Journal of Advanced Health Informatics Research (JAHIR)
Vol. 2, No. 2, (2024), pp. 64-74
ISSN: 2985-6124, DOI: 10.59247/jahir.v2i2.289

Comparison of Transfer Learning Performance in Lung and Colon Classification with Knowledge Distillation

Annastasya Nabila Elsa Wulandari [1,*], Aimar Yudhistira [2], Purwono[3], Abdel-Nasser Sharkawy [4]
[1-3]*Informatics, Universitas Harapan Bangsa, Indonesia*
[3] *Mechatronics Engineering, Mechanical Engineering Department, Faculty of Engineering, South Valley University, Qena 83523, Egypt, Egypt*

ARTICLE INFO

Article history:

Received May 03, 2024
Revised August 24, 2024
Published August 29, 2024

Keywords:

Lung;
Colon;
Knowledge;
Distillation;
Transfer Learning

ABSTRACT

This research aims to apply the knowledge distillation method to medical image classification, specifically in the case of lung and colon image classification using various transfer learning models. Knowledge distillation allows the transfer of knowledge from a larger model (teacher) to a smaller model (student), which enables more efficient model building without sacrificing accuracy. In this research, the DenseNet169 model is used as the teacher model. The student model uses several alternative transfer learning architectures such as DenseNet121, MobileNet, ResNet50, InceptionV3, and Xception. The data used consists of 25,000 histopathology images that have been processed and divided into training, validation, and test data. Data augmentation was performed to enlarge the dataset from 750 to 25,000 images, which helped improve the performance of the model. Model performance evaluation was performed by measuring the accuracy and loss value of each student model compared to the teacher model. The results showed that the student models generated through the knowledge distillation process performed close to or even exceeded the teacher model in some cases, with the Xception model showing the highest accuracy of 96.95%. In conclusion, knowledge distillation is effective in reducing model complexity without compromising performance, which is particularly beneficial for implementation on resource-constrained devices.

Corresponding Author:

Annastasya Nabila Elsa Wulandari, Informatics, Universitas Harapan Bangsa, Indonesia
Email: anstsya.new@gmail.com

1. INTRODUCTION

The classification of medical images such as lung and colon plays an important role in the healthcare field [1]. This classification allows healthcare professionals to diagnose diseases such as cancer, infections, or other conditions more accurately and efficiently [2]. It also contributes to the development of early detection and disease monitoring methods, potentially saving patients' lives [3]. Understanding the classification of lung and colon has significant implications for the prevention, diagnosis, and treatment of serious diseases [4].

Knowledge Distillation (KD) in machine learning involves transferring learning from a teacher model that has a large size to a student model that has a smaller size [5]. This process uses the soft prediction of the teacher model to train the student model, allowing it to learn not only from the original data but also from the probabilistic output of the teacher model [6]. This technique is essential for implementing models on resource-

constrained devices such as smart phones or embedded systems, as it enables the creation of smaller and faster models without sacrificing accuracy, thus improving the efficiency and performance of real-world applications, especially critical in fields such as medical image recognition. [7].

Advances in deep learning technology have had a significant impact on the classification of lung and colon cancers, offering a promising methodology for improving diagnostic accuracy and efficiency. Byeon et al. [8] developed and validated a deep learning model to classify digital pathology images of colon lesions and achieved an average diagnostic accuracy of up to 97.3%. Talukder et al. [9] introduced a hybrid ensemble feature extraction model for lung and colon cancer detection, so as to achieve a high degree of accuracy, which can support clinical diagnosis. Balcı et al. [10] presents a novel approach that combines 3D image analysis and series classification for lung nodule images with an accuracy value of 92.84. Alshmrani et al. [11] proposed deep learning architecture for multi-class lung disease classification, including COVID-19, with superior performance metrics. Research focusing on lung cancer detection using computational intelligence techniques on CT scans reported the modified AlexNet-SVM classification model managed to get an accuracy value of 97.98% [12]. Research by Pandit et al. [13] conducted research to improve prediction accuracy and reduce lung tumor recognition processing time using multispace images in the pooling layer of convolutional neural networks and achieved a classification accuracy value of 99.5%.

KD is a technique for compressing models and improving efficiency by transferring knowledge from complex models to simple models without compromising performance [14]. It is important in medical image classification to implement lightweight models on resource-constrained devices that enable fast and accurate diagnosis in clinical settings or with portable medical devices. This method maintains the predictive quality of deep models while reducing size and computational demands.

Several previous studies have used KD methods to produce lighter models for various classification tasks. Multi-target knowledge distillation via student self-reflection (MTKD-SSR) improves performance on visual recognition tasks by enhancing teachers' knowledge disclosure and students' knowledge digestibility [15]. Neuron manifold distillation (NMD) emulates teacher output distribution and learning feature geometry, exhibiting a consistent accuracy-speed trade-off [15]. In graph neural networks (GNN), KD improves performance by learning node-specific distillation temperatures, demonstrating its flexibility and applicability across different model architectures [16]. Generalized knowledge distillation (GKD) facilitates transfer of learning between different or overlapping task domains, emphasizing the versatility of the KD [17]. KD enables efficient model deployment on edge devices for tasks that require real-time processing, such as medical image classification, important in diagnosing critical conditions such as lung and colon cancer, where fast and accurate classification can have a major impact on patient outcomes [18].

This research aims to apply the knowledge distillation method to medical image classification, especially in the case of lung and colon image classification using various transfer learning models. This research contribution is carried out with the main purpose of compressing complex models into simpler models while maintaining a high level of accuracy in the lung and colon classification work. The model created is expected to be applied to devices with limited resources, such as mobile devices or edge devices.

