- CIFAR-10 [5]: This dataset consists of 60,000 32 × 32 color images across ten different classes, each containing 6000 images. The classes include common objects like cars, dogs, and cats. The addition of color and more diverse objects increases the complexity compared to MNIST and USPS. Criteria: larger image size (32 × 32 pixels), three-channel color images, more diverse classes, and significant background variations.
- CIFAR-100 [5]: Similar to CIFAR-10, CIFAR-100 has 100 classes, with 600 images per class. It covers a broader range of object categories, making it more challenging. The increased number of classes and the finer distinctions between categories make it a more complex classification task compared to the previous datasets. Criteria: same image size (32 × 32 pixels) and color channels as CIFAR10, but a much larger number of classes (100), increasing variability and the challenge of classification.
- USPS [11] is a digit dataset automatically scanned from envelopes by the U.S. Postal Service containing a total of 9298 16 × 16 pixel grayscale samples; the images are centered and normalized and show a broad range of font styles. Similar to MNIST, USPS contains images of handwritten digits. It is slightly more challenging than MNIST but still relatively simple. Criteria: small image size (16 × 16 pixels), same number of classes (10 digits), and slight variations in style and noise compared to MNIST.
- MNIST [41] is a dataset with 28 × 28 grayscale images of handwritten digits. It consists of ten different classes and is often used for image classification tasks. The dataset is relatively simple and is often used as a beginner's dataset for image classification tasks. Criteria: small image size (28 × 28 pixels), a limited number of classes (10 digits), simple and uniform structure with minimal noise.
- Fashion MNIST [7] is a dataset with 28 × 28 grayscale images of fashion items, such as clothing and accessories. It consists of ten different classes and is often used as a replacement for the traditional MNIST dataset for image classification tasks. The dataset is more complex than MNIST as it requires the model to recognize various types of clothing items, adding a bit more complexity to the classification task. Criteria: same image size (28 × 28 pixels) as MNIST, but with 10 different classes of clothing, introducing more variability in shapes, and textures.

The levels of complexity of the datasets were determined according to several key criteria, which include the following:

- Dimensionality: the resolution and color channels of the images. higher resolution and multiple colour channels generally increase the complexity of the dataset, as they require more sophisticated models to capture detail.
- Class diversity: the number and variability of classes within the dataset. A larger number of classes with significant differences between them increases complexity because the model has to distinguish between a larger set of categories.
- Data volume: the size of the dataset in terms of the number of samples. Larger datasets can be more complex to manage and require more computing resources, but they also provide more information for robust model formation.
- Variability: the level of noise, background variation, and object diversity within the dataset. Datasets with high variability in object appearance, backgrounds, and noise levels are more difficult for models to learn and generalize.

- Domain specificity: the within-domain specificity and variability of the dataset (e.g., handwritten figures versus real-world objects). Datasets from domains with high intra-class variability and inter-class similarity are considered more complex due to the more subtle distinctions that need to be learned.

The complexity increases from MNIST and USPS to FashionMNIST, CIFAR-10, and finally CIFAR-100, with the latter being the most challenging among the mentioned datasets for an image classification task using the ResNet architecture.

3.2. Model Architecture Details

Our experimental setup involves employing state-of-the-art model architectures as both teacher and student networks. Convolutional neural networks (CNNs) [42,43] have demonstrated exceptional performance in image classification tasks [44,45], and we leverage ResNet [46] architectures for our experiments. Table 1 shows the frequency of use of ResNet in the literature. The teacher model, being more complex, serves as the knowledge source, while the student model is designed with fewer parameters to facilitate efficient deployment.

ResNet, introduced by ref. [46], has become a pivotal architecture in deep learning due to its ability to tackle the vanishing gradient problem through the innovative use of residual connections [14].

The key innovation of ResNet lies in the use of residual blocks (Figure 2), where each block contains a shortcut connection that bypasses one or more convolutional layers. This shortcut connection enables the network to learn residual mappings, making it easier to optimize deeper architectures. ResNet architectures come in various depths, such as ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152 [46], each with a different number of layers. The following Table 3 shows the characteristics of the models used in our experiment.

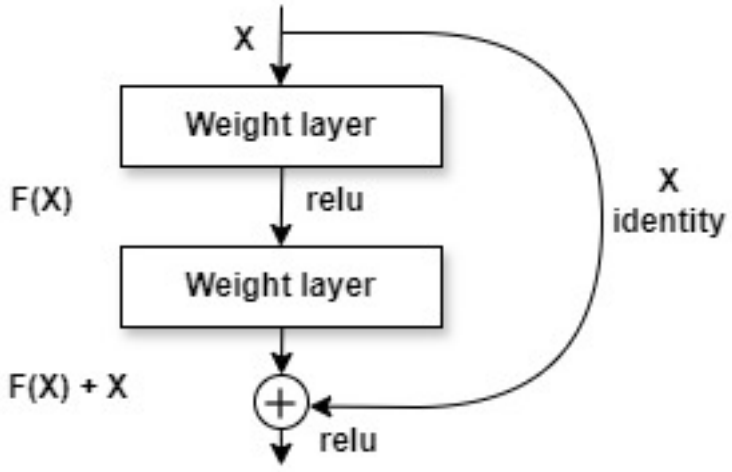


Figure 2. Residual blocks of ResNet architecture [46].

This table shows the key details of the ResNet-50 and ResNet-18 architectures [46] used in our experiments. The Bottleneck layers (ResNet-50) consist of three layers (1×1 convolution for channel reduction, 3×3 convolution, and 1×1 convolution for channel restoration), optimizing network efficiency and depth, and the Basic Unit layers (ResNet-18) consist of two 3×3 convolution layers, maintaining simplicity and reducing computational load.

Table 3. Details of the ResNet architectures used for the teacher and student models

Feature	ResNet50 (Teacher Model)	ResNet18 (Student Model)
Total layers	50	18
Initial Conv Layer	7×7, 64, stride 2	3×3 Max Pool, stride 2
Initial Pooling Layer	7×7, 64, stride 2	3×3 Max Pool, stride 2
Residual Block 1/Channels	3 Bottleneck Units/246	2 Basic Units/64
Residual Block 2/Channels	4 Bottleneck Units/512	2 Basic Units/128
Residual Block 3/Channels	6 Bottleneck Units/1024	2 Basic Units/256
Residual Block 4/Channels	3 Bottleneck Units/2048	2 Basic Units/512
Pooling Layer	Global Avg Pool	Global Avg Pool
Fully Connected Layer	1000-d FC Layer	1000-d FC Layer

3.3. Knowledge Distillation Processes

The knowledge distillation process involves transferring the knowledge from the teacher to the student model. We employ a combination of soft targets and intermediate representations during training. The soft targets, representing the teacher model's softened predictions, are integrated with traditional cross-entropy loss using the following formula.

$$L_{KD} = (1 - \alpha) L_{CE}(y, P^{(s)}) + \alpha \tau^2 D_{KL}(P^{(t)}/\tau, P^{(s)}/\tau) \quad (4)$$

where $\alpha \in (0, 1)$ is the balance factor between the two loss terms; L_{CE} is the cross-entropy loss; y is the one-hot label; $P^{(t)}$ is the teacher output; $P^{(s)}$ is the student output; D_{KL} is the KL divergence [47]; and τ is a temperature [8].

Additionally, we incorporate feature-matching techniques to ensure the student model captures intermediate representations from the teacher [20].

