

# Ensemble reverse knowledge distillation: training robust model using weak models

Christopher Gavra Reswara, Tjeng Wawan Cenggoro

School of Computer Science, Bina Nusantara University, West Jakarta, Indonesia

## Article Info

### Article history:

Received Sep 19, 2024

Revised Jun 28, 2025

Accepted Jul 13, 2025

### Keywords:

EfficientNet

Ensemble learning

Knowledge distillation

Transfer learning

Weak-to-strong

## ABSTRACT

To ensure that artificial intelligence (AI) can be aligned with humans, AI models need to be developed and supervised by humans. Unfortunately, it is possible for an AI to exceed human capabilities, which is commonly referred to as superalignment models. Thus, it raised the question of whether humans can still supervise a superalignment model, which is encapsulated in a concept called weak-to-strong generalization. To address this issue, we introduce ensemble reverse knowledge distillation (ERKD), which leverages two weaker models to supervise a more robust model. This technique is a potential solution for humans to manage a super-alignment of models. ERKD enables a more robust model to achieve optimal performance with the assistance of two weaker models. We tried to train a more robust EfficientNet model with weaker convolutional neural network (CNN) models in a supervised fashion. With this method, the EfficientNet model performed better than the model trained with the standard transfer learning (STL) method. It also performed better than a model that was supervised by a single weaker model. Finally, ERKD-trained EfficientNet models can perform better than EfficientNet models that are one or even two levels stronger.

*This is an open access article under the [CC BY-SA](#) license.*



## Corresponding Author:

Christopher Gavra Reswara

School of Computer Science, Bina Nusantara University

Kebon Jeruk Raya No. 27, West Jakarta, Indonesia

Email: christopher.reswara@binus.ac.id

## 1. INTRODUCTION

The development of artificial intelligence (AI) model must be integrated with human supervision to obtain a useful model for humans. For example, in the field of image classification, convolutional neural networks (CNN) models, such as ResNet [1], DenseNet [2], EfficientNet [3], Inception V3 [4], and MobileNet V3 [5] models, were asked to learn a collection of images labeled by experts, such as ImageNet [6], CIFAR-10 [7], Food-101 [8], Oxford 102 Flowers [9], Birdsnap [10], and other datasets. Large language models (LLMs) such as GPT-4 [11], Gemini 1.5 [12], and Llama-3 [13] were also built to learn human-generated text datasets to perform natural language processing (NLP) tasks. To add an additional guarantee of its alignment with humans, LLMs were also trained with an additional step called reinforcement learning from human feedback (RLHF), which rewards or punishes during learning based on human judgment [14]–[16]. Until now, all forms of AI have always been intentionally directed to align with human knowledge, experience, evaluation, and feedback to assist in completing human tasks.

However, the emergence of AI models that have better capabilities than humans, commonly referred to as superalignment models, is unavoidable. This is largely due to the fact that AI supervision was not usually done by a large crowd of humans. Most of the datasets that were used to train AI models nowadays were curated

via crowd-sourcing. This theoretically can crystallize the wisdom of the crowd within AI models, which can lead the models to be more intelligent than a single human. The emergence of superalignment models can also come from the practice of applying reinforcement learning without human supervision, which has been demonstrated multiple times in video games [17], board games [18], [19], and recently LLM [20].

The emergence of superalignment models raised the question: How can we as humans supervise these models to better align with us if they are better than us? As superalignment models can emerge from the wisdom of the crowd, perhaps we can also supervise these models via another wisdom of the crowd. This study aims to simulate this idea by having an ensemble of weaker models to supervise a stronger model. In the machine learning community, it is known that an ensemble of weaker models can form a strong model. This concept is named ensemble learning and has been used to form a strong machine learning model such as random forest [21] and XGBoost [22].

To achieve our aim, we designed a schema of more than one weaker teacher models to supervise one stronger model in the knowledge distillation (KD) framework [23]. We named this schema ensemble reverse knowledge distillation (ERKD). Figure 1 illustrates the ERKD schema with two weak teacher models. To simulate the idea of supervising a model that is already intelligent, we use transfer learning as the main task. In particular, we use transfer learning for image classification as the task. To measure the success of this study, we compare ERKD with a standard transfer learning (STL) procedure.

