Integration of LIME Explainable AI to Enhance Interpretability of Deep Learning Models in Box Palette Classification

Evan Raditya[1,*], Rarasmaya Indraswari[2]

Department of Information Systems, Institut Teknologi Sepuluh Nopember, Indonesia

[1]6026231037@student.its.ac.id; [2] raras@its.ac.id

*corresponding author

ARTICLE INFO

Article History
Received: 29 July 2024
Revised: 13 August 2024
Published: 30 August 2024

Keywords
Box Palette
Convolutional Neural Network
Detection System
Explainable Ai
Image Classification

ABSTRACT

In the food production system, manual box palette arrangement often encounters errors, such as incorrect stacking patterns, mismatched box quantities, and improper mixing of product variants. This issue occurs in a soy sauce production company with limited infrastructure, leading to the mixing of two product variants with the same box size in a single production line. This results in disruptions to the production flow and significant potential losses. This study proposes a solution using deep learning to detect box palette arrangement patterns. The Convolutional Neural Network (CNN) method is chosen because it has proven effective in image classification. Additionally, this study implements Explainable AI (XAI) to provide explanations related to classification results, increasing user confidence in the system. The Local Interpretable Model-agnostic Explanations (LIME) technique will be used to provide interpretations. This research produces the development of a deep learning model to classify box palette arrangements. Furthermore, the implementation of LIME in this study successfully provides interpretations of the model's predictions. The result is evident in the result where it shows that MobileNetV2 give an F1-Score of 100%, and LIME fidelity score of 0.2 and stability score of 0.2.

INTRODUCTION

Currently, the arrangement of pallet boxes can be done manually or automatically. In a food product production system that relies on manual labor, errors in box arrangement are common. Frequently encountered issues include incorrect arrangement patterns, incorrect quantities of boxes, and the presence of incorrect variants in the box arrangement. Such errors can disrupt the production system's workflow. One possible solution is to create a detection system using image classification. Image classification is employed in the system because it can classify the differences in each arrangement pattern. One image classification method that can be used for this problem is deep learning. Deep learning in image classification is widely used and developed. Additionally, due to its extensive development, deep learning offers accurate performance in image classification. For instance, the use of deep learning methods with Convolutional Neural Network (CNN) architecture in image classification [1]. CNN can be used for image classification tasks. The performance of CNNs can match or exceed human performance in some cases [2]. Previous CNN models have demonstrated outstanding performance in terms of classification, segmentation, and image extraction[3][4]. This is supported by various studies implementing CNN for different classification tasks. An example is the implementation of CNN for mineral classification tasks[5]. This research discusses various applications and studies related to image classification using CNN for mineral exploration. CNN provides a powerful new way to

analyze images related to mineral exploration. Another example is the use of CNN for white blood cell classification tasks[6]. In this study, two datasets were used: BCCD and Raabin-WBC. The research found that classification using the CNN DenseNet121 model produced good performance, with an accuracy of 95.87% on the BCCD dataset and 98.59% on the Raabin-WBC dataset.

However, in its development, deep learning operates as a black box. Therefore, a new advancement has emerged: the use of explainable AI (XAI) in deep learning algorithms for image classification. XAI is a development in machine learning that aims to explain the reasoning and provide context for the output of a deep learning model. An example is the implementation of XAI in the form of Local Interpretable Model-agnostic Explanations (LIME) in the classification of white blood cells using deep learning [7]. In this study, a comparison of five deep learning models was conducted on two white blood cell datasets. The results of this study indicate that XAI (LIME) can explain the reasoning behind the model's classification predictions because a model with high accuracy does not necessarily provide correct prediction results[8]. Therefore, with XAI results, it can be determined which parts of an image are referenced by a model to make its predictions[9]. Image classification using deep learning has been applied for classifying box pattern. For example, [10] propose a method to detect logistic box arrangements based on Mask R-CNN supported by Cycle Generative Adversarial Network (GAN). From the method proposed, the performance of box detection improved. This improvement was validated using 200 cases each of orderly and disorderly box arrangements.