The research question to be answered is to find out how well the classification performance with KD techniques is accompanied by conducting a comparative analysis of several popular transfer learning models namely DenseNet, MobileNet, InceptionV3, Xception and Resnet50. By answering these questions, this research is expected to make a valuable contribution in the development of more efficient and effective medical image classification methods, as well as increase our understanding of the potential of knowledge distillation in improving the performance of models used in healthcare applications.

2. METHODS

The framework proposed in this research includes several stages from data preparation to comparing the performance of each transfer learning model which can be seen in Figure 2.

2.1. Dataset

This dataset contains 25,000 histopathology images divided into 5 classes. All images are 768 x 768 pixels in size and jpeg format. The images were derived from an initial sample of HIPAA-compliant and validated sources, consisting of a total of 750 lung tissue images (250 benign lung tissue, 250 lung adenocarcinoma, and 250 lung squamous cell carcinoma) and a total of 500 colon tissue images (250 benign colon tissue and 250 colon adenocarcinoma), which were then expanded to 25,000 using the Augmentor package. This dataset consists of five classes, each having 5,000 images, viz: benign lung tissue, lung adenocarcinoma, lung squamous cell carcinoma, colon adenocarcinoma, and benign colon tissue. The dataset can be accessed at the

following link: *https://www.kaggle.com/datasets/andrewmvd/lung-and-colon-cancer-histopathological-images*.

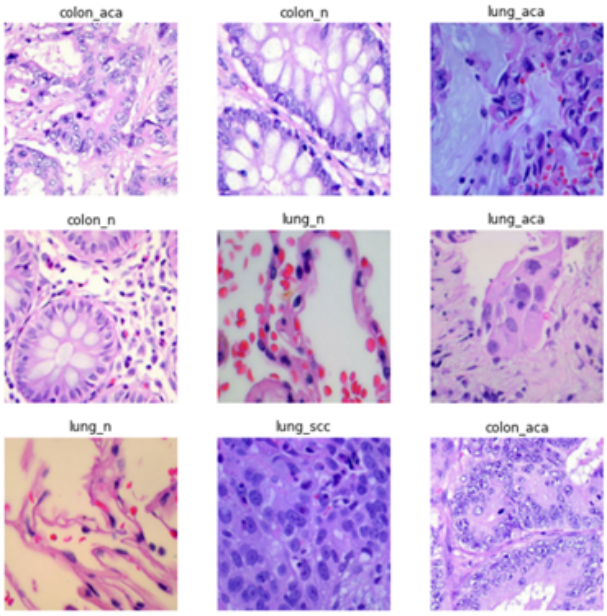


Fig. 1. Lung and Colon Dataset.

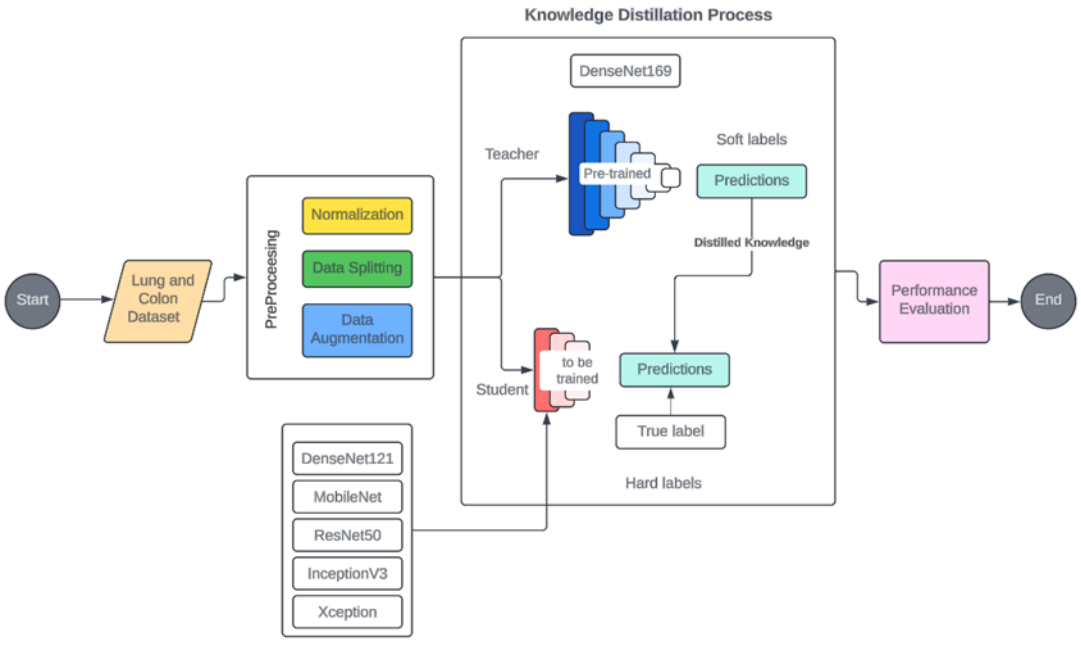


Fig. 2. Flow of proposed research framework

2.2. Preproceesing

The preprocessing of the image dataset is done before feeding the data into the model for training. The first step is normalization, where the pixel value ranges from 0-255 is converted to 0-1 using the Rescaling layer of TensorFlow. After that, the dataset was divided into three parts: training data, validation data, and test

data. The division is done with a ratio of 80% for training data, 20% for test data, and validation data is taken from the training dataset by taking 20% of the training data. The loaded data is then cached to speed up access, randomized in each batch, and refetched to prepare for the next batch during training. Finally, a garbage collection technique is performed to free up memory from objects that are no longer used, thus ensuring efficient memory usage during the training process.