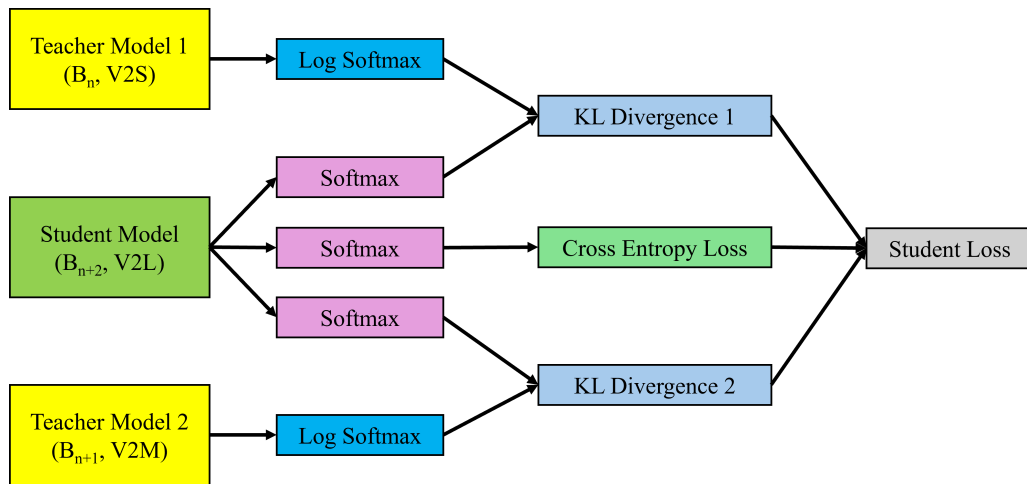


Figure 1. The ERKD schema with two weak teacher models

## 2. METHOD

### 2.1. Dataset

This study uses two image classification datasets, namely CIFAR-10 and CIFAR-100 [7]. Both datasets consist of 50,000 images for training and 10,000 images for testing. Both also have  $32 \times 32$  pixels resolution images. The difference between the two datasets is that CIFAR-10 only has ten classes, so each class consists of 6,000 images, while CIFAR-100 has 100 classes, so each class consists of 600 images. These two datasets are used in this study because they are commonly used in AI studies.

### 2.2. Train, validation, and test split data

The CIFAR-10 and CIFAR-100 datasets have been divided into 50,000 images for training and 10,000 for testing. All images in the training section have been randomized. Then, we split the training part into two parts, namely, 40,000 images used for training and 10,000 images used for validation. The 40,000 images used as the training model will be subjected to data augmentation. Meanwhile, the 10,000 validation images will calculate the error rate and validation when the model learns. Finally, 10,000 test images will be used to measure the performance of the model.

### 2.3. Data preprocessing

We preprocessed the dataset with z-score standardization on scale 0 to 1. Firstly, we normalize the pixel values from scale 0-to-255 to scale 0-to-1. Afterwards, we standardize the pixel values with z-score standardization, with the mean and standard deviation values derived from the dataset. For the CIFAR-10 dataset, the mean values were 0.4914, 0.4822, and 0.4465 for the red, green, and blue channels, respectively. The standard deviation values were 0.247, 0.243, and 0.261 for the red, green, and blue channels, respectively. For the CIFAR-100 dataset, the mean values were 0.5071, 0.4865, and 0.4409 for the red, green, and blue channels, respectively. The standard deviation values were 0.267, 0.256, and 0.276 for the red, green, and blue channels, respectively.

### 2.4. Data augmentation

To avoid overfitting, we applied data augmentation with a random crop to  $28 \times 28$  pixels and a random horizontal flip. This data augmentation procedure is applied only to the training dataset during model training. The data augmentation process was performed online for each epoch.

### 2.5. Models

For the transfer learning process in this study, we used EfficientNet and EfficientNet V2 [24] models, which were pre-trained on the ImageNet dataset [25], [26]. EfficientNet models have a hierarchy of weak models to strong models due to the use of systematic model scaling, i.e. from the weakest B0 to strongest B7 in EfficientNet and from the weakest V2S to the stronger V2M to the strongest V2L in EfficientNet v2. With this characteristic, EfficientNet models are perfect for the setup in this study.

