Available online at http://scik.org
Commun. Math. Biol. Neurosci. 2025, 2025:58
https://doi.org/10.28919/cmbn/9110
ISSN: 2052-2541

TRANSFER LEARNING APPROACH WITH EFFICIENTNET TO ENHANCE FOOD RECOGNITION SYSTEMS FOR HEALTH MONITORING

ISLAM NUR ALAM[1,*], FRANZ ADETA JUNIOR[1], RUDY SUSANTO[2]

[1]Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

[2]Computer Engineering Department, Faculty of Engineering, Bina Nusantara University, Jakarta 11480, Indonesia

Abstract: Food image classification is a challenging problem with significant potential benefits for real-world applications such as nutritional and energy estimation. Most prior research has proposed various Convolutional Neural Network (CNN) architectures to tackle this issue. However, given the large size and diverse nature of food image datasets, there remains considerable room for improvement, particularly in terms of accuracy and training speed. Typically, neural networks trained on small image classification datasets benefit from using pre-trained weights from large-scale image classification datasets like ImageNet. In this study, we explore the balance between using pre-trained networks as feature extractors and fine-tuning networks for food image classification. By leveraging transfer learning with EfficientNetV2B0, we achieve higher accuracy in food image classification. On the largest publicly available food image dataset, FOOD-101, our proposed method improves the previous best accuracy from 77.40% to 81.62%, while maintaining a prediction speed of 23 ms on a GPU.

Keywords: food recognition; transfer learning; EfficientNet; convolutional neural network.

2020 AMS Subject Classification: 68T01, 68T10, 97P80.

---

[*]Corresponding author
E-mail address: islam.alam@binus.ac.id
Received January 09, 2025

1. INTRODUCTION

This Identifying food ingredients is an easy activity for humans. It is human nature to recognize various characteristics and patterns, both simple and complex [1]. Recognizing foods marks the first step to maintaining a nutritious diet, preparing recipes and making dietary suggestions. When a person recognizes a particular food, they can then look up information about its nutritional content. This helps in consistently monitoring the nutrients consumed and required by the human body. Additionally, it allows the person to explore recipes associated with a particular food [2].

Humans can easily identify food ingredients, but food recognition systems can do more than just that. They can also inform people about the nutrients in food. This could lead to apps that track our daily nutrition, suggest foods or drinks to eat, and even suggest good recipes. Since most fitness apps don't have this feature, adding it would be a great way to compete with other fitness apps. In simpler terms, food recognition systems can be used to create apps that can help people track their diet and make healthier choices. This would be a valuable addition to most fitness apps, as it would give them a competitive advantage over other apps [3].

Convolutional neural networks (CNN) algorithm have significantly bolstered performance enhancements in classifying images based on objects. Ever since AlexNet [4] was presented in 2012, Deep Convolutional Neural Networks (DCNNs) have become the most popular approach for solving image classification tasks. This is due to their ability to achieve high accuracy levels by utilizing extensive labeled datasets and their millions of parameters. After achieving success, numerous other studies, including VGG16 [5], InceptionNet [6], and ResNet[7], have investigated various network architectures to enhance performance in terms of accuracy, memory efficiency, and speed of inference. Deep Convolutional Neural Networks (DCNNs) consist of multiple convolutional, linear, and non-linear layers, and are frequently used in computer vision tasks due to their convolution operation's translation equivariant property. CNNs are known for their ability to recognize patterns in images by applying a series of convolutional operations that detect features at different scales and orientations [8].

Training convolutional neural networks presents challenges for both small and large datasets. While large datasets require extensive manual annotation, small datasets struggle to provide enough training data [9]. The significant computational resources needed for training further complicate the process. Transfer learning emerges as a valuable tool for overcoming these limitations, particularly for working with smaller datasets [10].

To extract high-level semantic features from images, address data limitations, and achieve higher accuracy in food image recognition tasks, this paper proposes a method based on transfer learning [11]. This approach leverages the capability of convolutional neural networks to automatically extract image features. Pre-trained convolutional neural network models EfficientNet [9] from the ImageNet dataset are fine-tuned using the food-101 dataset. Experimental data is used to compare the performance of individual models on the food image dataset. The proposed method has demonstrated promising results, showcasing the feasibility of applying model transfer and combination strategies to food image recognition.