2.3. Model Architecture

The proposed architecture involves the use of knowledge distillation from the teacher model to the student model. In this process, the teacher model will perform transfer learning to the student model. Transfer Learning is a technique in machine learning where a model that has been trained on one task is reused as a starting point for training on another different task [19]. This technique is especially useful when the training data for a new task is limited or insufficient to train a model from scratch.

2.3.1 Teacher Model

The teacher architecture model uses DenseNet169, a type of convolutional neural network architecture consisting of several interconnected dense blocks [20]. The mathematical equation of DenseNet169 can be seen in Equation 1.

$$\mathbf{y} = \mathbf{x} + \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{F}(\mathbf{x_i}) \tag{1}$$

DenseNet169 connects each layer with every other layer in a feed-forward fashion. The equation expresses the output *y* as the sum of the input *x* and the *F* unction applied to all previous layers. This *F* function represents a non-linear operation that is usually a combination of convolution, batch normalization, and ReLU activation operations. This approach helps mitigate the vanishing gradient problem and allows reutilization of features from previous layers, leading to better parameter efficiency and improved performance [21].

After the last block of DenseNet169, the GlobalAveragePooling2D layer is added to calculate the spatial average of the output features of each channel. This results in a feature vector that is reduced in dimension but retains important information. This is followed by a Dense layer containing 512 units and a ReLU activation function, aiming to learn a more abstract representation of the features. To reduce overfitting during training, Dropout is used with a dropout rate of 0.5 to randomly deactivate some units.

The training configuration in the teacher model uses the Adam optimizer which is efficient and often used in training neural networks. The loss function used is sparse categorical crossentropy, suitable for multi-class classification with labels that are not encoded to be one-hot. The metric used is accuracy, measuring how well the model can predict the image class from the test dataset.

2.3.2 Student Model

The student model uses several popular transfer learning types such as DenseNet121, MobileNet, Resnet50, InceptionV3 and Xception. DenseNet121 is one of the variants of the DenseNet (Densely Connected Convolutional Networks) architecture, which was introduced by Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger in 2017. DenseNet is designed to improve the efficiency of using the features extracted by the layers in the network [22]. In the DenseNet121 architecture, each layer is directly connected to all subsequent layers. This means that the features generated by each layer are used as inputs for all subsequent layers, which helps in reducing the gradient gradation problem and makes the network more efficient in utilizing the features.

A dense block in DenseNet121 consists of multiple convolution layers with small kernel sizes (typically 3x3), and each convolution layer takes input from all previous layers in the block [23]. With this principle, DenseNet121 can reutilize all the features generated by the previous layers, thus enabling high efficiency and improved performance on various pattern recognition and image classification tasks.

MobileNet is a convolutional neural network architecture designed by Google for use on mobile devices and embedded applications with limited computing resources [24]. MobileNet aims to provide a good balance between performance and efficiency, enabling the use of deep learning on devices with low processing power and limited power consumption. In general, the MobileNet architecture consists of a series of depthwise separable convolution blocks followed by several standard convolution layers and a pooling layer.

ResNet50 (50-layer Residual Network) is one of the very popular and successful convolutional neural network architectures, which was introduced by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun in

the paper "Deep Residual Learning for Image Recognition" in 2015 [25]. ResNet50 is known for its ability to train very deep networks with hundreds of layers without suffering from degradation issues common in very deep networks.

InceptionV3 is one of the convolutional neural network architectures developed by Google, introduced in the paper "Rethinking the Inception Architecture for Computer Vision" by Christian Szegedy in 2015 [26]. InceptionV3 is an advanced and more efficient version of previous Inception architectures (including GoogLeNet and InceptionV1/V2 [27].

Xception is a convolutional neural network architecture introduced by François Chollet in the paper "Xception: Deep Learning with Depthwise Separable Convolutions" in 2017 [28]. Xception stands for "Extreme Inception" and is an extension of the Inception architecture. Xception replaces the standard Inception module with depthwise separable convolutions, which aims to improve model efficiency and performance.

2.4. Knowledge Distillation

Knowledge Distillation (KD) is a method used to transfer knowledge from larger and complex models to smaller and simpler models [29]. The main goal of knowledge distillation is to produce smaller models with performance close to or even equivalent to that of larger models [30]. This process enables the use of models that are more efficient in terms of computation and memory, making them suitable for implementation on resource-constrained devices [31].

In this experiment, the transfer learning architecture uses DenseNet169 as the teacher model and several types of transfer learning architectures as the student model. DenseNet169 was chosen because of its superior ability to extract features from data with complex structures, while DenseNet121, MobileNet, ResNet50, InceptionV3 and Xception were chosen as student models because they have simpler and lighter architectures.

The distillation process is done by transferring knowledge from the teacher model to the student model. This is achieved by comparing the output of the two models using a predetermined loss function, namely Sparse Categorical Crossentropy. The Sparse Categorical Crossentropy equation can be seen in Formula 2.

$$\boldsymbol{L_{SCE}} = -\sum_{\boldsymbol{i}} \mathbf{log}\,(\boldsymbol{q_i}, \boldsymbol{y_i}) \tag{2}$$

Where $\boldsymbol{q_i}, \boldsymbol{y_i}$ is the probability predicted by the student model for the original label $\boldsymbol{y_i}$.

Sparse Categorical Crossentropy is a loss function used in classification problems, especially when the output is an integer label and not a one-hot encoded vector [32]. It is a variant of the Categorical Crossentropy function that is more efficient and practical in situations where we have integer labels. This loss function not only considers the prediction error of the student model against the original label, but also the prediction error against the probability distribution generated by the teacher model [33].