### 2.6. Training process

The training process in all experiments in this study used Adam optimization [27] with a learning rate of  $10^{-3}$  and a ridge regularization of  $10^{-5}$ . In addition, training was conducted with 100 epochs, a batch size of 32, and the random seed used was 42. Furthermore, the temperature used in the KD process was 2.0. The checkpoint model technique is used during training based on the best validation accuracy. The image resolution scale in the EfficientNet study is also adjusted for each model in this study. EfficientNet models B0 to B7 use image sizes 32, 34, 38, 44, 54, 66, 76, and 86, respectively. Meanwhile, the EfficientNet V2 models, V2S, V2M and V2L, use image sizes of 32, 40, and 48, respectively.

### 2.7. Experiment setup

In ERKD, we used two weaker models to supervise a stronger model. For example, a stronger model EfficientNet B2 was supervised by using EfficientNet B1 and B0. The weaker models were first trained with STL on the CIFAR-10 and CIFAR-100 datasets. Afterwards, these two models were used as teachers by producing soft labels to train a stronger student model in a response-based KD framework. The stronger student model was optimized to match the distribution of the soft labels using the Kullback-Leibler divergence (KL divergence) loss function.

## 3. RESULTS AND DISCUSSION

In Table 1, we compare the accuracy of STL and three different variations of ERKD with a different proportion given to the loss functions: i) equal proportion, ii) 10% for cross entropy loss and 45% for KL divergence, and iii) 30% for cross entropy loss and 35% for KL divergence. The icons in the table indicate that ERKD outperforms the STL. The square indicates the best accuracy, the circle indicates the second-best accuracy, and the triangle indicates the third-best accuracy. As seen in the table, all variations of ERKD outperform STL. This proves that two weaker models can still supervise the stronger model, e.g. EfficientNet B0 and EfficientNet B1 can still supervise EfficientNet B2.

In addition, we also experimented using only one weaker model as a teacher of the stronger model. For example, EfficientNet model B2 is taught only by B0 or B1. The proportion of between the cross entropy loss and the KL divergence loss are both 50%. The results can be seen in Table 2. The icons in the table indicates that ERKD outperforms the STL and a single-teacher method. The square indicates the best accuracy, the circle indicates the second-best accuracy, and the triangle indicates the third-best accuracy. We found that at least one variation of ERKD can outperform using only one weaker model. This proved that the ensemble learning concept in ERKD is also effective in improving model performance. For example, EfficientNet B2 is more optimal when supervised by B0 and B1 than B0 or B1 alone.

Table 1. Comparison of the student model's accuracy between ERKD and STL

Teacher 1		Teacher 2		Student		Dataset	STL	Student accuracy			
Model	Image size	Model	Image size	Model	Image size			Average (%)	Teacher 1 and 2		
									10 45 45	30 35 35	
								(%)	(%)	(%)	
B1	34	B0	32	B2	38	CIFAR-10	88.84	89.79 ■	89.40 ▲	89.49 ●	
B2	38	B1	34	B3	44		90.63	91.34 ■	91.21 ●	91.18 ▲	
B3	44	B2	38	B4	54		91.69	92.87 ■	92.56 ▲	92.68 ●	
B4	54	B3	44	B5	66		92.63	92.94 ▲	93.23 ●	93.43 ■	
B5	66	B4	54	B6	76		93.02	93.23 ▲	93.61 ■	93.51 ●	
B6	76	B5	66	B7	86		93.18	93.78 ●	93.75 ▲	93.94 ■	
V2M	40	V2S	32	V2L	48	CIFAR-100	92.09	92.47 ▲	92.63 ●	92.65 ■	
B1	34	B0	32	B2	38		64.93	68.77 ■	68.39 ▲	68.68 ●	
B2	38	B1	34	B3	44		68.39	70.09 ▲	70.42 ■	70.36 ●	
B3	44	B2	38	B4	54		71.27	72.72 ▲	74.01 ■	73.84 ●	
B4	54	B3	44	B5	66		71.35	73.87 ●	74.27 ■	73.83 ▲	
B5	66	B4	54	B6	76		72.63	75.61 ●	75.12 ▲	75.80 ■	
B6	76	B5	66	B7	86		73.11	74.75 ▲	75.89 ●	75.92 ■	
V2M	40	V2S	32	V2L	48		69.82	70.98 ▲	70.99 ●	72.13 ■	