3.3.1. Response-Based Knowledge Distillation (RKD)

Response-based knowledge distillation (RKD) is a variant of knowledge distillation that refers to the neural response of the last output layer of the teacher model [48]. The operating principle of the RKD is illustrated in Figure 3. According to Figure 3, response-based knowledge focuses on the final output layer of the teacher model. This is accomplished by the assumption that the student model will learn to mimic the predictions of the teacher model.

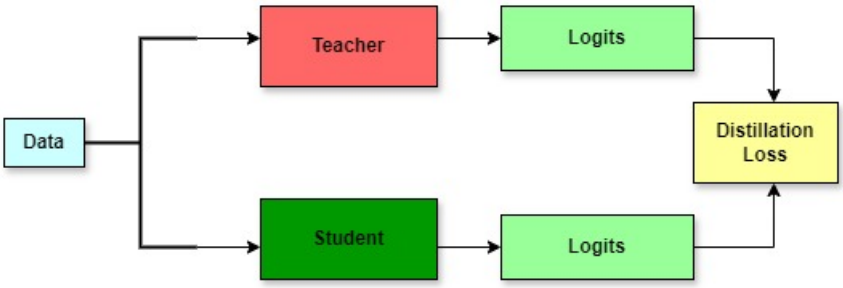


Figure 3. Response-based knowledge distillation [48].

The illustration in Figure 3 shows that this can be achieved by using a loss function, called the distillation loss, which captures the difference between the respective logits of the student model and the teacher model. As this loss would be minimized during the learning process, the student model would become increasingly capable of making the same predictions as the teacher model. By considering the decision-making process of the teacher model, response-based methods can potentially improve the generalization ability and robustness of the student model.

3.3.2. Intermediate Knowledge Distillation

Intermediate-based knowledge distillation (IKD), or feature-based knowledge distillation, is a variant of knowledge distillation in DNN that highlights knowledge learned from hidden layers. The operating principle of IKD is illustrated in Figure 4.

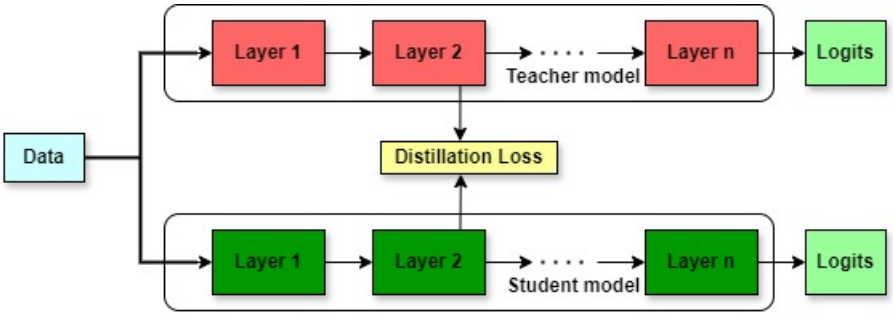


Figure 4. Intermediate knowledge distillation [48].

According to Figure 4, IKD extends traditional knowledge distillation by transferring knowledge not just from the final output layer of the teacher model but also from intermediate layers. Indeed, a trained teacher model also captures knowledge of the data in its intermediate layers, which is particularly relevant for deep neural networks. Thus, the intermediate layers learn to discriminate specific features, and this knowledge can be used to train a student model. As depicted in Figure 4, the aim is to train the student model to learn the same feature activations as the teacher model. The distillation loss function achieves this goal by minimizing the difference between the feature activations of the teacher model and the student model. IKD requires careful design to balance the complexity of transferring knowledge from multiple layers while ensuring computational efficiency and avoiding issues such as vanishing gradients.

4. Experimental Setup and Results Analysis

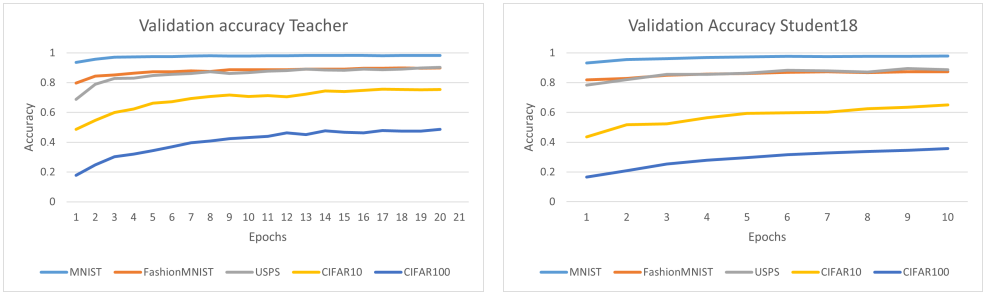
To investigate the impact of datasets, we conduct experiments with varying configurations, including knowledge distillation with and without dataset-specific adaptations. The success of these manipulations depends on the optimal configuration of experimental parameters and a logical, transparent experimental protocol, which we present in Section 4.1 below.

4.1. Experimental Setup

As shown in Figure 1 and further motivated by the literature review in Section 2, we use teacher-student architecture to distill the knowledge in DNN. So ResNet50 was used as the teacher model and ResNet18 as the student model.

The teacher model is first trained on the original dataset, producing accurate predictions. We also trained the students from scratch to later compare the results after training the students via distillation. Figure 5 shows the validation accuracy over epochs during the training of the teacher and the student from scratch.

During the knowledge distillation process, the student model is trained on the same dataset using a combination of ground truth labels and soft targets generated by the teacher. This dual learning approach helps the student model generalize better and capture intricate patterns. The loss function used in knowledge distillation incorporates both the traditional cross-entropy loss, comparing the student's predictions with the ground truth labels, and a distillation loss, quantifying the similarity between the student's predictions and the soft targets provided by the teacher. The distillation loss encourages the student to mimic the teacher's decision-making process.



(**a**) (**b**)

Figure 5. Variation in the validation accuracy by epochs for (**a**) the teacher model (ResNet50) and (**b**) the student model (ResNet18).

As ref. [15] confirms that good data augmentation can be used to obtain considerable knowledge distillation. For data augmentation, we use RandomRotation with the value of 15 to randomly rotate the image by up to 15 degrees, RandomHorizontalFlip to randomly flip the image horizontally, and RandomVerticalFlip to randomly flip the image vertically. We transform the images to a PyTorch (version 2.1.2) tensor, and finally, we normalize the data. The cross-entropy loss was used to train all models with the ground truth label, and the distillation loss used was Kullback–Leibler divergence. The hyperparameter controlling the balance between the two losses was $\alpha = 0.7$. The temperature was $t = 4$ [8]. We trained the teacher model within 20 epochs and the students within 10 epochs. We use SGD as an optimizer, and the value of the learning rate was $lr = 0.001$. The Kaggle environment (GPU P100) was used as the hardware and PyTorch as the software to conduct experiments. Each dataset was split into three different subdatasets for training, validation, and testing. The following Table 4 shows the different sizes of each sub-dataset.

Table 4. Distribution of different data sizes for training (83.33%), validation (11.66%), and testing (5%). These data sizes were chosen based on experimental results from the literature review.

Dataset	Training	Validation	Test
CIFAR-10	50,000	7000	3000
CIFAR-100	50,000	7000	3000
USPS	7291	1404	603
MNIST	60,000	7000	3000
Fashion MNIST	60,000	7000	3000

Evaluation metrics encompass traditional classification metrics such as accuracy as shown in Table 1. We conducted multiple runs for each experiment to account for variability and report averaged results for robust conclusions.