In addition to using loss functions, KD also uses the Kullback-Leibler (KL) Divergence method to measure how well the student model output distribution approaches the teacher model output distribution that has been softened using temperature ($\boldsymbol{\tau}$) [34]. The equation for KL Divergence can be seen in the Formula 3.

$$\boldsymbol{L_{KD}} = \boldsymbol{\tau^2} \sum_{\boldsymbol{i}} \boldsymbol{p_i^T} \mathbf{log}\,(\frac{\boldsymbol{p_i^T}}{\boldsymbol{q_i^T}}) \tag{3}$$

Where $\boldsymbol{p_i^T}$ and $\boldsymbol{q_i^T}$ are the probability distributions of teacher and student with softening using temperature $\boldsymbol{\tau}$. The total loss function is a combination of Sparse Categorical Cross-Entropy Loss and KL Divergence as shown in Formula 4.

$$\mathbf{L} = \boldsymbol{\alpha L_{SCE}} + \mathbf{1}(\mathbf{1} - \boldsymbol{\alpha})\boldsymbol{L_{KD}} \tag{4}$$

Here, α is a weighting factor that controls the contribution of both loss components. By choosing appropriate values of α and τ, the student model is trained to emulate the behavior of the teacher model while maintaining good performance on the original training data.

The weight optimization of the student model is done using Adam's optimizer, which is known for its adaptive ability in setting the learning rate for each parameter. This optimization process aims to minimize the loss function so that the student model can approach the performance of the teacher model [35].

In addition, the proposed distillation model uses callbacks to save the best weights of the model during the training process [36]. This means that whenever there is an improvement in performance on the validation data, the model weights will be saved. This technique helps in preventing overfitting and ensures that the resulting model is the best one possible.

2.5. Model Evaluation

In this research, the model evaluation process is conducted through a final evaluation on a pre-prepared test dataset. The three models evaluated include the teacher model, the distilled student model, and the student model trained from scratch. This evaluation aims to comprehensively compare the relative performance of the three models. The evaluation process involves measuring the accuracy of each model on the test dataset, which is then displayed as a bar graph.

The bar graph serves to provide a clear visual understanding of how well the model generated from the distillation process performs compared to the reference model (teacher) and the model trained from scratch. As such, it not only displays accuracy figures, but also facilitates visual analysis that makes interpretation of the results easier.

Model evaluation is an important stage in this research as it allows us to measure the effectiveness of the knowledge distillation method in producing smaller models that still perform well. The results of this evaluation provide insight into how knowledge distillation can maintain or even improve the accuracy of student models compared to teacher models and models trained from scratch. In addition, this evaluation can also reveal potential weaknesses or areas of improvement of the distillation method used, thus providing direction for further research and development.

3. RESULTS AND DISCUSSION

According to the research methods conducted, the results of this study show how each stage in the proposed framework contributes to the achievement of the research objectives. Experimental results for the training of the teacher model (DenseNet169), student models (DenseNet121, MobileNet, ResNet50, InceptionV3 and Xception) trained from scratch, and student models generated from the knowledge distillation process. Each model is evaluated based on its training and validation performance, measured using accuracy and loss metrics. To provide a comprehensive overview, the results are visualized through accuracy and loss curves over the training epochs, as well as through a table summarizing the accuracy and loss values at the end of training.

Figure 4 shows the training and validation accuracy curves for each model. These curves help visualize how the model performance improves as the number of epochs increases. The DenseNet121 student model during training has almost the same accuracy value as the DenseNet169 teacher model. The DenseNet121 student model also generally outperformed the performance of the student model built from scratch without using transfer learning during validation. The MobileNet student model during training has a higher accuracy value than the DenseNet169 teacher model. The MobileNet student model also generally outperformed the performance of the student model built from scratch without using transfer learning during validation. The ResNet50 student model during training has an accuracy value that is equal to the teacher model, DenseNet169. The ResNet50 student model also generally outperformed the performance of the student model built from scratch without using transfer learning during validation. The InceptionV3 student model during training has an accuracy value that is equal to the teacher model, DenseNet169. The InceptionV3 student model has also generally outperformed the performance of the student model built from scratch without using transfer learning during validation. The Xception student model during training has a higher accuracy value than the DenseNet169 teacher model. The Xception student model has also generally outperformed the performance of the student model built from scratch without using transfer learning during validation.

Figure 5 displays the distiller loss curve, which illustrates the decrease in loss value during the training process. Lower loss values indicate that the model is able to minimize its prediction error. All transfer learning models applied to the student model show lower loss values than the teacher model and student stracth model. The comparison between the teacher and student models will provide insight into the efficiency of the distillation process in reducing loss without overfitting.

Table 1 summarizes the training performance of the models by presenting the final values of accuracy and loss on both the training and validation data. These values provide a quantitative indication of how well each model learns from the given data and how well they can generalize to data they have not seen before. The data shows that the performance of the Xception model is the model with the highest accuracy value in the student model.

Table 1. Comparison of Student Model Accuracy Value.