Table 2. Comparison of the student model's accuracy between ERKD using teachers 1 and 2, and STL

Teacher 1		Teacher 2		Student		Dataset	STL (%)	Teacher 1 Only (%)	Student accuracy			
Model	Image size	Model	Image size	Model	Image size				Teacher 2 Only (%)	Average (%)	Teacher 1 and 2	
										10 45 45	30 35 35	
										(%)	(%)	(%)
B1	34	B0	32	B2	38	CIFAR-10	88.84	89.54	89.24	89.79 ■	89.40	89.49
B2	38	B1	34	B3	44		90.63	90.87	91.24	91.34 ■	91.21	91.18
B3	44	B2	38	B4	54		91.69	92.19	92.33	92.87 ■	92.56 ▲	92.68 ●
B4	54	B3	44	B5	66		92.63	93.02	93.11	92.94	93.23 ●	93.43 ■
B5	66	B4	54	B6	76		93.02	93.17	93.33	93.23	93.61 ■	93.51 ●
B6	76	B5	66	B7	86		93.18	93.89	93.86	93.78	93.75	93.94 ■
V2M	40	V2S	32	V2L	48	CIFAR-100	92.09	92.52	91.98	92.47	92.63 ●	92.65 ■
B1	34	B0	32	B2	38		64.93	67.41	67.28	68.77 ■	68.39 ▲	68.68 ●
B2	38	B1	34	B3	44		68.39	69.29	69.34	70.09 ▲	70.42 ■	70.36 ●
B3	44	B2	38	B4	54		71.27	73.05	73.60	72.72	74.01 ■	73.84 ●
B4	54	B3	44	B5	66		71.35	74.12	72.86	73.87	74.27 ■	73.83
B5	66	B4	54	B6	76		72.63	74.26	75.06	75.61 ●	75.12 ▲	75.80 ■
B6	76	B5	66	B7	86		73.11	74.51	74.70	74.75 ▲	75.89 ●	75.92 ■
V2M	40	V2S	32	V2L	48		69.82	70.08	68.73	70.98 ▲	70.99 ●	72.13 ■

To check whether architectural similarity can influence the performance of ERKD, we picked other CNN models to replace EfficientNet models as teachers. The other CNN models were picked and mapped to replace EfficientNet models on the basis of similar accuracy on ImageNet dataset. Other CNN model architectures we finally picked were ResNet, RegNet [28], ConvNext [29], and ResNeXt. Table 3 provides the mapping of the other CNN model to their EfficientNet equivalent.

With the addition of other CNN models, we now have four candidates to be used as teachers: two weaker EfficientNet models and two other CNN models equivalent to the EfficientNet models. For the sake of simplicity, we named the first two EfficientNet models as teacher 1 and teacher 2, while the other two CNN models as teacher 3 and teacher 4. For example, to supervise EfficientNet B2, teacher 1 and teacher 2 are respectively B1 and B0, meanwhile teacher 3 and teacher 4 are respectively ResNet-101 and ResNet-152.

In Tables 4 and 5, we show the result of experiments on substituting only one EfficientNet teacher with other CNN models. The result with the icon in Table 4 indicates that ERKD outperforms the STL and a single teacher method. The square indicates the best accuracy, the circle indicates the second-best accuracy, and the triangle indicates the third-best accuracy. While, the result with the icon in Tables 5 and 6 indicates that ERKD outperforms the STL and a single-teacher method. The square indicates the best accuracy, the

circle indicates the second-best accuracy, the equilateral triangle indicates the third-best accuracy, and the right triangle indicates the fourth-best accuracy.

Table 3. The mapping of other CNN models to the EfficientNet models based on similar accuracy on ImageNet dataset

EfficientNet model	Others CNN model	EfficientNet accuracy (%)	Others CNN model (%)
B0	ResNet-101	77.692	77.374
B1	ResNet-152	77.692	77.374
B2	RegNet Y 16GF	77.692	77.374
B3	ConvNeXt Tiny	77.692	77.374
B4	ResNeXt101 64X4D	77.692	77.374
B5	ResNeXt101 64X4D	77.692	77.374
B6	ConvNeXt Small	77.692	77.374
V2S	ConvNeXt Base	77.692	77.374
V2M	ConvNeXt Large	77.692	77.374