XAI has been implemented in various classifications task, such as in finance or medical field. Previous research [6] implemented XAI by using LIME method into white blood classification. LIME implementation can provide interpretable and transparent results for users, thereby increasing their confidence and trust in automatic diagnoses. LIME results also can refine pre-trained models on smaller domain-specific datasets, reducing computational resources and time needed to train models from scratch. Another research, [11] implement another XAI method, called SHAP (Shapley Additive Explanations) in skin cancer classification task. This research shows that SHAP can generate local explanations by localizing the model using a smaller model and perturbing the input data to observe how the output changes. By using SHAP results, the researchers successfully performed feature reduction to improve classification performance efficiency. On another field, [12] used XAI in remote sensing image classification. In this research, a new XAI method called "What I Know" (WIK) is proposed to verify the reliability of a deep learning model. WIK evaluates similarity based on features extracted by the model, rather than input data, to determine whether the training data adequately represents the target data. This research shows that WIK, as an example of an XAI method, has applications in several key areas where reliable explanations are crucial to avoid decision-making errors. It helps users understand the AI model's decision-making process and fosters trust in the AI system. Other than that, [13] also implementing XAI on EMG data for hand gesture classification. XAI successfully helped in feature reduction to improve classification performance efficiency.

Currently, there has been extensive research on the implementation of deep learning for image pattern classification. However, studies specifically focused on classifying box arrangement patterns in a production system, especially food production, are quite rare. Therefore, this research will contribute to the development of a detection system to classify box palette arrangement patterns using CNN. Additionally, this research will implement explainable AI (XAI), specifically using LIME to increase trust in the system, particularly in systems that frequently interact with users. With LIME, classification results can be explained, enabling even lay users to understand the reasoning and context behind the

classification outcomes. This research aims to produce a reliable system that can classify box arrangement patterns to address errors in box stacking. Additionally, this research is expected to provide new insights into the implementation of LIME in a classification system using CNN.

METHODS

Dataset

This study use dataset that consist of photos of box arrangements. The photos were taken manually using a camera from a top-down position over the box arrangements. The photo data collected consisted of box arrangements for two product variants: 60 ml boxes and 63 ml boxes. For each variant, two patterns were used within a single box palette. Additionally, for each pattern, there were two classes: data for correct arrangements and data for incorrect arrangements. Overall, there were eight data classes. For each class of correct arrangements, 50 photos were taken, while for each class of incorrect arrangements, 100 photos were taken. In total, 600 photos were collected to form the dataset. The data was captured using a smartphone from above the box palette arrangement.

Table 1. Data Example

Class	Data Example
60ml_P1_Correct	
60ml_P1_Wrong	
60ml_P2_Correct	
60ml_P2_Wrong	
63ml_P1_Correct	

63ml_P1_Wrong	
63ml_P2_Correct	
63ml_P2_Wrong	

Based on Table 1, it can be seen that for each class, the data is very similar. The box has two sets of patterns for even and odd layer each, so the pattern must follow the predetermined pattern, thus it can be considered as the correct class. As for the wrong class, the box was arranged incorrectly and not following the predetermined pattern. The most visible example can be seen on the 60ml_P1_Wrong class where there is one box that placed vertically in the arrangement, thus considered as the wrong class.

Data Preprocessing

The collected data needs to go through a preprocessing stage. In this step, the data is grouped for each class. Each class is separated into different folders. There are eight classes in total, so the data is divided into eight folders. Then, data augmentation is performed to address the limited size of the dataset. Augmentation aims to modify the training data using one or more predefined operations. The augmentations applied include rotation, flipping, zooming, and shifting of the existing dataset. The augmented data will be the input for the network in each training iteration of the model. In each iteration, a different set of data with the same amount will be used, so the model can develop good generalization capabilities. The better the generalization capability of a model, the better it can handle test data that it has not been trained on.