Our methodology combines a diverse set of datasets, state-of-the-art model architectures, and a nuanced knowledge distillation process. This comprehensive approach aims to elucidate the impact of datasets on the effectiveness of knowledge distillation in image classification, providing valuable insights for researchers and practitioners in the field.

4.2. Results Analysis

After simulations, the analysis of the results obtained consists in turn of analyzing and comparing the performances of the teacher (ResNet50) and student (Resnet18 from scratch) models on all the data sets (Section 4.2.1). Then, the analysis and comparison of knowledge distilled between the teacher (ResNet50) and the pupil (ResNet18) models in RKD and IKD in Sections 4.2.2 and 4.2.3, respectively.

4.2.1. Analysis of the Results of the Teacher and Student Models from Scratch

Let us remember once again that the first step in knowledge transfer is to train the teacher model since its results will guide the learning of the student model. Figure 6 shows the results of the teacher model after training on the different databases. It also shows the results of the student model from scratch, which will serve as a basis for comparison after knowledge distillation.

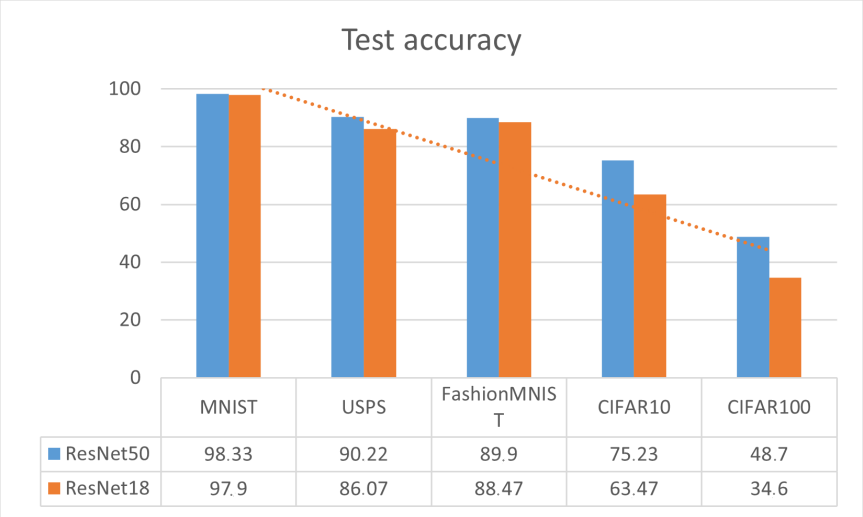


Figure 6. Test accuracy for the teacher and instance student model from scratch.

Looking at Figure 6, we can easily notice that the teacher model performs better than the student model. Indeed, as the student model is shallower than the teacher model, it will also be less accurate. Table 5 completes this figure by presenting the performance differences between the two models on the involved databases. From these representations, we can also see that the performance of both models decreases with database complexity. Further analysis after distillation will enable us to determine whether the same behavior will be observed.

Table 5. Difference between teacher and student accuracy.

Dataset	ResNet50	ResNet18	Difference
MNIST	98.33	97.9	−0.43
FashionMNIST	89.9	88.47	−1.43
USPS	90.22	86.07	−4.15
CIFAR10	75.23	63.47	−11.76
CIFAR100	48.7	34.6	−14.1

Once the teacher model has been trained, the training of the student model can be followed by knowledge distillation. We carried out two different types of distillation experiments, namely RKD [8] for response-based KD and IKD [20] for intermediate-based KD. Sections 4.2.2 and 4.2.3 present the results of these distillations, respectively.

4.2.2. RKD Performance Results Analysis

In the RKD architecture, the student model is trained and guided by the results of the last layer of the teacher model [8]. Figure 7 shows the results of the student model after training by RKD on the different databases.

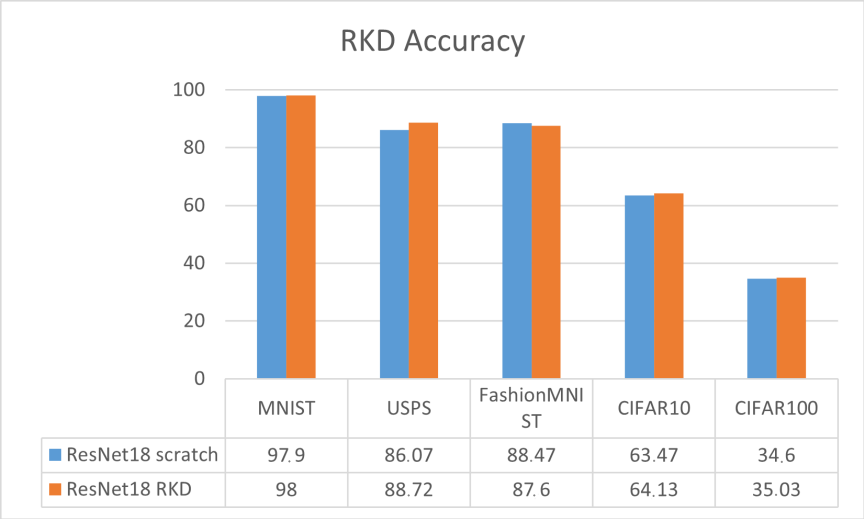


Figure 7. Test accuracy for the teacher and instance student models RKD.

In Figure 7, we can generally see a slight performance gain for the student model. This gain increases as the complexity of the database increases. To complement Figure 7, Table 6 shows the performance gap between the student instance trained from scratch and that trained by response-based knowledge distillation.

Table 6. Difference between the student model from scratch and the student RKD accuracy.

Dataset	ResNet18 Scratch	ResNet18 RKD	Difference
MNIST	97.9	98	+0.1
FashionMNIST	88.47	87.6	−0.87
USPS	86.07	88.72	+2.65
CIFAR10	63.47	64.13	+0.66
CIFAR100	34.6	35.03	+0.43

Part (b) of Figures A1, A3, A5, A7 and A9 shows the precision and loss curves respectively during the epochs of RKD training of the student model on the MNIST, USPS, FashionMNIST, CIFAR10 and CIFAR100 databases.

4.2.3. IKD Performance Results Analysis

In the IKD architecture, the student model is trained and guided by the results of the teacher model's intermediate layer [20]. Figure 8 shows the results of the student model after training by IKD on the different databases.

According to the illustration in Figure 8, we observe a considerable overall performance gain for the student model. This gain is even greater as the complexity of the database increases and is much better than that of RKD. Once again, Table 7 completes Figure 8 by presenting the numerical differences in performance between the student instance trained from scratch and that trained by intermediate-based knowledge distillation (IKD).

Part (c) of Figures A1, A3, A5, A7 and A9 shows the precision and loss curves during the epochs of IKD training of the student model on the MNIST, USPS, FashionMNIST, CIFAR10 and CIFAR100 databases, respectively.