2. RELATED WORK

Le Bu and Xiuliang Zhang's paper proposed a food image recognition method using transfer learning and ensemble learning [12]. They extracted generic image features from pre-trained convolutional neural network models (VGG19, ResNet50, MobileNetV2, AlexNet) on the ImageNet dataset. Chang Liu et al. [13] developed a Convolutional Neural Network (CNN) based algorithm for food image recognition to tackle challenges in this field. Introduces DeepFood, a system that uses deep learning for food image recognition, aiming to improve computer-aided diet assessment accuracy, and shows promising results on real datasets. They applied the algorithm to two real food image datasets, UEC-256 and Food-101, and achieved outstanding results.

Metwalli et al.[14] introduces DenseFood, a food classification model using a densely connected convolutional neural network architecture. It leverages a combination of softmax and center loss to minimize intra-category variation and maximize inter-category variation. Then, fine tune pre-trained DenseNet121 and ResNet50 models to extract features from the dataset. Hasan et al. [15] propose a novel approach utilizing a genetic algorithm to automatically select blocks of layers, rather than individual layers. This method leverages the recently introduced metric, [16]OTDD, to assess the significance of these blocks in feature extraction. By applying OTDD, they quantify the contribution of each block towards identifying relevant features. Their research evaluates the effectiveness of this method using three diverse datasets: Food-101, CIFAR-100, and MangoLeafBD. For their CNN architecture, they employ pre-trained EfficientNet [17] models on the ImageNet dataset.

From these previous works, we can get some key information. Firstly, all those studies rely on neural network training to achieve the best results. Secondly, Transfer Learning can significantly improve classifier performance on small and large datasets. However, there is no comprehensive study on applying Transfer Learning using EfficientNetV2 [9] on the largest food dataset, FOOD-101. The purpose of this study is to compare the results of Transfer Learning with different networks and different training strategies.

3. MATERIALS AND METHODS

In this section, we will first explain the transfer learning techniques, followed by the networks used in the experiments, including the EfficientNet architecture, and the training procedures employed, which involve Transfer Learning training. Next, we will provide a detailed explanation of the experimental techniques used for developing the food recognition model.

3.1. Transfer Learning

Training convolutional neural networks is data-hungry, requiring extensive labeled datasets. This data acquisition process is time-consuming and tedious, relying on manual annotation. Moreover, training these networks demands significant computational power, often beyond the reach of many individuals. Recent advancements in transfer learning offer a potential solution [18]. This technology allows models to learn from a source domain and apply that knowledge to a new, target domain. By transferring existing labeled data to unlabeled data, it enables the creation of deep learning models tailored to specific tasks, reducing the need for extensive manual annotation and computational resources [19].

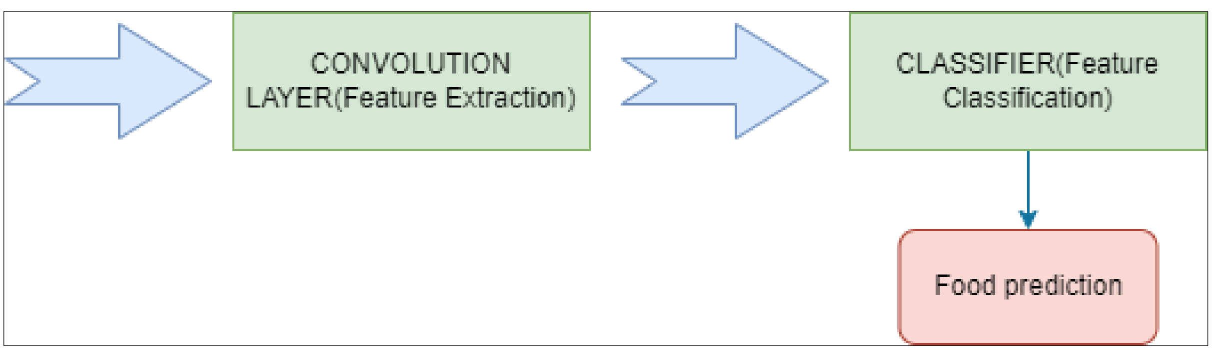

FIGURE 1. Transfer learning based EfficientNet model architecture for Food Recognition.