Transfer Learning Student Model	Accuracy
DenseNet121	0.8074
MobileNet	0.9432
ResNet50	0.8905
InceptionV3	0.8905
Xception	**0.9695**

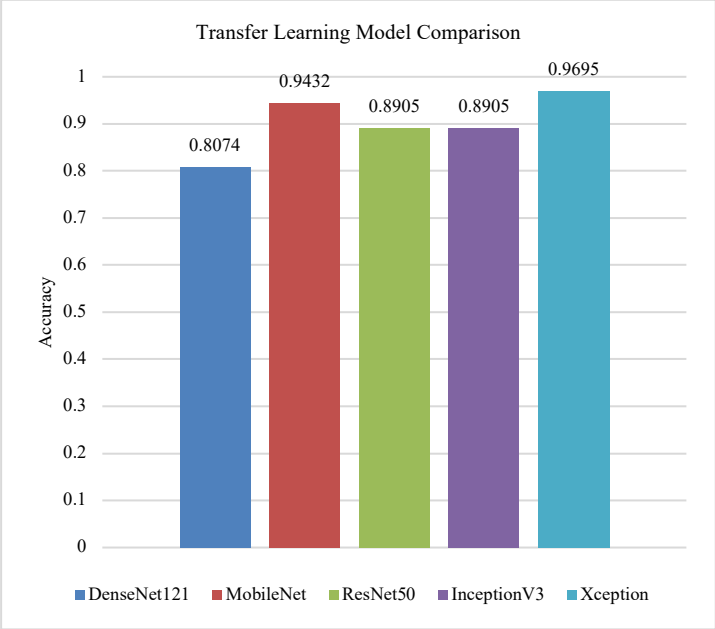
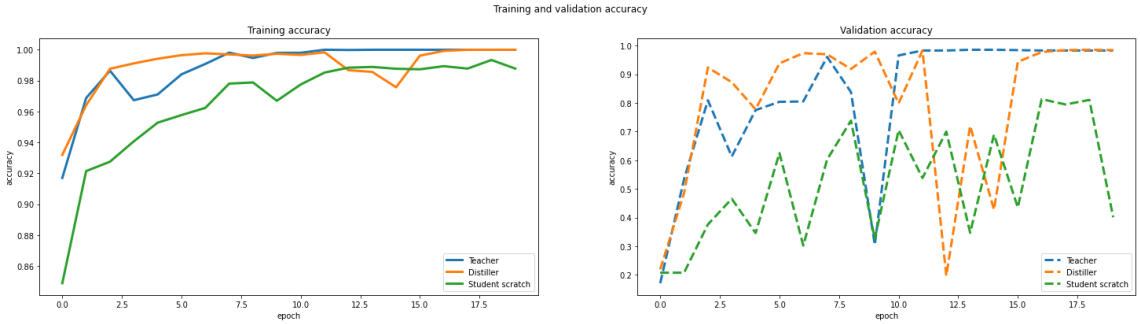


Fig. 3. Comparison Chart of Transfer Learning Model on Student Model

These training results will form the basis for further evaluation on test datasets, which will help determine the effectiveness of the knowledge distillation method in generating smaller models that still perform well. By analyzing the training results in detail, we can understand the strengths and weaknesses of the proposed approach and provide recommendations for future research.



(a)