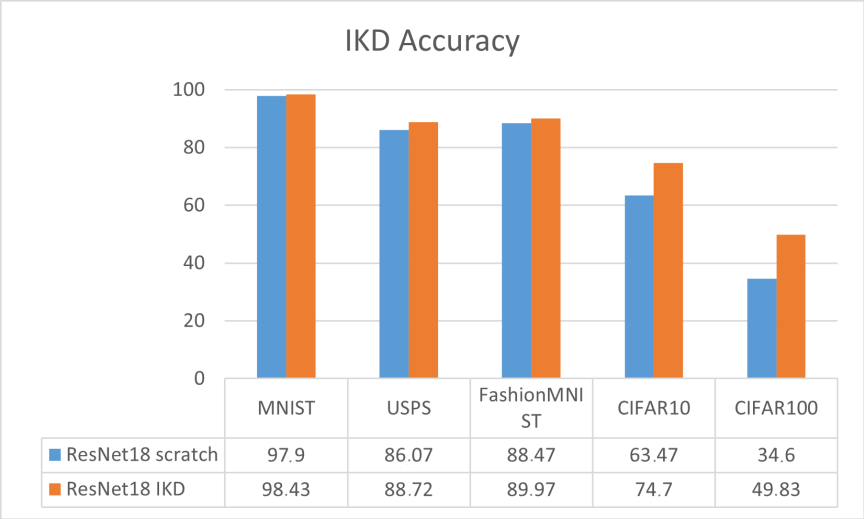


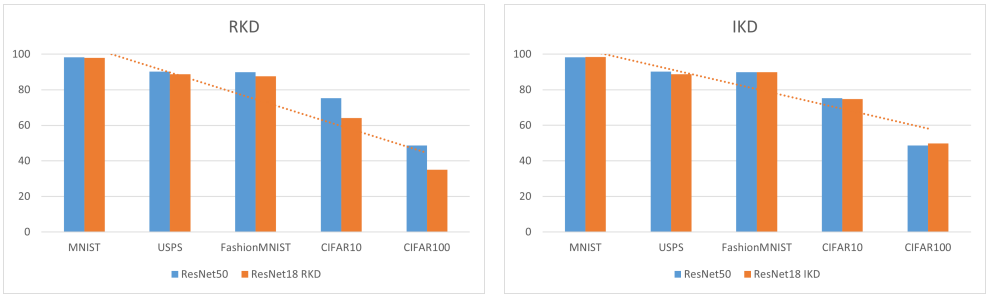
Figure 8. Difference between the student model from scratch and the student IKD accuracy.

Table 7. Table of difference between the student model from scratch and student IKD accuracy.

Dataset	ResNet18 Scratch	ResNet18 IKD	Difference (KD)
MNIST	97.9	98.43	+0.53
FashionMNIST	88.47	89.97	+1.5
USPS	86.07	88.72	+2.65
CIFAR10	63.47	74.7	+10.6
CIFAR100	34.6	49.83	+15.23

4.2.4. Analysis of the Impact of the Database on Knowledge Distillation

After analysing and comparing the results of the teacher model with those of the different instances of the student model, in this section we will analyse the effect of the databases on the distillation itself. To do this, we will first look at Figure 9 which shows the results of the different distillations compared with those of the teacher model; then we will look at Figures 10 and 11 which present the effect of distillation on the different databases and finally we will observe Figure 12 which presents the impact of datasets on knowledge distillation.



(**a**) (**b**)

Figure 9. Difference between the teacher and instance student model distilled from RKD (**a**) and instance student model distilled from IKD (**b**).

We can draw two major observations from Figure 9 by comparing it with Figure 6. The first observation concerns the RKD: although the student model gains in performance from the RKD, this gain is nevertheless slight, and the observation that the performance of the two models decreases as the complexity of the database increases is confirmed. On the other hand, when we look at the IKD, the gain for the student model is much more

significant. Here, we see that, unlike the others, the student model gains much more in performance as the complexity of the database increases.

Figure 10 shows the gains in student performance after distillation. We first note that IKD [20] performs significantly better than RKD [8]. We note that the more complex the database, the greater the gain in terms of performance. Part (c) IKD of Figures A2, A4, A6, A8 and A10 confirms this last observation. Indeed, we observe a significant increase in the f1-score compared to part (a), from scratch, and part (b), RKD. This increase is proportional to the complexity of the database. We can conclude from this that the more complex the database, the greater the effect of distillation.

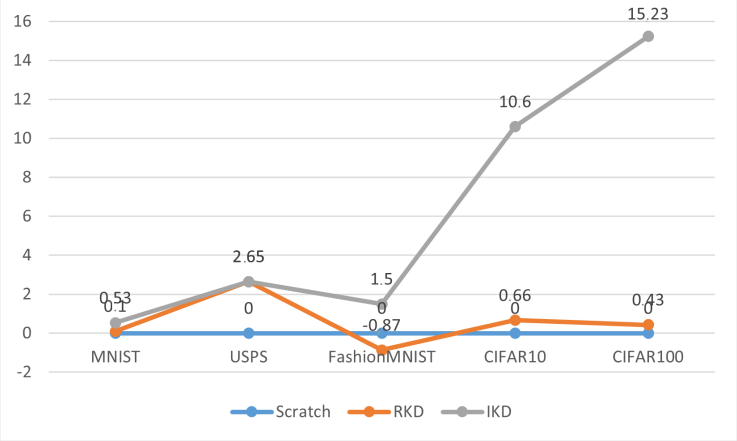
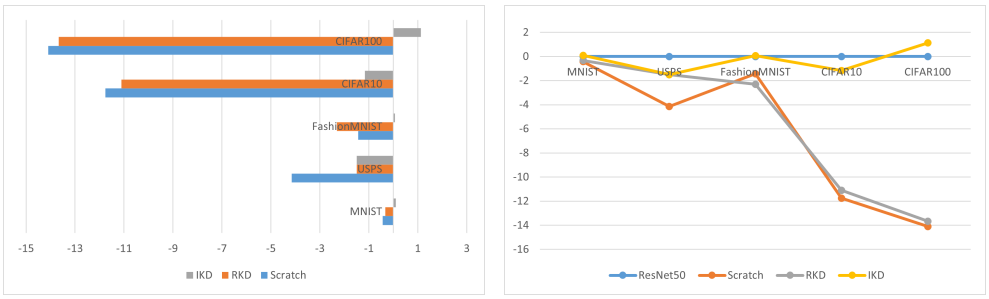


Figure 10. Student performance gain after distillation.

Figure 11 shows us the differences in performance between the different instances of the student model (from scratch, RKD, and IKD) compared to that of the teacher model. Knowledge distillation is indeed effective, and we even note that in the case of IKD, the student performs better than the teacher. On the other hand, we observe that in the least complex databases (MNIST, USPS and FashionMNIST), the performances between the teacher and the different instances of the student are approximately the same. We observe a notable difference in the IKD framework on the CIFAR10 and CIFAR100 databases. This leads us to draw two conclusions:

1. Knowledge distillation has a considerable effect on problems with complex databases. The more complex the database, the deeper and more powerful the model used for training. With a powerful teacher model capable of characterizing knowledge, the transfer to the student model will be assured.
2. By observing the performance provided by RKD and that provided by IKD on different databases, we conclude that the choice of the IKD method will be preferable to that of RKD when dealing with complex databases.



(**a**) (**b**)

Figure 11. All instances of student performance compared to teacher performance. bar visualisation (**a**) and curve visualisation (**b**).

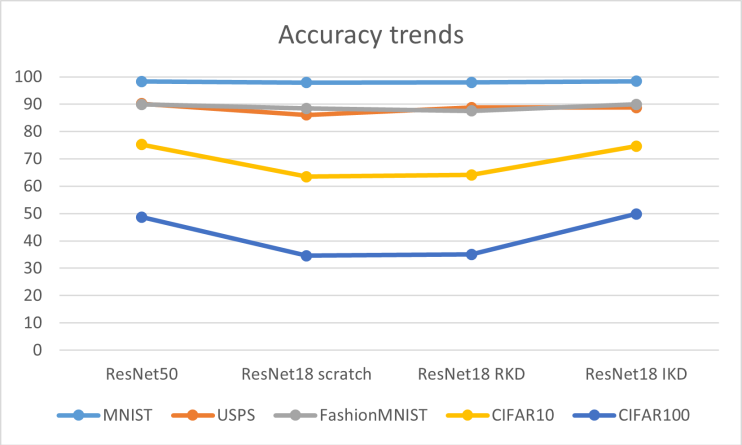


Figure 12. Impact of dataset.