Table 4. Comparison of the student model's accuracy between ERKD using teachers 1 and 4, and STL

Teacher 1	Teacher 4	Student	Dataset	STL (%)	Student accuracy				
Model	Model	Model			Teacher 1 Only (%)	Teacher 4 Only (%)	Average (%)	Teacher 1 and 4 10 45 45 (%)	30 35 35(%)
B1	B0	B2	CIFAR-10	88.84	89.54	89.46	89.54	89.57 ●	89.67 ■
B2	B1	B3		90.63	90.87	90.56	91.07 ●	90.94 ▲	91.34 ■
B3	B2	B4		91.69	92.19	92.20	92.54 ▲	92.83 ■	92.62 ●
B4	B3	B5		92.63	93.02	92.81	92.98	93.03 ●	93.34 ■
B5	B4	B6		93.02	93.17	93.62	94.05 ■	93.42	94.03 ●
B6	B5	B7		93.18	93.89	93.89	93.85	93.74	94.28 ■
V2M	V2S	V2L	CIFAR-100	92.09	92.52	92.30	92.63 ■	92.35	92.33
B1	B0	B2		64.93	67.41	65.95	67.59 ■	67.22	66.99
B2	B1	B3		68.39	69.29	68.57	69.53 ●	69.17	69.60 ■
B3	B2	B4		71.27	73.05	73.22	74.49 ■	73.10	73.51 ●
B4	B3	B5		71.35	74.12	72.76	74.32 ●	73.90	74.50 ■
B5	B4	B6		72.63	74.26	74.21	74.93 ●	74.88 ▲	75.14 ■
B6	B5	B7		73.11	74.51	74.66	75.62 ■	75.23 ▲	75.26 ●
V2M	V2S	V2L		69.82	70.08	69.95	72.30 ■	71.16 ▲	71.32 ●

Table 5. Comparison of the student model's accuracy between ERKD using teachers 2 and 3, and STL method

Teacher 2	Teacher 3	Student	Dataset	STL (%)	Teacher 2 Only (%)	Teacher 3 Only (%)	Student accuracy						
Model	Model	Model					Average (%)	10 45 45 (%)	20 40 40 (%)	30 35 35 (%)			
B1	B0	B2	CIFAR-10	88.84	89.24	89.58	89.79 ■	89.33	89.75 ●	89.64 ▲			
B2	B1	B3		90.63	91.24	91.14	91.56 ■	91.39 ▲	91.45 ●	91.40 ▲			
B3	B2	B4		91.69	92.33	92.46	92.10	92.42	92.63 ●	92.91 ■			
B4	B3	B5		92.63	93.11	93.26	93.46 ■	93.20	93.30 ●	93.20			
B5	B4	B6		93.02	93.33	93.95	93.41	93.79	94.15 ■	93.73			
B6	B5	B7		93.18	93.86	94.24	93.66	93.71	94.04	93.83			
V2M	V2S	V2L	CIFAR-100	92.09	91.98	92.47	92.31	92.63 ■	92.29	92.26			
B1	B0	B2		64.93	67.28	66.39	67.81 ■	67.03	67.65 ●	67.49 ▲			
B2	B1	B3		68.39	69.34	69.30	70.19 ■	70.06 ●	69.99 ▲	69.85 ▲			
B3	B2	B4		71.27	73.60	72.96	73.26	73.93 ■	73.24	73.41			
B4	B3	B5		71.35	72.86	72.99	73.47 ▲	73.81 ●	73.66 ▲	73.98 ■			
B5	B4	B6		72.63	75.06	74.67	74.93	75.61 ▲	76.19 ■	75.65 ●			
B6	B5	B7		73.11	74.70	74.55	75.60 ■	74.94 ▲	74.86 ▲	75.04 ●			
V2M	V2S	V2L		69.82	68.73	70.80	70.92 ▲	70.91 ▲	71.18 ●	71.39 ■			

In Table 4, only teacher 1 and teacher 4 are used. Meanwhile, only teacher 2 and teacher 3 are used in Table 5. We add a new proportion of 20% for cross entropy loss and 40% for KL divergence loss in Table 5. We also show the result of substituting all the EfficientNet teachers with other CNN models in Table 6. From these results, we found ERKD can generally still improve the accuracy compared to STL and using one teacher only. This fact is especially obvious when we see the accuracy of using different teachers combination side by side in Table 7, where there is no combination that dominantly outperforms other combination. The squares in Table 7 indicate superior performance. Thus, ERKD still works regardless of the architectural similarity.