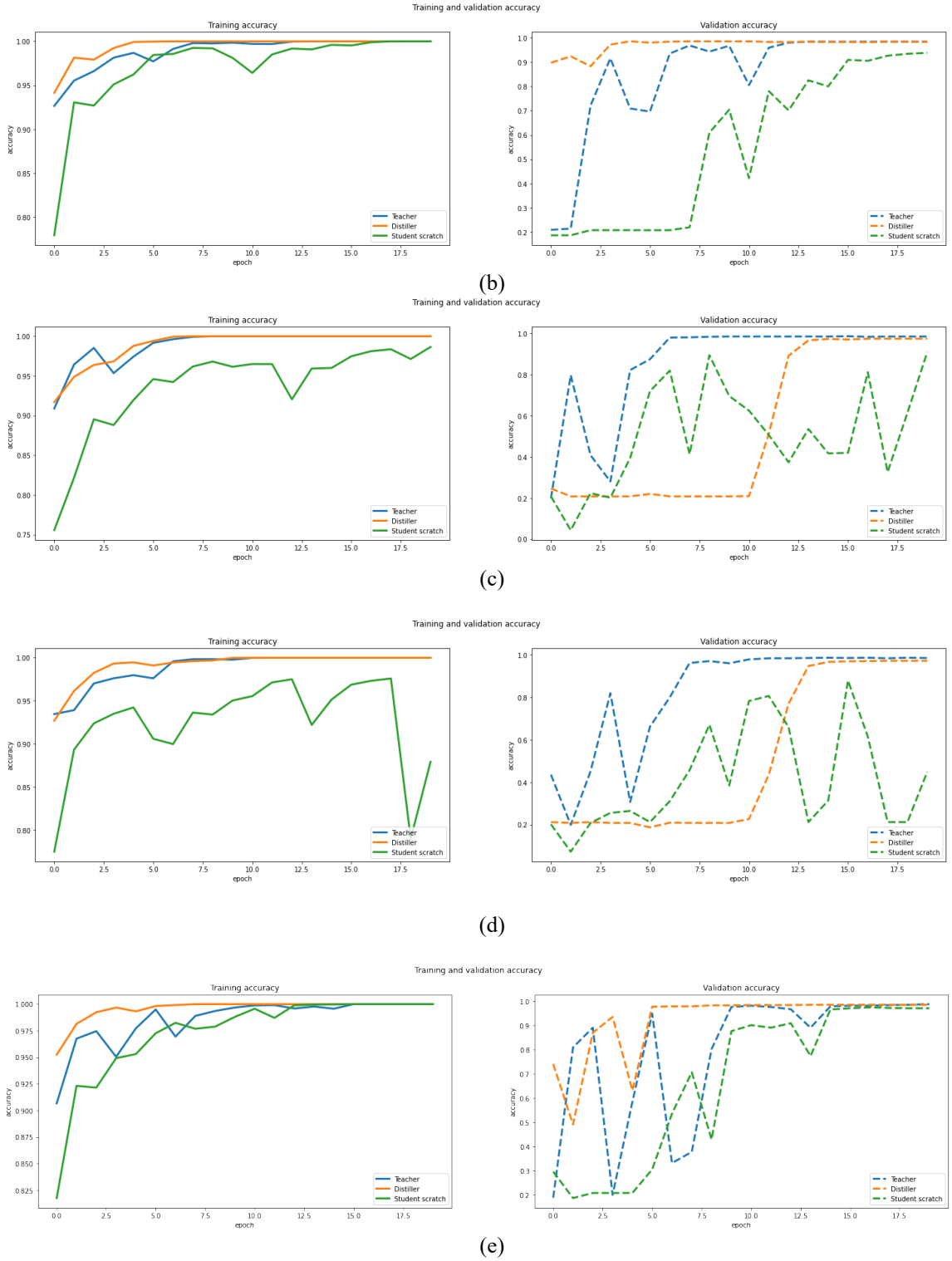


Fig. 4. Training dan Validation Accuracy from (a) DenseNet121, (b) MobileNet, (c) ResNet50, (d) InceptionV3 and (d) Xception

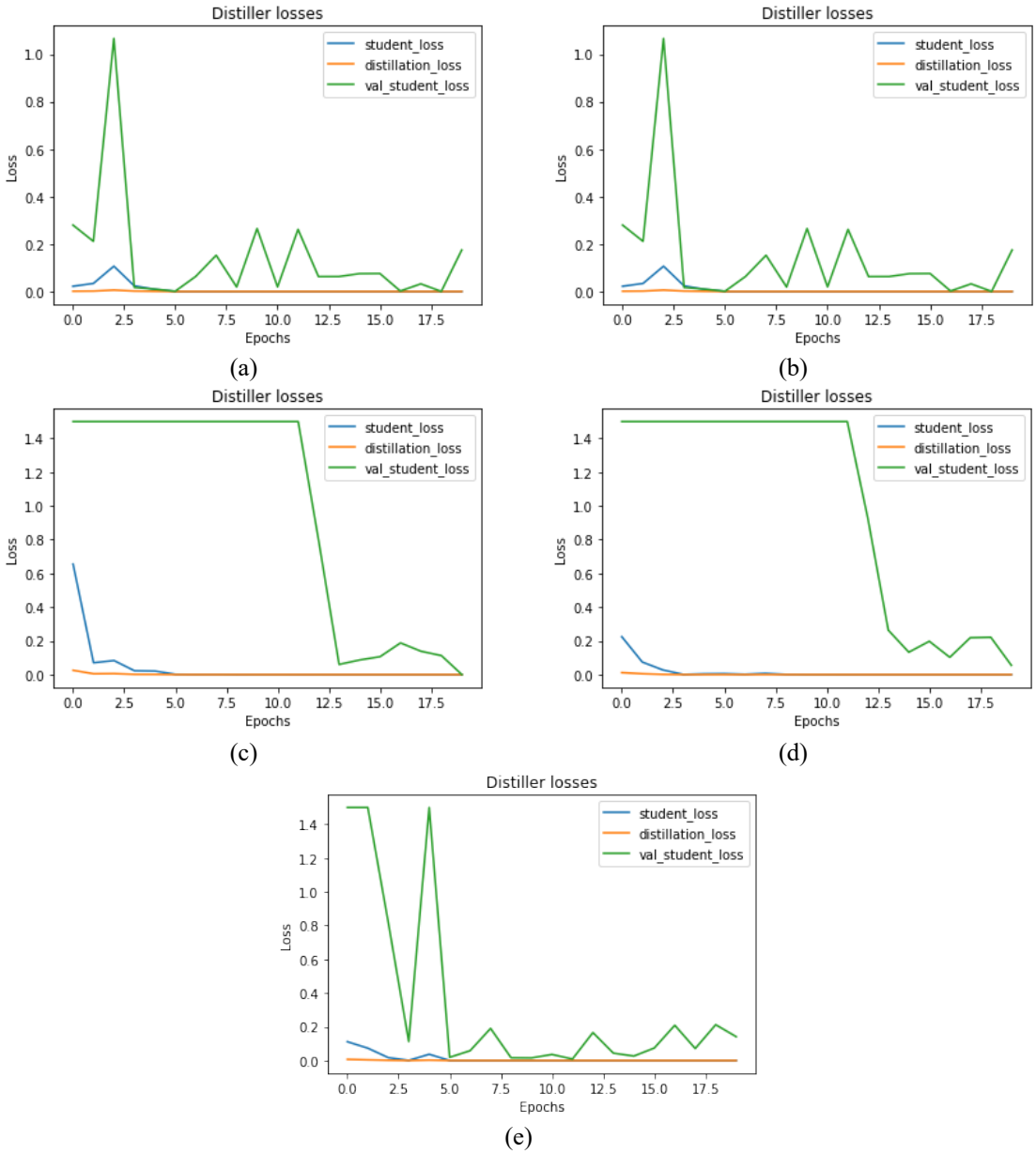


Fig. 5. Distiller losses from (a) DenseNet121, (b) MobileNet, (c) ResNet50, (d) InceptionV3, (e) Xception

4. CONCLUSION

This research successfully demonstrates that the knowledge distillation method can be effectively used to compress complex models into simpler models without sacrificing performance. Using DenseNet169 as the teacher model and various transfer learning architectures as student models (DenseNet121, MobileNet, ResNet50, InceptionV3, and Xception), this research evaluates the performance of each model based on accuracy and loss value. The results showed that all student models generated through the KD process were able to achieve accuracy close to or even exceeding the teacher model in some cases. In particular, the Xception model as a student achieved the highest accuracy of 96.95%. The evaluation process also revealed that KD can maintain or even improve the accuracy of student models compared to teacher models and models trained from scratch without transfer learning. In conclusion, KD is an efficient method to reduce the size and computational demands of deep learning models, which is particularly beneficial for applications on devices with limited resources such as mobile or edge devices.