According to Figure 12, we see a slight variation in the curve for the LOW and LOW TO MODERATE databases, namely MNIST and FashionMNIST. The MODERATE (USPS) database curve shows a slightly more marked variation. Finally, the most complex CIFAR10 and CIFAR100 databases (MODERATE TO HIGH and HIGH) show a significant variation.

5. Discussion

The analysis of the results sheds valuable light on the effect of databases on knowledge distillation. By highlighting the importance of choosing the appropriate distillation method according to the complexity of the data and the learning objectives, these results could have important implications for the development of more robust and generalizable learning models. We highlight these insights in Sections 5.1 and 5.2.

5.1. Impact of Database Complexity on Distillation

By examining the performance curves for different databases, we observed significant variations according to the complexity of the data. This observation highlights the importance of considering the diversity of the data and its specific characteristics when designing learning models. The results show a significant difference between RKD and IKD. While RKD shows modest performance gains, IKD shows much more significant improvements, especially with complex databases. This raises questions about the mechanisms underlying these two approaches and their effectiveness in different contexts. More specifically, IKD outperforms RKD, mainly because of the nature of the information that each method transfers from the teacher's model to the student's model. IKD focuses on aligning the student's internal representations or feature maps with those of the teacher at different levels [20,49]. This method ensures that the student model not only learns the final results but also mimics the teacher's hierarchical feature extraction process, capturing richer and more nuanced information throughout its architecture [20]. Indeed, the theoretical underpinnings support this advantage. Intermediate representations contain fine-grained information and hierarchical abstractions that are crucial for complex tasks. By transferring these representations, the student model is better equipped to understand and generalize from the data. This approach exploits the concept of learning intermediate features, which are often more informative than final logs alone, particularly in deep networks where each layer captures progressively higher-level abstractions. In contrast, RKD relies solely on the teacher's final logits [8,9]. Although this method helps the student to know the ultimate limits of the decision, it does not provide the intermediate knowledge essential for a comprehensive understanding of the input [48]. This can lead to less effective transfer, as the student does not benefit from the multi-level learning process followed by the teacher. Interestingly, IKD seems to be more resilient to increasing data complexity; the results show that in some cases, the distilled student (especially with IKD) can even outperform the

teacher in terms of performance. This suggests that the transmission of knowledge through abstract features may be more robust in varied or complex data environments.

That said, knowledge distillation may lead to better generalization or adaptation to specific test data.

5.2. Optimisation of Distillation Strategies

The results indicate the need to develop more sophisticated distillation strategies that take into account the specific nature of the data and the characteristics of the models. In fact, the more complex the database, the greater the effect of distillation on improving the performance of the student model. This observation highlights the importance of taking into account the specific nature of the data when choosing the distillation method and designing the model. According to the results obtained, the IKD method is preferable to RKD due to its greater performance gains.

5.3. Limitation of the Study

Although the results obtained in our work are very interesting, we are aware that our study may have certain limitations. The limited choice of model architecture (ResNet50, ResNet18) used, the fact that the scope was limited to image classification tasks, the nature of the data used, and the choice of distillation methods (RKD, IKD) were deliberate choices to maintain a controlled and detailed analysis in a well-defined context. The performance measures and evaluation methods used in this study could also be a limitation. The scope of the literature search was limited due to access restrictions on some articles, leading to our potentially overlooking important findings that could influence our results. The limitation in isolating the variable impact on knowledge distillation performance; indeed, we compared KD performance on very different datasets rather than systematically varying individual parameters while holding other factors constant. The interpretation of the results is also open to discussion.

6. Conclusions

We conducted a thorough examination of the impact of databases on knowledge distillation in the context of image classification. We have used a diverse array of databases with different levels of complexity. We were able to derive several important and meaningful conclusions by meticulously analyzing the performance of both teacher and student models across various distillation methods.

Firstly, our results clearly demonstrated that knowledge distillation can be pivotal in enhancing the performance of student models, particularly in scenarios where the data are intricate and heterogeneous. Specifically, the IKD method exhibited more substantial performance improvements compared to the RKD method, underscoring the significance of transferring knowledge through abstract and generalizable representations. Furthermore, we observed that the complexity of the database plays a critical role in determining the effectiveness of knowledge distillation. Our findings indicated that as the complexity of the database increases, so do the performance gains of the student model, emphasizing the necessity of considering the unique characteristics of the data during the distillation process.

Additionally, our comprehensive analyses allowed us to compare the performance of the teacher and student models in detail, revealing instances where the distilled student models actually outperformed their teacher counterparts. This observation highlights the remarkable potential of knowledge distillation to foster improved generalization and adaptation to specific test data. Moreover, our results provided guidance on selecting the most appropriate distillation method based on the complexity of the database. Specifically, they suggest that the IKD method is particularly advantageous in scenarios involving complex and varied data.

Overall, our study offers valuable insights into the influence of databases on knowledge distillation, contributing important perspectives for the development of more robust, generalizable, and efficient machine learning models applicable to a wide range of domains. By delving into the nuances of how different distillation methods perform across diverse datasets, we provide a deeper understanding that can inform future research and practical applications in the field of machine learning and image analysis in particular.

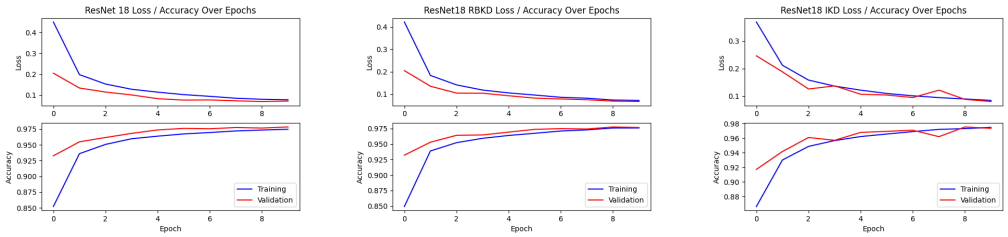
Author Contributions: Conceptualization, A.G.B. and C.S.T.K.; methodology, A.G.B. and C.S.T.K.; software, A.G.B. and C.S.T.K.; validation, A.G.B., C.S.T.K., M.E.H. (Mohamed El Haziti) and M.E.H. (Mohammed El Hassouni); data curation, A.G.B. and C.S.T.K.; writing—original draft preparation, A.G.B. and C.S.T.K.; writing—review and editing, A.G.B., C.S.T.K., M.E.H. (Mohamed El Haziti) and M.E.H. (Mohammed El Hassouni); project administration, M.E.H. (Mohamed El Haziti) and M.E.H. (Mohammed El Hassouni); funding acquisition, A.G.B. and C.S.T.K.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A



(**a**) (**b**) (**c**)

Figure A1. Accuracy and loss over epochs during the training phase of the student model in the MNIST dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