Table 6. Comparison of the student model's accuracy between ERKD using teachers 3 and 4, and STL

Teacher 3 Model	Teacher 4 Model	Student Model	Dataset	STL (%)	Teacher 3 Only (%)	Teacher 4 Only (%)	Student accuracy							
							Average (%)	10 (%)	45 (%)	Teacher 3 and 4 (%)	30 (%)	35 (%)	50 (%)	25 (%)
B1	B0	B2	CIFAR-10	88.84	89.58	89.46	89.50	89.23		89.52			89.85	■
B2	B1	B3		90.63	91.14	90.56	91.03	91.35	●	91.04			91.39	■
B3	B2	B4		91.69	92.46	92.20	92.22	92.54	●	92.43			92.63	■
B4	B3	B5		92.63	93.26	92.81	93.46	■	92.80	93.35	●		92.95	
B5	B4	B6		93.02	93.95	93.62	94.11	●	93.92	93.56			94.24	■
B6	B5	B7		93.18	94.24	93.89	94.00	93.99		93.91			93.92	
V2M	V2S	V2L		92.09	92.47	92.30	92.66	■	92.54	●			92.23	
B1	B0	B2	CIFAR-100	64.93	66.39	65.95	66.33	66.64	▲	66.73	●		66.80	■
B2	B1	B3		68.39	69.30	68.57	69.61	▲	69.80	■			68.53	
B3	B2	B4		71.27	72.96	73.22	72.82	73.31	■	72.65			72.33	
B4	B3	B5		71.35	72.99	72.76	73.76	■	73.39	▲	73.56	●	73.25	▲
B5	B4	B6		72.63	74.67	74.21	74.85	▲	75.03	●	74.70	▲	75.20	■
B6	B5	B7		73.11	74.55	74.66	75.07	■	74.82	▲	74.96	●	74.10	
V2M	V2S	V2L		69.82	70.80	69.95	71.58	●	71.03	▲	71.69	■	71.10	▲

Table 7. The accuracy of comparison of ERKD with various teachers

Student model	Dataset	Teacher 1 and 2 (%)	Teacher 1 and 4 (%)	Teacher 2 and 3 (%)	Teacher 3 and 4 (%)
B2	CIFAR-10	89.79	89.67	89.79	89.85 ■
B3		91.34	91.34	91.56 ■	91.39
B4		92.87	92.83	92.91 ■	92.63
B5		93.43	93.34	93.46 ■	93.46 ■
B6		93.61	94.05	94.15	94.24 ■
B7		93.94	94.28 ■	94.04	94.00
V2L		92.65	92.63	92.63	92.66 ■
B2	CIFAR-100	68.77 ■	67.59	67.81	66.80
B3		70.42 ■	69.60	70.19	69.80
B4		74.01	74.49 ■	73.93	73.31
B5		74.27	74.50 ■	73.98	73.76
B6		75.80	75.14	76.19 ■	75.20
B7		75.92 ■	75.62	75.60	75.07
V2L		72.13	72.30 ■	71.39	71.69

When we tried to compare the accuracy of models trained with ERKD with a stronger model trained with STL, we found a surprising result that sometimes a weaker model with ERKD can be stronger than a stronger model with STL. For example, we compared the performance of EfficientNet B2 model using ERKD with the performance of the EfficientNet model using the STL method. The results can be seen in Table 8, which shows that some models with certain datasets can beat stronger models. The result with the icon in Table 8 indicates that ERKD outperforms the STL of a one-level higher robust model.

Similarly, the result with the icon in Table 9 shows that ERKD outperforms the STL of a two-level higher robust model. The square indicates the best accuracy, the circle indicates the second-best accuracy, the equilateral triangle indicates the third-best accuracy, and the right triangle indicates the fourth-best accuracy. For CIFAR-10 dataset; EfficientNet models B4, B5, and B6 with ERKD can outperform EfficientNet models B5, B6, and B7. Meanwhile for CIFAR-100 dataset; EfficientNet models B2, B4, B5, and B6 can outperform EfficientNet models B3, B5, B6, and B7.