Figure 1 illustrates the overall architecture of a convolutional neural network (CNN), where the convolutional base is derived from a pre-trained EfficientNet, excluding its original classifier, and the classifier has been replaced with newly added layers specifically designed for food recognition. Repurposing a pre-trained deep convolutional neural network (DCNN) involves two main steps: replacing the original classifier with a new one and fine-tuning the model. The new classifier typically consists of one or more fully connected dense layers. In transfer learning, it is crucial to select an appropriate pre-trained model and to establish a size-similarity matrix for fine-tuning [20]. There are three common strategies for fine-tuning: training the entire model, training some layers while leaving others frozen, and training only the classifier while keeping the convolutional base frozen. For tasks that are similar to the original task, training only the classifier and/or a few layers is sufficient. Conversely, for tasks that are dissimilar, training the entire model is necessary. Fine-tuning is therefore conducted on the added classifier and either a selected portion or the entirety of the convolutional base. Selecting the appropriate portion for fine-tuning and the best training methods is a complex process that is handled in this study through a pipeline strategy to achieve an optimal food recognition model [21].

3.2. EfficientNet

EfficientNet is a convolutional neural network that is created to scale a network's depth, width, and resolution in proportion to a compound coefficient. This means that by increasing computational resources with a coefficient ϕ, we can increase the network's depth by α, width by β, and image size by γ, with α, β, and γ being constant coefficients determined through a small grid search. The initial study [17] introduced 8 variations of scaled networks. In this research, our focus is solely on the network with an image input size of 224×224 to ensure a fair comparison with other approaches.

EfficientNetV2 enhances the performance of EfficientNetV1 in terms of computational efficiency, network size, and predictive accuracy. To accelerate training and reduce the network's size, it limits the maximum image scaling size to 480×480 and eliminates unnecessary search options like pooling skip operations that were found in the original EfficientNetV1. Paper [9] also introduces the concept of progressive learning, where the networks gradually enhance regularization as they learn over an extended period.

Overall, based on the testing results shown in Table 1 and Figure 5, it can be concluded that the proposed LSTM model using input type 4 was the best among all other input types. The LSTM model using input type 4 was the best at predicting TDS and water temperature, while the LSTM model using input type 2 was the best at predicting water pH. A quick glance at Figure 5 shows that the LSTM model using input type 1 failed to predict all water quality parameters, including pH, TDS, and temperature, by only resulting in a horizontal straight line. In our analysis, the prediction may not be good enough due to the data tending to be pattern less without containing any pattern such as seasonality, and due to the characteristic of the dataset depicted in PCC values not containing any strong correlation but only containing negligible, weak, and moderate correlation. In our conjecture, the data may not be large enough to make the predictive model learn. We also assumed the dataset used in our research may contain noise because the data was obtained from sensors, which makes predicting water quality more challenging using this dataset.

3.3. The Recognition Method for Food Images based on Transfer Learning

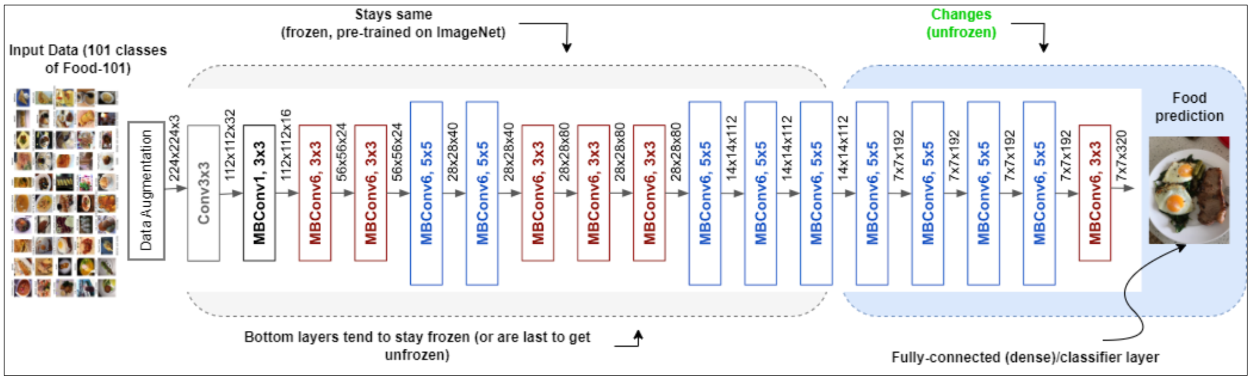

FIGURE 2. The Whole process of food image recognition based on Transfer Learning with EfficientNet Model.