REFERENCES

[1] A. Hosny, C. Parmar, J. Quackenbush, L. H. Schwartz, and H. J. W. L. Aerts, "Artificial intelligence in radiology," *Nat Rev Cancer*, vol. 18, no. 8, pp. 500–510, 2018, doi: 10.1038/s41568-018-0016-5.

[2] S. Tresker, "A typology of clinical conditions," *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences*, vol. 83, p. 101291, 2020, doi: https://doi.org/10.1016/j.shpsc.2020.101291.

[3] B. Jiang, D. Xie, S. Wang, X. Li, and G. Wu, "Advances in early detection methods for solid tumors," *Front Genet*, vol. 14, 2023, [Online]. Available: https://www.frontiersin.org/journals/genetics/articles/10.3389/fgene.2023.1091223

[4] M. Mamun, M. I. Mahmud, M. Meherin, and A. Abdelgawad, "LCDctCNN: Lung Cancer Diagnosis of CT scan Images Using CNN Based Model," in *2023 10th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2023, pp. 205–212. doi: 10.1109/SPIN57001.2023.10116075.

[5] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge Distillation: A Survey," *Int J Comput Vis*, vol. 129, no. 6, pp. 1789–1819, 2021, doi: 10.1007/s11263-021-01453-z.

[6] F. Sarfraz, E. Arani, and B. Zonooz, "Knowledge Distillation Beyond Model Compression," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 6136–6143. doi: 10.1109/ICPR48806.2021.9413016.

[7] J. Nayem *et al.*, "Few Shot Learning for Medical Imaging: A Comparative Analysis of Methodologies and Formal Mathematical Framework," in *Data Driven Approaches on Medical Imaging*, B. Zheng, S. Andrei, M. K. Sarker, and K. D. Gupta, Eds., Cham: Springer Nature Switzerland, 2023, pp. 69–90. doi: 10.1007/978-3-031-47772-0_4.

[8] S. Byeon, J. Park, Y. A. Cho, and B.-J. Cho, "Automated histological classification for digital pathology images of colonoscopy specimen via deep learning," *Sci Rep*, vol. 12, no. 1, p. 12804, 2022, doi: 10.1038/s41598-022-16885-x.

[9] Md. A. Talukder, Md. M. Islam, M. A. Uddin, A. Akhter, K. F. Hasan, and M. A. Moni, "Machine learning-based lung and colon cancer detection using deep feature extraction and ensemble learning," *Expert Syst Appl*, vol. 205, p. 117695, 2022, doi: https://doi.org/10.1016/j.eswa.2022.117695.

[10] M. A. Balcı, L. M. Batrancea, Ö. Akgüller, and A. Nichita, "A Series-Based Deep Learning Approach to Lung Nodule Image Classification," *Cancers (Basel)*, vol. 15, no. 3, 2023, doi: 10.3390/cancers15030843.

[11] G. M. M. Alshmrani, Q. Ni, R. Jiang, H. Pervaiz, and N. M. Elshennawy, "A deep learning architecture for multi-class lung diseases classification using chest X-ray (CXR) images," *Alexandria Engineering Journal*, vol. 64, pp. 923–935, 2023, doi: https://doi.org/10.1016/j.aej.2022.10.053.

[12] I. Naseer, S. Akram, T. Masood, M. Rashid, and A. Jaffar, "Lung Cancer Classification Using Modified U-Net Based Lobe Segmentation and Nodule Detection," *IEEE Access*, vol. 11, pp. 60279–60291, 2023, doi: 10.1109/ACCESS.2023.3285821.

[13] B. R. Pandit *et al.*, "Deep learning neural network for lung cancer classification: enhanced optimization function," *Multimed Tools Appl*, vol. 82, no. 5, pp. 6605–6624, 2023, doi: 10.1007/s11042-022-13566-9.

[14] J. Gou, X. Xiong, B. Yu, L. Du, Y. Zhan, and D. Tao, "Multi-target Knowledge Distillation via Student Self-reflection," *Int J Comput Vis*, vol. 131, no. 7, pp. 1857–1874, 2023, doi: 10.1007/s11263-023-01792-z.

[15] Z. Tao, Q. Xia, S. Cheng, and Q. Li, "An Efficient and Robust Cloud-Based Deep Learning with Knowledge Distillation," *IEEE Transactions on Cloud Computing*, vol. 11, no. 02, pp. 1733–1745, 2023, doi: 10.1109/TCC.2022.3160129.

[16] A. Hoyle, P. Goel, and P. Resnik, *Improving Neural Topic Models using Knowledge Distillation*. 2020. doi: 10.18653/v1/2020.emnlp-main.137.

[17] C. Yang *et al.*, "Learning to Distill Graph Neural Networks," in *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, in WSDM '23. New York, NY, USA: Association for Computing Machinery, 2023, pp. 123–131. doi: 10.1145/3539597.3570480.