Table 8. The accuracy of comparison between ERKD and STL model with one level higher robust model

Student model	Dataset	STL 1 level higher (%)	Teacher 1 and 2 (%)	Teacher 1 and 4 (%)	Teacher 2 and 3 (%)	Teacher 3 and 4 (%)
B2	CIFAR-10	90.63	89.79	89.67	89.79	89.85
B3		91.69	91.34	91.34	91.56	91.39
B4		92.63	92.87 ●	92.83 ▲	92.91 ■	92.63
B5		93.02	93.43 ●	93.34 ▲	93.46 ■	93.46 ■
B6		93.18	93.61 ▲	94.05 ▲	94.15 ●	94.24 ■
B2		68.39	68.77 ■	67.59	67.81	66.80
B3	CIFAR-100	71.27	70.42	69.60	70.19	69.80
B4		71.35	74.01 ●	74.49 ■	73.93 ▲	73.31 ▲
B5		72.63	74.27 ●	74.50 ■	73.98 ▲	73.76 ▲
B6		73.11	75.80 ●	75.14 ▲	76.19 ■	75.20 ▲

Table 9. The accuracy of comparison between ERKD and STL model with two levels higher robust model

Student model	Dataset	STL 2 level higher (%)	Teacher 1 and 2 (%)	Teacher 1 and 4 (%)	Teacher 2 and 3 (%)	Teacher 3 and 4 (%)
B2	CIFAR-10	91.69	89.79	89.67	89.79	89.85
B3		92.63	91.34	91.34	91.56	91.39
B4		93.02	92.87	92.83	92.91	92.63
B5		93.18	93.43 ●	93.34 ▲	93.46 ■	93.46 ■
B2		71.27	68.77	67.59	67.81	66.80
B3	CIFAR-100	71.35	70.42	69.60	70.19	69.80
B4		72.63	74.01 ●	74.49 ■	73.93 ▲	73.31 ▲
B5		73.11	74.27 ●	74.50 ■	73.98 ▲	73.76 ▲

We also tried to compare ERKD with the two-level stronger models with STL. For example, the performance of the EfficientNet B2 model using ERKD is compared with the performance of the EfficientNet B4 model using the STL method. The results can be seen in Table 9. Surprisingly, we still found that some weaker models can be stronger with ERKD than the two-level stronger models with STL. Using ERKD, EfficientNet model B5 with the CIFAR-10 dataset performs better than EfficientNet model B7 with the CIFAR-10 dataset. In addition, EfficientNet Models B4 and B5 with CIFAR-100 dataset using ERKD also perform better than EfficientNet models B6 and B7. These two surprising results prove that ERKD effectively improves model performance.

#### 4. CONCLUSION

All experiments proved that ERKD can improve the model's performance. The model's performance with the ERKD method can be better than the STL and single-teacher methods. It can also be better than the STL method's one or two-level, stronger model. Thus, the ERKD method is suitable for supervising stronger models using weaker models. This study also proved that the ERKD method can improve the model's performance even though the weak and strong models' architectures are different. The EfficientNet models can still outperform even when assisted by other CNN models. Despite using weaker AI instead of human, the result of this study shows a glimmer of hope that an AI with stronger intelligence than human can still be supervised by humans. The trick is to have several humans to collaborate in managing a super-alignment model. Future studies could investigate a similar study but without using the trained model. They could also investigate ERKD methods in other computer vision tasks, such as image detection or image segmentation. In addition, they can also experimented on using more than two weaker models to supervise a stronger model to get the optimal number of weaker models.

#### FUNDING INFORMATION

Authors state there is no funding involved.

#### AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Christopher Gavra Reswara	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Tjeng Wawan Cenggoro	✓									✓		✓		

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal Analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review &amp; Editing

Vi : Visualization

Su : Supervision

P : Project Administration

Fu : Funding Acquisition

## CONFLICT OF INTEREST STATEMENT

Authors state there is no conflict of interest.

## DATA AVAILABILITY

No new data were generated or analyzed during this study.