The research presents a food image recognition method using transfer learning, involving three key stages: data preprocessing, network pre-training, and fine-tuning. A visual overview of the method is depicted in Figure 2. In the data preprocessing phase, food images are enhanced primarily through data augmentation techniques. For network pre-training and fine-tuning, pre-trained (EfficientNet) convolutional neural networks from ImageNet datasets are chosen. These models are adapted to the food image dataset, refining them based on extracted common image features. While some feature layers remain fixed, the rest, along with all final layers, are fully retrained. Following additional training and predictive analysis, the final recognition output is obtained. The discussion of the research results obtained can be presented in the form of theoretical description, both qualitatively and quantitatively. In practice, this section can be used to compare the results of the research obtained in the current research on the results of the research reported by previous researchers referred to in this study. Scientifically, the results of research obtained in the study may be new findings or improvements, affirmations, or rejection of a scientific phenomenon from previous researchers.

3.4. Data Preprocessing

Data preprocessing serves as the cornerstone for analyzing digital images in machine vision. Its primary aim is to eliminate different types of noise from images, enhance their quality, facilitate the retrieval of valuable information, and simplify subsequent tasks like feature extraction, image segmentation, and image recognition. This ultimately boosts the reliability of digital image analysis and processing. In this approach, image preprocessing consists of two steps: data augmentation and data standardization [22].

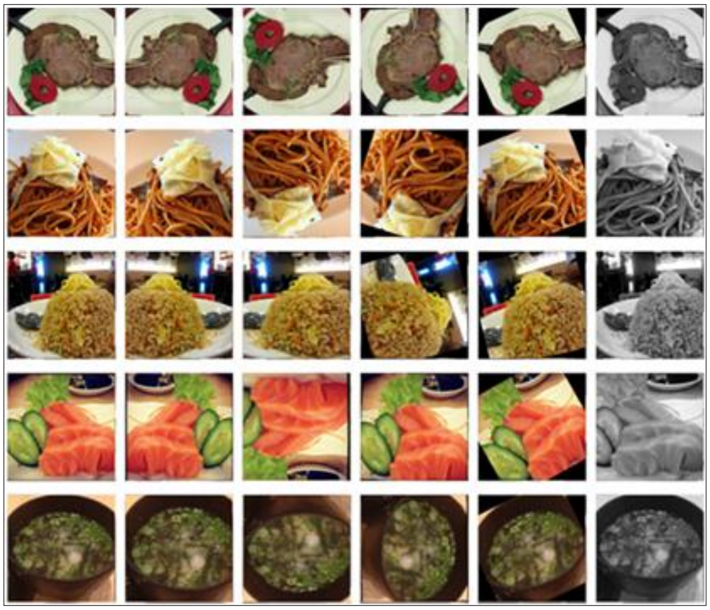

FIGURE 3. Example of data augmentation.

Data augmentation is an effective technique for increasing the sample size of a dataset. With a larger number of training samples, convolutional neural networks can learn a wider range of image features, which enhances the model's generalization capabilities. The data augmentation methods employed in this study include: 1) random cropping, 2) horizontal flipping, 3) vertical flipping, 4) 180˚ rotation, 5) affine transformations, and 6) conversion to grayscale. An example of data augmentation applied in this method can be seen in Figure 3. Data standardization refers to the process of normalizing each channel of an image (224 × 224). This technique aids in improving the convergence of the model during training.

3.5. Networks Fine-tuning

In image classification tasks, the number of labeled classes often differs between a large dataset and a smaller one. Consequently, the first step in fine-tuning a pre-trained deep neural network (EfficientNet) involves modifying the last output layer so that it matches the number of classes in the target dataset. This step is akin to using the convolutional layers as feature extractors while only updating the fully connected layer. This initial training of the classifier is crucial for ensuring stable training during the subsequent step, where the entire network is fine-tuned. The training process is illustrated in Figure 2. The top portion of the figure shows a network pre-trained on the ImageNet dataset [23]. The weights from the convolutional layers are reused in the middle network, where only the fully connected layer is trained. The lower network displays the fine-tuning of the entire model, with both the convolutional and fully connected layers being trained end-to-end. Typically, a lower learning rate is employed during the fine-tuning process.