Test Accuracy: 97.90%
Classification Report:

	precision	recall	f1-score	support
0	0.98	0.99	0.99	307
1	1.00	0.99	1.00	356
2	0.93	0.99	0.96	300
3	0.99	0.99	0.99	312
4	0.98	1.00	0.99	306
5	0.98	0.92	0.95	256
6	0.98	0.97	0.98	301
7	0.97	0.97	0.97	296
8	1.00	0.99	0.99	293
9	0.99	0.97	0.98	273
accuracy			0.98	3000
macro avg	0.98	0.98	0.98	3000
weighted avg	0.98	0.98	0.98	3000

Confusion Matrix:

```
[[305   0   0   0   0   1   1   0   0]
 [  1 354   0   0   0   0   1   0   0]
 [  0   0 296   1   0   3   0   0   0]
 [  0   0   1 308   0   1   0   2   0]
 [  0   0   0   0 305   0   0   0   1]
 [  2   0  12   2   1 235   2   2   0]
 [  2   1   3   0   0   0 293   1   1]
 [  0   0   4   1   1   0   2 287   0]
 [  0   0   2   0   1   0   0   0 290]
 [  1   0   0   0   3   1   2   2   0 264]]
```

(**a**)

Test Accuracy: 98.00%
Classification Report:

	precision	recall	f1-score	support
0	0.99	0.99	0.99	307
1	0.98	0.99	0.99	356
2	0.95	0.97	0.96	300
3	0.99	0.99	0.99	312
4	0.99	0.98	0.99	306
5	0.98	0.93	0.95	256
6	1.00	0.97	0.98	301
7	0.95	0.99	0.97	296
8	1.00	1.00	1.00	293
9	0.98	0.98	0.98	273
accuracy			0.98	3000
macro avg	0.98	0.98	0.98	3000
weighted avg	0.98	0.98	0.98	3000

Confusion Matrix:

```
[[304   0   0   0   1   1   0   1   0   0]
 [  0 354   0   0   0   0   0   2   0   0]
 [  0   1 291   2   0   4   0   2   0   0]
 [  0   0   1 308   0   0   0   2   1   0]
 [  0   1   0   0 301   0   0   0   0   4]
 [  0   0  13   1   0 239   1   2   0   0]
 [  2   4   0   0   0   0 292   2   0   1]
 [  0   1   2   1   0   0   0 292   0   0]
 [  0   0   0   0   0   0   0   1 292   0]
 [  0   0   0   0   2   1   0   3   0 267]]
```

(**b**)

Test Accuracy: 98.43%
Classification Report:

	precision	recall	f1-score	support
0	0.99	0.99	0.99	307
1	0.99	1.00	0.99	356
2	0.94	0.98	0.96	300
3	1.00	1.00	1.00	312
4	0.99	1.00	1.00	306
5	0.99	0.93	0.96	256
6	0.98	0.98	0.98	301
7	0.98	0.97	0.98	296
8	0.99	1.00	1.00	293
9	0.99	0.98	0.98	273
accuracy			0.98	3000
macro avg	0.98	0.98	0.98	3000
weighted avg	0.98	0.98	0.98	3000

Confusion Matrix:

```
[[305   0   0   0   0   0   1   1   0   0]
 [  0 355   0   0   0   0   0   1   0   0]
 [  0   0 295   0   0   2   1   1   1   0]
 [  0   0   0 311   0   0   0   0   1   0]
 [  0   0   0   0 305   0   0   0   0   1]
 [  1   1  14   0   0 238   1   1   0   0]
 [  1   2   2   0   0   0 296   0   0   0]
 [  1   1   3   0   1   0   0 288   0   2]
 [  0   0   0   0   0   0   0   0 293   0]
 [  0   0   0   1   1   1   2   1   0 267]]
```

(**c**)

Figure A2. Student metrics after the training phase of the student model in MNIST dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

(**a**) (**b**) (**c**)

Figure A3. Accuracy and loss over epochs during the training phase of student model in USPS dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

Test Accuracy: 86.07%
Classification Report:

	precision	recall	f1-score	support
0	0.93	0.97	0.95	115
1	0.99	0.99	0.99	67
2	0.78	0.75	0.76	71
3	0.83	0.94	0.88	48
4	0.95	0.87	0.90	60
5	0.81	0.69	0.75	55
6	0.79	0.85	0.81	52
7	0.79	0.88	0.84	43
8	0.89	0.83	0.86	48
9	0.72	0.70	0.71	44
accuracy			0.86	603
macro avg	0.85	0.85	0.85	603
weighted avg	0.86	0.86	0.86	603

Confusion Matrix:

```
[[112   0   2   0   0   0   1   0   0]
 [  0  66   0   0   0   0   0   1   0   0]
 [  2   0  53   5   0   4   2   2   3   0]
 [  1   0   0  45   0   2   0   0   0   0]
 [  0   0   1   0  52   0   1   2   1   3]
 [  2   0   8   2   0  38   0   3   1   1]
 [  0   0   0   0   0   0   0  44   1   0   7]
 [  0   0   2   1   1   1   0  38   0   0]
 [  1   1   2   1   0   2   0   0  40   1]
 [  2   0   0   0   2   0   9   0   0  31]]
```

(**a**)

Test Accuracy: 88.72%
Classification Report:

	precision	recall	f1-score	support
0	0.93	0.98	0.96	115
1	0.99	0.99	0.99	67
2	0.84	0.76	0.80	71
3	0.96	0.92	0.94	48
4	0.96	0.85	0.90	60
5	0.81	0.85	0.83	55
6	0.80	0.87	0.83	52
7	0.90	0.86	0.88	43
8	0.89	0.88	0.88	48
9	0.72	0.82	0.77	44
accuracy			0.89	603
macro avg	0.88	0.88	0.88	603
weighted avg	0.89	0.89	0.89	603

Confusion Matrix:

```
[[113   0   1   0   0   0   1   0   0   0]
 [  0  66   0   0   0   0   0   1   0   0]
 [  4   1  54   0   1   6   1   0   4   0]
 [  0   0   1  44   0   3   0   0   0   0]
 [  0   0   1   0  51   0   2   0   1   5]
 [  2   0   1   2   0  47   0   2   0   1]
 [  0   0   1   0   0   0  45   1   0   5]
 [  1   0   3   0   1   0   1  37   0   0]
 [  0   0   1   0   0   2   0   0  42   3]
 [  1   0   1   0   0   0   6   0   0  36]]
```

(**b**)

Test Accuracy: 88.72%
Classification Report:

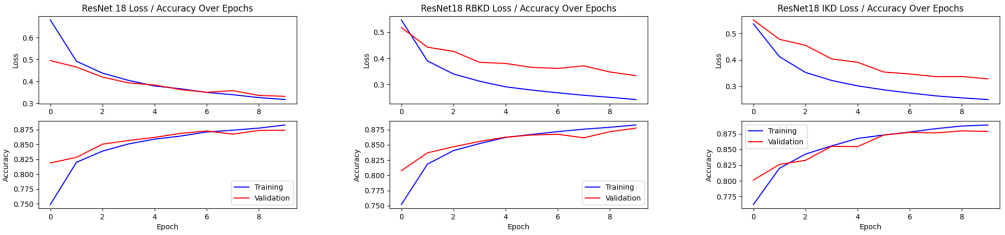
	precision	recall	f1-score	support
0	0.94	0.99	0.97	115
1	1.00	0.99	0.99	67
2	0.85	0.79	0.82	71
3	0.93	0.88	0.90	48
4	0.96	0.88	0.92	60
5	0.76	0.80	0.78	55
6	0.86	0.81	0.83	52
7	0.84	0.88	0.86	43
8	0.92	0.94	0.93	48
9	0.71	0.80	0.75	44
accuracy			0.89	603
macro avg	0.88	0.87	0.88	603
weighted avg	0.89	0.89	0.89	603

Confusion Matrix:

```
[[114   0   1   0   0   0   0   0   0   0]
 [  0  66   0   0   0   0   0   1   0   0]
 [  2   0  56   1   0   7   1   1   2   1]
 [  0   0   0  42   0   6   0   0   0   0]
 [  0   0   0   1   0  53   0   0   1   1   4]
 [  2   0   4   2   0  44   0   2   1   0]
 [  1   0   1   0   0   0  42   1   0   7]
 [  0   0   2   0   1   0   2  38   0   0]
 [  0   0   1   0   0   0   0   0  45   2]
 [  2   0   0   0   1   1   4   1   0  35]]
```

(**c**)

Figure A4. Student metrics after the training phase of the student model in the USPS dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.