[18] H.-J. Ye, S. Lu, and D.-C. Zhan, "Generalized Knowledge Distillation via Relationship Matching," *IEEE Trans Pattern Anal Mach Intell*, vol. 45, no. 2, pp. 1817–1834, 2023, doi: 10.1109/TPAMI.2022.3160328.

[19] A. H. Ali, M. G. Yaseen, M. Aljanabi, S. A. Abed, and C. GPT, "Transfer Learning: A New Promising Techniques," *Mesopotamian Journal of Big Data*, pp. 29–30, Feb. 2023, doi: 10.58496/MJBD/2023/004.

[20] T. Zhou, X. Ye, H. Lu, X. Zheng, S. Qiu, and Y. Liu, "Dense Convolutional Network and Its Application in Medical Image Analysis," *Biomed Res Int*, vol. 2022, pp. 1–22, Apr. 2022, doi: 10.1155/2022/2384830.

[21] G. Huang, Z. Liu, G. Pleiss, L. van der Maaten, and K. Q. Weinberger, "Convolutional Networks with Dense Connectivity," *IEEE Trans Pattern Anal Mach Intell*, vol. 44, no. 12, pp. 8704–8716, Dec. 2022, doi: 10.1109/TPAMI.2019.2918284.

[22] G. Huang, Z. Liu, G. Pleiss, L. van der Maaten, and K. Q. Weinberger, "Convolutional Networks with Dense Connectivity," *IEEE Trans Pattern Anal Mach Intell*, vol. 44, no. 12, pp. 8704–8716, Dec. 2022, doi: 10.1109/TPAMI.2019.2918284.

[23] G. Huang, Z. Liu, G. Pleiss, L. van der Maaten, and K. Q. Weinberger, "Convolutional Networks with Dense Connectivity," *IEEE Trans Pattern Anal Mach Intell*, vol. 44, no. 12, pp. 8704–8716, Dec. 2022, doi: 10.1109/TPAMI.2019.2918284.

[24] Y. Lin, Y. Zhang, and X. Yang, "A Low Memory Requirement MobileNets Accelerator Based on FPGA for Auxiliary Medical Tasks," *Bioengineering*, vol. 10, no. 1, p. 28, Dec. 2022, doi: 10.3390/bioengineering10010028.

[25] T. N. V. S. Praveen, D. Sivathmika, G. Jahnavi, and J. Bolledu, "An In-depth Exploration of ResNet-50 for Complex Emotion Recognition to Unraveling Emotional States," in *2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, IEEE, May 2023, pp. 1–5. doi: 10.1109/InCACCT57535.2023.10141774.

[26] A. A. Mahmood, S. Sadeq, Y. I. Aljanabi, and A. H. Sabry, "Developing a convolutional neural network for classifying tumor images using Inception v3," *Eastern-European Journal of Enterprise Technologies*, vol. 3, no. 9 (123), pp. 86–93, Jun. 2023, doi: 10.15587/1729-4061.2023.281227.

[27] L. Luo *et al.*, "A Reconfigurable Spatial Architecture for Energy-Efficient Inception Neural Networks," *IEEE J Emerg Sel Top Circuits Syst*, vol. 13, no. 1, pp. 7–20, Mar. 2023, doi: 10.1109/JETCAS.2023.3243619.

[28] A. R. Kusumastuti, Y. Kristian, and E. Setyati, "Klasifikasi Ketertarikan Belajar Anak PAUD Melalui Video Ekspresi Wajah Dan Gestur Menggunakan Convolutional Neural Network," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 10, no. 2, pp. 182–188, Aug. 2021, doi: 10.32736/sisfokom.v10i2.1146.

[29] H. Zhang, D. Chen, and C. Wang, "Confidence-Aware Multi-Teacher Knowledge Distillation," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, May 2022, pp. 4498–4502. doi: 10.1109/ICASSP43922.2022.9747534.

[30] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge Distillation: A Survey," *Int J Comput Vis*, vol. 129, no. 6, pp. 1789–1819, Jun. 2021, doi: 10.1007/s11263-021-01453-z.

[31] J. H. Cho and B. Hariharan, "On the Efficacy of Knowledge Distillation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Oct. 2019, pp. 4793–4801. doi: 10.1109/ICCV.2019.00489.

[32] T. Andrei-Alexandru and D. E. Henrietta, "Low Cost Defect Detection Using a Deep Convolutional Neural Network," in *2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, IEEE, May 2020, pp. 1–5. doi: 10.1109/AQTR49680.2020.9130004.

[33] F. Yuan *et al.*, "Reinforced Multi-Teacher Selection for Knowledge Distillation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 16, pp. 14284–14291, May 2021, doi: 10.1609/aaai.v35i16.17680.

[34] M. Cesarini *et al.*, "Usage of the Kullback–Leibler divergence on posterior Dirichlet distributions to create a training dataset for a learning algorithm to classify driving behaviour events," *Journal of Computational Mathematics and Data Science*, vol. 8, p. 100081, Aug. 2023, doi: 10.1016/j.jcmds.2023.100081.

[35] E. U. Haq, H. Jianjun, X. Huarong, K. Li, and L. Weng, "A Hybrid Approach Based on Deep CNN and Machine Learning Classifiers for the Tumor Segmentation and Classification in Brain MRI," *Comput Math Methods Med*, vol. 2022, pp. 1–18, Aug. 2022, doi: 10.1155/2022/6446680.

[36] H. Jang, J. Jung, J. Song, J. Yu, Y. Kim, and J. Lee, "Pipe-BD: Pipelined Parallel Blockwise Distillation," in *2023 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, IEEE, Apr. 2023, pp. 1–6. doi: 10.23919/DATE56975.2023.10137044.