3.6. Hyper-parameters Setup

Training deep networks is often time-consuming and resource-intensive, with hyperparameter tuning being a significant contributing factor. To streamline this process, we avoided extensive hyperparameter searches by assigning reasonable values to some of them. Specifically, we configured the batch size to 64 and chose the Adam optimizer as our stochastic optimizer due to its effectiveness in training deep neural networks [24]. For the feature extraction training and the last fully connected layer, we set the learning rates to $1 \times 10{-}3$, with $\beta1$ and $\beta2$ values of 0.9 and 0.999, respectively. When fine-tuning the networks, we lowered the learning rates to $1 \times 10{-}5$ to avoid making significant changes to the weights. Throughout the training, we allowed the networks to learn for 10 epochs. Finally, the weight decay was set to $5 \times 10{-}4$.

3.7. Loss Function

Let y be a one-hot ground-truth label vector, y' represent the predicted probabilities for each class from the networks and let cc denote each possible class. We employed the widely used multi-class cross-entropy as our loss function. Mathematically, it is defined as:

$$L(\gamma, y') = -\sum_{c} y_c \log y_c{}' \quad (1)$$

4. Experiment Results and Analysis

In this section, we present the results of our experiments on food recognition utilizing transfer learning with EfficientNet architectures. We provide a detailed analysis of the performance metrics, including accuracy, precision, recall, and F1 score, to evaluate the effectiveness of our models. Additionally, we discuss the training duration for each experiment, highlighting the computational resources required and any challenges encountered during the process. Furthermore, we offer observations regarding the model's behavior, including its strengths and weaknesses in recognizing different types of food, as well as insights into how the transfer learning approach impacted the overall performance. This comprehensive evaluation aims to provide a better understanding of the capabilities of EfficientNet in the context of food recognition tasks.

4.1. Experiment Setup

We outline the setup for our experiments, which focused on two main architectures: EfficientNetB0 and EfficientNetV2B0. For both experiments, we adopted a two-step training process. Initially, the models were trained using a feature extraction approach, allowing us to leverage pre-trained weights while adapting the models to our specific dataset. Following this, we performed fine-tuning on the entire model to improve its performance and learning capability further.

To prepare the data, both the training and validation datasets underwent comprehensive preprocessing, including normalization and resizing of images. We also applied various data augmentation techniques, such as rotation, flipping, and color adjustment, to enhance the diversity of the training set and mitigate overfitting. This thorough experiment setup aimed to maximize the effectiveness of the EfficientNet architectures in the food recognition tasks.

4.2. Feature Extraction Results

The EfficientNetB0 model was trained for a total of 5 epochs with a batch size of 32. The results of the training process are summarized in Table 1. The model achieved a peak validation accuracy of 73.36% at the 5th epoch, indicating a reasonable performance in food recognition tasks using transfer learning.

TABLE 1. EfficientNetB0 Feature Extraction Results

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	1.7167	58.16%	1.1158	70.60%
2	1.2005	68.95%	1.0270	71.88%
3	1.0553	72.39%	1.0047	72.62%
4	0.9587	74.78%	0.9684	72.83%
5	0.8893	76.45%	0.9713	73.36%

In a similar manner, the EfficientNetV2B0 model was also trained for 5 epochs, and the results are summarized in Table 2.

TABLE 2. EfficientNetV2B0 Feature Extraction Results

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	1.7470	57.61%	1.1461	69.65%
2	1.2120	68.57%	1.0396	72.64%
3	1.0638	72.09%	0.9921	73.46%
4	0.9674	74.63%	0.9720	73.89%
5	0.8976	76.24%	0.9605	74.42%

The EfficientNetV2B0 model demonstrated slightly improved performance compared to EfficientNetB0, achieving a peak validation accuracy of 74.42%. This suggests that the EfficientNetV2B0 architecture may provide enhancements suitable for food recognition tasks, supporting the effectiveness of transfer learning in this domain.

4.2. Fine-Tuning Results

After the initial feature extraction, both EfficientNetB0 and EfficientNetV2B0 models underwent a fine-tuning process to further enhance their performance on the food recognition tasks. This fine-tuning involved strategies such as learning rate reduction and early stopping to mitigate overfitting. The fine-tuning process for EfficientNetB0 was carried out for up to 100 epochs, and the results are summarized in Table 3.

TABLE 3. EfficientNetB0 Fine-Tuning Results

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	0.7953	78.28%	0.7972	77.07%
2	0.4783	86.57%	0.8014	78.50%
3	0.2594	92.59%	0.8485	78.68%
4	0.0616	98.56%	0.9499	80.32%

The fine-tuned EfficientNetB0 model achieved a peak validation accuracy of 80.32% at the 4th epoch, indicating a significant improvement in performance compared to the feature extraction phase. EfficientNetV2B0 Fine-Tuning Similarly, the fine-tuning process for EfficientNetV2B0 was also conducted for up to 100 epochs. The results are detailed in Table 4.