Model Development

At this stage, deep learning models are created using the base model architectures of ResNet50V2, MobileNetV2, VGG16, and InceptionV3. These models are examples of transfer learning models. Transfer learning is a machine learning method where models are initially trained on a different dataset, allowing the training process on a new dataset to be shortened. Models that have been pre-trained are referred to as pre-trained models. This process is called transfer learning because it leverages knowledge gained from training on one dataset to improve the performance of learning models on a different dataset. These model architectures are used because, based on existing research, they have excellent performance in pattern classification tasks and are some of the popular examples of transfer learning models. In the transfer learning process, the top layer of the pre-trained model is modified by adding convolutional or pooling layers. This first layer is referred to as the head

model. Then, the following layers are frozen so that the existing weights and the feed-forward process do not change. These frozen layers are called the base model. After this modification process, training is then conducted using the second dataset, and parameter tuning is performed.

Model Test and Evaluation

At this stage, several experiments are conducted related to model training. First, experiments are conducted using the base model architectures ResNet50V2, MobileNetV2, VGG16, and InceptionV3 for pattern classification of box palette arrangements. Optimization methods that determine the minimum value of the function in the classification model are also tested. The experiments will be conducted using 50 epochs. Epochs determine the number of training processes on the training data, and batch size determines the number of samples in each batch.

Then, at this stage, the model is evaluated using the metrics of accuracy, precision, recall, and F1 Score.

- Accuracy
  The ratio of correct predictions (positive and negative) to the total data is calculated as follows:
  $$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}} \quad \text{Eq. (1)}$$
- Precision
  The ratio of true positive predictions to the total predicted positives is calculated as follows:
  $$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad \text{Eq. (2)}$$
- Recall
  The ratio of true positive predictions to the total actual positives is calculated as follows:
  $$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad \text{Eq. (3)}$$
- F1-Score
  The weighted average of precision and recall is calculated as follows:
  $$\text{F1 Score} = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall+Precision}} \quad \text{Eq. (4)}$$

LIME Test and Evaluation

At this step, XAI implementation is carried out using LIME (Local Interpretable Model-agnostic Explanations). LIME is used to provide explanations for the model's classification results. In this study, LIME is used because it is one of the popular methods in XAI implementation. LIME is a model-agnostic XAI method, meaning that LIME does not depend on a specific model and can be used for various models. The implementation of LIME is carried out through several steps. The implementation of LIME on image classification can be seen on Figure 1. Next, there are several parameters that can be tested for LIME implementation. The tests to be conducted are testing LIME on various datasets and testing the number of samples. After LIME produces output, the model is then evaluated. The evaluation is conducted using the fidelity measurement metric.

- Fidelity (R^2 Score):
  Fidelity measures how well the interpretable model (local LIME model) approximates the predictions of the original black-box model. Mathematically, this metric quantifies the alignment between predictions made by the local model and those made by the more complex original model for perturbed data points used to generate explanations. Fidelity measurement considers not only attribution but also

noise and its counting score. Fidelity measurement is done by providing significant perturbations to the input or original instances. These significant perturbations can be provided by two methods: using a noisy baseline and square removal[14].

- Stability:

The stability metric is used to evaluate the consistency of explanations produced by the machine learning model. Specifically, this metric measures how much the explanations change when the input data is slightly perturbed. If small changes in the input data cause large changes in the explanations, the explanations are considered less stable (or less trustworthy)[15].

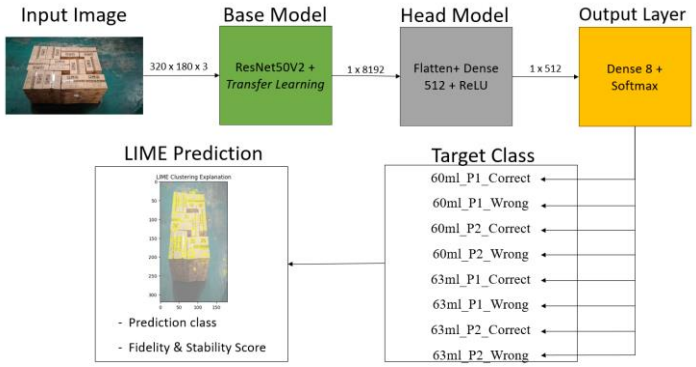


Figure 1. LIME Implementation on Image Classification

RESULTS AND DISCUSSION

In this research, the testing was conducted using Python running locally on a computer. The computer used has the following specifications: Intel Core i3-10100f, 16 GB RAM, and NVIDIA GeForce RTX 3060 TI. Various evaluation metrics were used to assess the performance of the deep learning model and LIME. The model is trained on each dataset according to base models experiment with a total of 50 epochs.