(**a**) (**b**) (**c**)

Figure A5. Accuracy and loss over epochs during the training phase of the student model in FashionMNIST dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

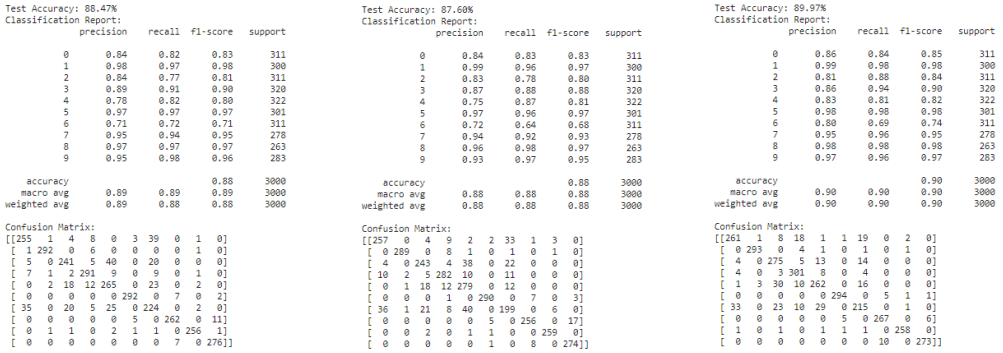


Figure A6. Student metrics after the training phase of the student model in the FashionMNIST dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

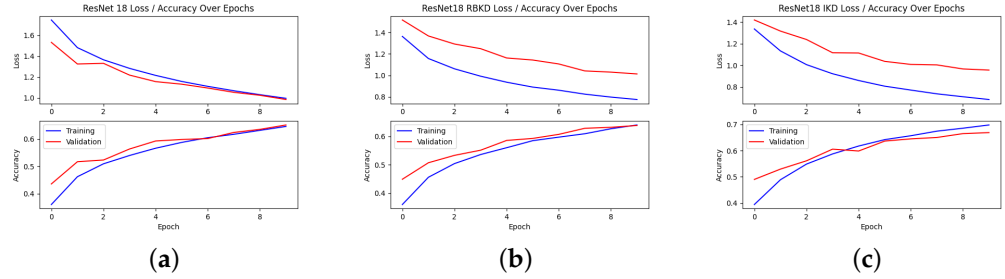


Figure A7. Accuracy and loss over epochs during the training phase of the student model in the CIFAR10 dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

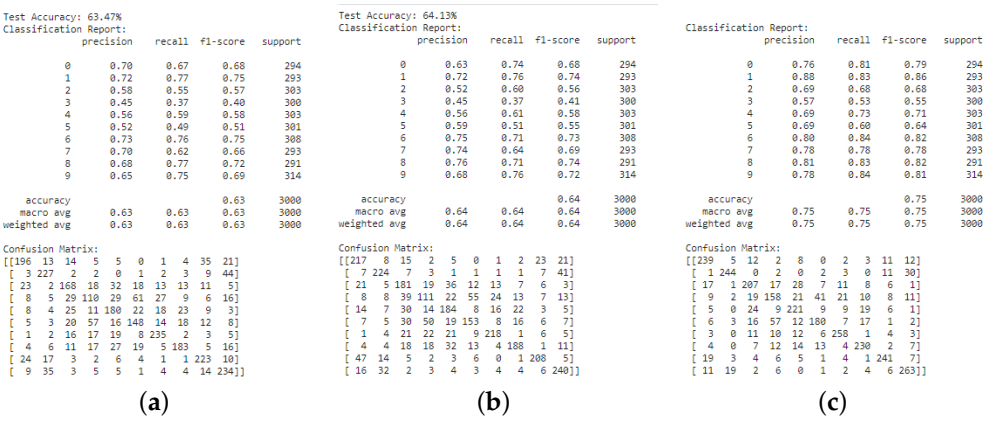


Figure A8. Student metrics after the training phase of the student model in the CIFAR10 dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

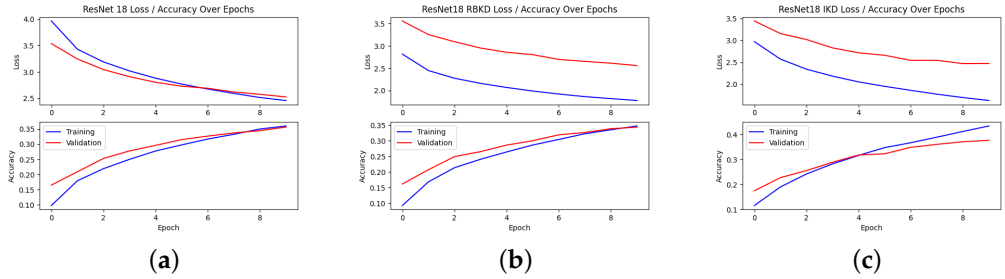
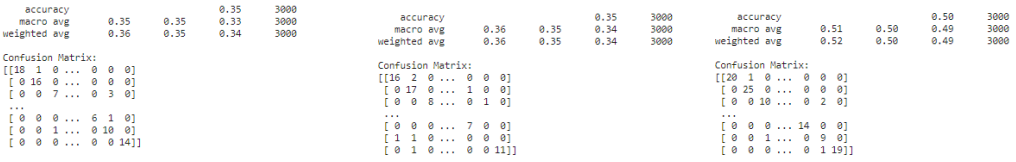


Figure A9. Accuracy and loss over epochs during the training phase of the student model in the CIFAR100 dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.



(**a**) (**b**) (**c**)

Figure A10. Student metrics after the training phase of the student model in the CIFAR100 dataset. (**a**) Training student from scratch, (**b**) RKD student training, and (**c**) IKD student training.