TABLE 4. EfficientNetV2B0 Fine-Tuning Results

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	0.7990	78.14%	0.7310	79.85%
2	0.4608	87.02%	0.7419	80.72%
3	0.2328	93.32%	0.8700	79.24%
4	0.0450	99.01%	0.9090	81.62%

The EfficientNetV2B0 model reached its peak validation accuracy of 81.62% at the 4th epoch, demonstrating a greater enhancement in performance compared to EfficientNetB0. Overall, the fine-tuning phase significantly enhanced the models' capabilities in food recognition tasks, with EfficientNetV2B0 outperforming EfficientNetB0 in terms of validation accuracy. This demonstrates the effectiveness of fine-tuning in optimizing transfer learning models for specific applications. The fine-tuned EfficientNetB0 model achieved a peak validation accuracy of 80.32% at the 4th epoch, reflecting a substantial improvement in performance compared to the feature extraction phase.

5. DISCUSSION

The results indicate that both EfficientNetB0 and EfficientNetV2B0 architectures are effective for food recognition tasks when fine-tuned. Fine-tuning significantly improved the validation accuracy, with EfficientNetV2B0 achieving the highest accuracy of 81.62%. Our results are consistent with previous research that demonstrates the effectiveness of transfer learning and fine-tuning in improving model performance for image classification tasks. The high accuracy achieved by the models suggests that they can be effectively used in real-world food recognition applications, such as automated dietary monitoring and food tracking systems.

6. CONCLUSIONS

In this study, we explored the effectiveness of transfer learning using EfficientNet architectures for food recognition tasks, specifically employing EfficientNetB0 and EfficientNetV2B0 to classify images into 101 food categories. Both models demonstrated strong performance, with EfficientNetV2B0 slightly outperforming EfficientNetB0, achieving peak validation accuracies of 81.62% and 74.42%, respectively, after fine-tuning. The fine-tuning process significantly enhanced model performance, improving validation accuracies compared to the initial feature extraction phase. EfficientNetV2B0 showed superior results both in the initial training phase and after fine-tuning, suggesting that the newer architecture benefits from advanced features and optimizations. The improvements observed through fine-tuning, such as using a lower learning rate and early stopping, contributed to better generalization and reduced overfitting.

The results are consistent with existing literature on transfer learning in image classification tasks, supporting the utility of these models in practical applications such as automated food recognition systems. The high performance of these models highlights their potential for real-world use in dietary tracking and food recognition. Future research could focus on exploring advanced or hybrid architectures, expanding datasets, and incorporating data augmentation techniques to further enhance model accuracy and efficiency. Overall, EfficientNet architectures, particularly EfficientNetV2B0, prove to be a robust framework for food recognition, effectively addressing complex image classification challenges.

ACKNOWLEDGEMENT

This work is supported by Bina Nusantara University as a part of Bina Nusantara University's BINUS Research for Early Career Researchers entitled "Food Recognition Benchmark: Using Deep Learning to Recognize Food in Images" with contract number: 069A/VRRTT/III/2024 and contract date: March 18, 2024

AUTHOR CONTRIBUTIONS

Islam Nur Alam: Conceptualization, Methodology, Software, Writing- Original Draft Preparation; Franz Adeta Junior: Data Curation, Validation, Supervision; Rudy Susanto: Software, Validation, Writing-Reviewing and Editing.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

REFERENCES

[1] Z. Zahisham, C.P. Lee, K.M. Lim, Food Recognition with ResNet-50, in: 2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), IEEE, Kota Kinabalu, Malaysia, 2020: pp. 1–5. https://doi.org/10.1109/IICAIET49801.2020.9257825.

[2] C. Liu, Y. Cao, Y. Luo, et al. DeepFood: Deep Learning-Based Food Image Recognition for Computer-Aided Dietary Assessment, arXiv:1606.05675 [cs.CV] (2016). https://doi.org/10.48550/ARXIV.1606.05675.

[3] C.S. Won, Multi-Scale CNN for Fine-Grained Image Recognition, IEEE Access 8 (2020), 116663–116674. https://doi.org/10.1109/ACCESS.2020.3005150.

[4] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, Commun. ACM 60 (2017), 84–90. https://doi.org/10.1145/3065386.