Table 2. Model Evaluation Result (box palette)

Base Model	Model Performance (%)				*Running Time (s)*
	Accuracy	*Precision*	*Recall*	*F1-Score*	
ResNet50V2	98	98	98	98	279
VGG16	100	100	100	100	315
InceptionV3	96	96	96	96	308
MobileNetV2	**100**	**100**	**100**	**100**	**268**

From the evaluation conducted, the best model for the box palette dataset is MobileNetV2 with 50 epochs. This model has the best F1-score with a score of 100%. This model is preferred over VGG16 due to its faster running time. This is because VGG16 has a much larger number of parameters and uses standard convolutional layers, which are computationally intensive. Meanwhile MobileNetV2 is designed to be lightweight and efficient, using depthwise separable convolutions that reduce the computational load, resulting in faster processing times. Other than faster running time, the use of depthwise separable convolutions make the model to be more efficient. This efficiency likely allows MobileNetV2 to effectively learn the features necessary for classifying the box palette

dataset without overfitting, which is often a concern with smaller or simpler datasets. The box palette dataset seems to benefit from the architecture of MobileNetV2, where the combination of fewer parameters and efficient computation aligns well with the dataset's complexity. The fact that MobileNetV2 outperforms more complex models like ResNet50V2, VGG16, and InceptionV3 suggests that the task does not require the additional complexity these models offer. Then, experiments and evaluations were conducted on LIME explainable AI. The experiments involved implementing LIME on the classification using the previously tested models. Then, LIME is evaluated using the fidelity and stability metrics.

Table 3 LIME Fidelity Score (box palette)

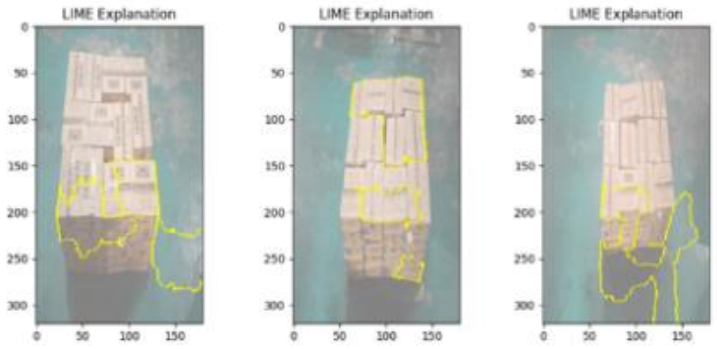
Num Samples	LIME (50 Images)	
	Avg Fidelity (higher is better)	*Running Time* (s)
500	0.2666	556
1000	0.215	870
2500	0.1752	1909

In Table 3, the evaluation results of the fidelity score for LIME can be seen. When using LIME, the highest average fidelity score is obtained when using 2500 samples. With an average score around 0.2, the fidelity metric score for LIME can be considered fairly good. This means that the local surrogate accurately represents the behavior of the original complex model in the vicinity of the instance being explained.

Table 4. LIME Stability Score (box palette)

Class	LIME Stability Metric (lower is better)
60ml_P1_Correct	0.115
63ml_P2_Correct	0.29
63ml_P1_Wrong	0.146

Based on Table 4, the evaluation results of LIME to measure the stability of explanations on the box palette dataset are shown. From the results, it can be seen that the regular LIME method produces stability scores ranging from 0.1 to 0.2 in each tested class, indicating that the stability score of LIME is fairly good. This also means that the explanations provided by the local surrogate model of LIME closely approximates the behavior of the original complex model for the perturbed instances, enhancing the interpretability and trustworthiness of the model's predictions.



(a) (b) (c)

Figure 2. LIME Explanation Results (box palette)

Then, in the images, the LIME explanations for the box palette classification for each class can be seen. Figure 2a shows the explanation for the class 60ml_P1_Correct, Figure 2b for the class 63ml_P2_Correct, and Figure 2c for the class 63ml_P1_Wrong. The results from LIME highlight features of the box pattern and slightly highlight parts of the floor. This indicates that the model considers the box pattern and the floor as important features when making predictions or classifications.