References

1. Chen, G.; Choi, W.; Yu, X.; Han, T.; Chandraker, M. Learning efficient object detection models with knowledge distillation. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; Volume 30.
2. Feng, Y.; Wang, H.; Hu, H.R.; Yu, L.; Wang, W.; Wang, S. Triplet distillation for deep face recognition. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Virtual, 25–28 October 2020; pp. 808–812.
3. Jiao, X.; Yin, Y.; Shang, L.; Jiang, X.; Chen, X.; Li, L.; Wang, F.; Liu, Q. Tinybert: Distilling bert for natural language understanding. *arXiv* **2019**, arXiv:1909.10351.
4. Wang, H.; Li, Y.; Wang, Y.; Hu, H.; Yang, M.H. Collaborative distillation for ultra-resolution universal style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1860–1869.
5. Krizhevsky, A.; Hinton, G. Learning Multiple Layers of Features from Tiny Images. Master's Thesis, Department of Computer Science, University of Toronto, Toronto, ON, Canada, 2009. Available online: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf (accessed on1 June 2024).
6. LeCun, Y. The MNIST Database of Handwritten Digits. 1998. Available online: http://yann.lecun.com/exdb/mnist/ (accessed on 15 July 2024).
7. Xiao, H.; Rasul, K.; Vollgraf, R. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. *arXiv* **2017**, arXiv:1708.07747. [CrossRef]
8. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531. [CrossRef]
9. Ba, J.; Caruana, R. Do deep nets really need to be deep? *arXiv* **2014**, arXiv:1312.6184.
10. Radosavovic, I.; Dollár, P.; Girshick, R.; Gkioxari, G.; He, K. Data distillation: Towards omni-supervised learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4119–4128. [CrossRef]
11. Hull, J. A database for handwritten text recognition research. *IEEE Trans. Pattern Anal. Mach. Intell.* **1994**, *16*, 550–554. [CrossRef]
12. Che, Z.; Purushotham, S.; Khemani, R.; Liu, Y. Distilling knowledge from deep networks with applications to healthcare domain. *arXiv* **2015**, arXiv:1512.03542. [CrossRef]
13. Tian, Y.; Krishnan, D.; Isola, P. Contrastive representation distillation. *arXiv* **2019**, arXiv:1910.10699. [CrossRef]
14. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31. [CrossRef]
15. Wang, H.; Lohit, S.; Jones, M.N.; Fu, Y. What makes a "good" data augmentation in knowledge distillation-a statistical perspective. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 13456–13469.
16. Das, D.; Massa, H.; Kulkarni, A.; Rekatsinas, T. An empirical analysis of the impact of data augmentation on knowledge distillation. *arXiv* **2020**, arXiv:2006.03810.
17. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
18. Tung, F.; Mori, G. Similarity-preserving knowledge distillation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1365–1374. [CrossRef]
19. Alabbasy, F.M.; Abohamama, A.; Alrahmawy, M.F. Compressing medical deep neural network models for edge devices using knowledge distillation. *J. King Saud-Univ.-Comput. Inf. Sci.* **2023**, *35*, 101616. [CrossRef]
20. Romero, A.; Ballas, N.; Kahou, S.E.; Chassang, A.; Gatta, C.; Bengio, Y. Fitnets: Hints for thin deep nets. *arXiv* **2014**, arXiv:1412.6550. [CrossRef]
21. Zhang, X.; Chang, H.; Hao, Y.; Chang, D. MKTN: Adversarial-Based Multifarious Knowledge Transfer Network from Complementary Teachers. *Int. J. Comput. Intell. Syst.* **2024**, *17*, 72. [CrossRef]
22. Zhou, T.; Chiam, K.H. Synthetic data generation method for data-free knowledge distillation in regression neural networks. *Expert Syst. Appl.* **2023**, *227*, 120327. [CrossRef]
23. Zhang, J.; Tao, Z.; Zhang, S.; Qiao, Z.; Guo, K. Soft Hybrid Knowledge Distillation against deep neural networks. *Neurocomputing* **2024**, *570*, 127142. [CrossRef]
24. Wang, C.; Wang, Z.; Chen, D.; Zhou, S.; Feng, Y.; Chen, C. Online adversarial knowledge distillation for graph neural networks. *Expert Syst. Appl.* **2024**, *237*, 121671. [CrossRef]

25. Guermazi, E.; Mdhaffar, A.; Jmaiel, M.; Freisleben, B. MulKD: Multi-layer Knowledge Distillation via collaborative learning. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108170. [CrossRef]
26. Ojha, U.; Li, Y.; Sundara Rajan, A.; Liang, Y.; Lee, Y.J. What knowledge gets distilled in knowledge distillation? *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 11037–11048. [CrossRef]
27. Zagoruyko, S.; Komodakis, N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv* **2016**, arXiv:1612.03928. [CrossRef]
28. Li, H.T.; Lin, S.C.; Chen, C.Y.; Chiang, C.K. Layer-level knowledge distillation for deep neural network learning. *Appl. Sci.* **2019**, *9*, 1966. [CrossRef]
29. Chen, L.; Chen, Y.; Xi, J.; Le, X. Knowledge from the original network: Restore a better pruned network with knowledge distillation. *Complex Intell. Syst.* **2022**, *8*, 709–718. [CrossRef]
30. Yim, J.; Joo, D.; Bae, J.; Kim, J. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4133–4141. [CrossRef]
31. Srinivas, S.; Fleuret, F. Knowledge transfer with jacobian matching. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 4723–4731. Available online: https://proceedings.mlr.press/v80/srinivas18a.html (accessed on 25 June 2024).
32. Mirzadeh, S.I.; Farajtabar, M.; Li, A.; Levine, N.; Matsukawa, A.; Ghasemzadeh, H. Improved knowledge distillation via teacher assistant. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 5191–5198. [CrossRef]
33. Bang, D.; Lee, J.; Shim, H. Distilling from professors: Enhancing the knowledge distillation of teachers. *Inf. Sci.* **2021**, *576*, 743–755. [CrossRef]
34. Li, Z.; Li, X.; Yang, L.; Song, R.; Yang, J.; Pan, Z. Dual teachers for self-knowledge distillation. *Pattern Recognit.* **2024**, *151*, 110422. [CrossRef]
35. Shang, R.; Li, W.; Zhu, S.; Jiao, L.; Li, Y. Multi-teacher knowledge distillation based on joint Guidance of Probe and Adaptive Corrector. *Neural Netw.* **2023**, *164*, 345–356. [CrossRef] [PubMed]
36. Cho, Y.; Ham, G.; Lee, J.H.; Kim, D. Ambiguity-aware robust teacher (ART): Enhanced self-knowledge distillation framework with pruned teacher network. *Pattern Recognit.* **2023**, *140*, 109541. [CrossRef]
37. Torralba, A.; Efros, A.A. Unbiased look at dataset bias. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 1521–1528.
38. Oliva, A.; Torralba, A. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vis.* **2001**, *42*, 145–175. [CrossRef]
39. Basu, M.; Ho, T.K. *Data Complexity in Pattern Recognition*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006. [CrossRef]
40. Shah, B.; Bhavsar, H. Time complexity in deep learning models. *Procedia Comput. Sci.* **2022**, *215*, 202–210. [CrossRef]
41. Deng, L. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Process. Mag.* **2012**, *29*, 141–142. [CrossRef]
42. Albawi, S.; Mohammed, T.A.; Al-Zawi, S. Understanding of a convolutional neural network. In Proceedings of the 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 1–6.
43. O'shea, K.; Nash, R. An introduction to convolutional neural networks. *arXiv* **2015**, arXiv:1511.08458.
44. Sharma, N.; Jain, V.; Mishra, A. An analysis of convolutional neural networks for image classification. *Procedia Comput. Sci.* **2018**, *132*, 377–384. [CrossRef]
45. Rawat, W.; Wang, Z. Deep convolutional neural networks for image classification: A comprehensive review. *Neural Comput.* **2017**, *29*, 2352–2449. [CrossRef] [PubMed]
46. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 21–30 June 2016; pp. 770–778. [CrossRef]
47. Kullback, S. *Information Theory and Statistics*; Courier Corporation: North Chelmsford, MA, USA, 1997.
48. Gou, J.; Yu, B.; Maybank, S.J.; Tao, D. Knowledge distillation: A survey. *Int. J. Comput. Vis.* **2021**, *129*, 1789–1819. [CrossRef]
49. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828. [CrossRef]