[5] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv:1409.1556 [cs.CV] (2015). https://doi.org/10.48550/ARXIV.1409.1556.

[6] C. Szegedy, W. Liu, Y. Jia, et al. Going Deeper with Convolutions, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Boston, 2015: pp. 1–9. https://doi.org/10.1109/CVPR.2015.7298594.

[7] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Las Vegas, NV, USA, 2016: pp. 770–778. https://doi.org/10.1109/CVPR.2016.90.

[8] F.N. Iandola, A.E. Shaw, R. Krishna, K.W. Keutzer, SqueezeBERT: What Can Computer Vision Teach NLP about Efficient Neural Networks?, arXiv:2006.11316 [cs.CL] (2020). https://doi.org/10.48550/ARXIV.2006.11316.

[9] M. Tan, Q.V. Le, EfficientNetV2: Smaller Models and Faster Training, arXiv:2104.00298 [cs.CV] (2021). https://doi.org/10.48550/ARXIV.2104.00298.

[10] I.N. Alam, I.H. Kartowisastro, P. Wicaksono, Transfer Learning Technique with EfficientNet for Facial Expression Recognition System, Rev. Intell. Artif. 36 (2022), 543–552. https://doi.org/10.18280/ria.360405.

[11] F. Zhuang, Z. Qi, K. Duan, et al. A Comprehensive Survey on Transfer Learning, Proc. IEEE 109 (2021), 43–76. https://doi.org/10.1109/JPROC.2020.3004555.

[12] L. Bu, C. Hu, X. Zhang, Recognition of Food Images Based on Transfer Learning and Ensemble Learning, PLOS ONE 19 (2024), e0296789. https://doi.org/10.1371/journal.pone.0296789.

[13] S. Minaee, M. Minaei, A. Abdolrashidi, Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network, Sensors 21 (2021), 3046. https://doi.org/10.3390/s21093046.

[14] A.-S. Metwalli, W. Shen, C.Q. Wu, Food Image Recognition Based on Densely Connected Convolutional Neural Networks, in: 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), IEEE, Fukuoka, Japan, 2020: pp. 027–032. https://doi.org/10.1109/ICAIIC48513.2020.9065281.

[15] M.M. Hasana, M. Ibrahim, M.S. Ali, Speeding Up EfficientNet: Selecting Update Blocks of Convolutional Neural Networks Using Genetic Algorithm in Transfer Learning, arXiv:2303.00261 [cs.LG] (2023). https://doi.org/10.48550/ARXIV.2303.00261.

[16] D. Alvarez-Melis, N. Fusi, Geometric Dataset Distances via Optimal Transport, in: H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc., 2020: pp. 21428–21439.

[17] M. Tan, Q.V. Le, EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, arXiv:1905.11946 [cs.LG], 2019. https://doi.org/10.48550/arXiv.1905.11946.

[18] F. Zhuang, Z. Qi, K. Duan, et al. A Comprehensive Survey on Transfer Learning, Proc. IEEE 109 (2021), 43–76. https://doi.org/10.1109/JPROC.2020.3004555.

[19] J. Hu, Y. Zou, B. Sun, et al. Raman Spectrum Classification Based on Transfer Learning by a Convolutional Neural Network: Application to Pesticide Detection, Spectrochim. Acta Part A: Mol. Biomol. Spectrosc. 265 (2022), 120366. https://doi.org/10.1016/j.saa.2021.120366.

[20] W. Rawat, Z. Wang, Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review, Neural Comput. 29 (2017), 2352–2449. https://doi.org/10.1162/neco_a_00990.

[21] Y. Bengio, Learning Deep Architectures for AI, Found. Trends Mach. Learn. 2 (2009), 1–127. https://doi.org/10.1561/2200000006.

[22] Z. Yang, R.O. Sinnott, J. Bailey, Q. Ke, A Survey of Automated Data Augmentation Algorithms for Deep Learning-Based Image Classification Tasks, Knowl. Inf. Syst. 65 (2023), 2805–2861. https://doi.org/10.1007/s10115-023-01853-2.

[23] M.A.H. Akhand, S. Roy, N. Siddique, et al. Facial Emotion Recognition Using Transfer Learning in the Deep CNN, Electronics 10 (2021), 1036. https://doi.org/10.3390/electronics10091036.

[24] D.P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, arXiv:1412.6980 [cs.LG] (2014). https://doi.org/10.48550/ARXIV.1412.6980.