## REFERENCES




- [1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
- [2] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.
- [3] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 6105–6114, 2019.
- [4] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.
- [5] A. Howard *et al.*, "Searching for MobileNetV3," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1314–1324, doi: 10.1109/ICCV.2019.00140.
- [6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.
- [7] A. Krizhevsky, "Learning multiple layers of features from tiny images," *M.Sc. Thesis*, Department of Computer Science, University of Toronto, Toronto, Canada, 2009.
- [8] L. Bossard, M. Guillaumin, and L. V. Gool, "Food-101—mining discriminative components with random forests," in *Computer Vision - European Conference on Computer Vision (ECCV)*, pp. 446–461, 2014, doi: 10.1007/978-3-319-10599-4\_29.
- [9] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, 2008, pp. 722–729, doi: 10.1109/ICVGIP.2008.47.
- [10] T. Berg, J. Liu, S. W. Lee, M. L. Alexander, D. W. Jacobs, and P. N. Belhumeur, "Birdsnap: Large-scale fine-grained visual categorization of birds," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2019–2026, doi: 10.1109/CVPR.2014.259.
- [11] J. Achiam *et al.*, "GPT-4 technical report," *arXiv-Computer Science*, Mar. 2023.
- [12] P. Georgiev *et al.*, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," *arXiv-Computer Science*, Dec. 2024.
- [13] A. Grattafiori *et al.*, "The Llama 3 herd of models," *arXiv-Computer Science*, Nov. 2024.
- [14] A. Glaese *et al.*, "Improving alignment of dialogue agents via targeted human judgements," *arXiv-Computer Science*, Sep. 2022.
- [15] Y. Bai *et al.*, "Training a helpful and harmless assistant with reinforcement learning from human feedback," *arXiv-Computer Science*, Apr. 2022.
- [16] L. Ouyang *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27730–27744, 2022.
- [17] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, pp. 529–533, Feb. 2015, doi: 10.1038/nature14236.
- [18] D. Silver *et al.*, "Mastering the game of go without human knowledge," *nature*, vol. 550, pp. 354–359, Oct. 2017, doi: 10.1038/nature24270.
- [19] D. Silver *et al.*, "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018, doi: 10.1126/science.aar6404.
- [20] D. Guo *et al.*, "Deepseek-R1: Incentivizing reasoning capability in llms via reinforcement learning," *arXiv-Computer Science*, pp. 1–22, Jan. 2025.
- [21] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.
- [22] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.
- [23] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv-Statistics*, pp. 1–9, Mar. 2015.
- [24] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *Proceedings of the 38th International Conference on Machine Learning (ICML)*, Apr. 2021, pp. 10096–10106.






- [25] L. F. -Fei, J. Deng, and K. Li, "ImageNet: Constructing a large-scale image database," *Journal of Vision*, vol. 9, no. 8, pp. 1037–1037, 2009, doi: 10.1167/9.8.1037.
- [26] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, pp. 211–252, 2015, doi: 10.1007/s11263-015-0816-y.
- [27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv-Computer Science*, pp. 1-15, Jan. 2017.
- [28] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10425–10433, doi: 10.1109/CVPR42600.2020.01044.
- [29] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 11966–11976, doi: 10.1109/CVPR52688.2022.01167.

## BIOGRAPHIES OF AUTHORS



**Christopher Gavra Reswara**    received his bachelor's degree in Computer Science from Bina Nusantara University, where he is pursuing a master's degree in the same field. He also works as a programmer at the Bina Nusantara IT Division. His research focuses on artificial intelligence, recommendation systems, and computer vision, and he has authored two conference papers on recommendation systems. He can be contacted at email: christopher.reswara@binus.ac.id.



**Tjeng Wawan Cenggoro**    received a bachelor's degree in Information Technology from STMIK Widya Cipta Dharma and a master's degree in Information Technology from Bina Nusantara University. He is currently an AI researcher focusing on developing deep learning algorithms for applications in computer vision, natural language processing, and bioinformatics. He is also an NVIDIA Deep Learning Institute certified instructor. Throughout his 9+ year career, he has led numerous research projects related to AI and data science, with applications in many domains such as e-commerce, agriculture, and health. He has published over 80 peer-reviewed publications and reviewed for prestigious journals, such as Scientific Reports, IEEE Access, and PLOS ONE. In addition to this, he also holds 4 copyrights for AI-based video/image analytics software. He can be contacted at email: tjeng.cenggoro@binus.ac.id.