CONCLUSION

Based on the research conducted, it can be concluded that the use of CNN models with transfer learning can provide good performance for image classification. On all three datasets, the best model performance was achieved using the MobileNetV2 architecture with an F1 score of 100% for box palette dataset. Furthermore, the use of LIME can enhance the interpretability of the model by effectively explaining the model's classification results. This can be seen from LIME's fidelity score on the dataset. The score indicates that LIME's explanations closely resemble the behavior of the original model. Even though the fidelity scores are good, it can be improved. LIME's explanations also demonstrated good stability, as shown by the metric results. This indicates that LIME's explanations are consistent even when the input is slightly perturbed. LIME's visualizations successfully highlighted important features when tested on the box palette dataset, namely the box arrangement pattern. However, LIME also highlighted non-important features, such as the floor. Several factors could account for this. First, the dataset factor. The box palette dataset was self-collected within a limited timeframe, resulting in visually subtle differences between classes. This may cause the trained CNN model to have difficulty recognizing the differences and only perform well on the training and validation dataset. Therefore, although the model's evaluation performance is good, the model might underperform when given data outside the training and validation sets.

REFERENCES

[1] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 12, pp. 943–947, Nov. 2015, doi: 10.22214/ijraset.2022.47789.

[2] M. A. Islam, S. I. Rashid, N. U. I. Hossain, R. Fleming, and A. Sokolov, "An integrated convolutional neural network and sorting algorithm for image classification for efficient flood disaster management," *Decis. Anal. J.*, vol. 7, no. November 2022, p. 100225, Jun. 2023, doi: 10.1016/j.dajour.2023.100225.

[3] A. Bhandare, M. Bhide, P. Gokhale, and R. Chandavarkar, "Applications of convolutional neural networks," *Int. J. Comput. Sci. Inf. Technol.*, vol. 7, no. 5, pp. 2206–2215, 2016.

[4] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)," no. 1, pp. 2–8, Mar. 2018, [Online]. Available: http://arxiv.org/abs/1803.08375

[5] Y. Liu, X. Wang, Z. Zhang, and F. Deng, "A review of deep leaning in image classification for mineral exploration," *Miner. Eng.*, vol. 204, no. November 2022, p. 108433, 2023, doi: 10.1016/j.mineng.2023.108433.

[6] K. Bhatia, S. Dhalla, A. Mittal, S. Gupta, A. Gupta, and A. Jindal, "Integrating explainability into deep learning-based models for white blood cells classification," *Comput. Electr. Eng.*, vol. 110, no. June, p. 108913, 2023, doi: 10.1016/j.compeleceng.2023.108913.

[7] M. Ribeiro, S. Singh, and C. Guestrin, "'Why Should I Trust You?': Explaining the Predictions of Any Classifier," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, Feb. 2016, pp. 97–101. doi: 10.18653/v1/N16-3020.

[8] M. R. Zafar and N. M. Khan, "DLIME: A Deterministic Local Interpretable Model-Agnostic Explanations Approach for Computer-Aided Diagnosis Systems," Jun. 2019, [Online]. Available: http://arxiv.org/abs/1906.10263

[9] C. (Abigail) Zhang, S. Cho, and M. Vasarhelyi, "Explainable Artificial Intelligence (XAI) in auditing,"

Int. J. Account. Inf. Syst., vol. 46, no. August, p. 100572, Sep. 2022, doi: 10.1016/j.accinf.2022.100572.

[10] J. Yoon, J. Han, and T. P. Nguyen, "Logistics box recognition in robotic industrial de-palletising procedure with systematic RGB-D image processing supported by multiple deep learning methods," *Eng. Appl. Artif. Intell.*, vol. 123, no. April, p. 106311, Aug. 2023, doi: 10.1016/j.engappai.2023.106311.

[11] T. Khater, S. Ansari, S. Mahmoud, A. Hussain, and H. Tawfik, "Skin cancer classification using explainable artificial intelligence on pre-extracted image features," *Intell. Syst. with Appl.*, vol. 20, no. May, p. 200275, 2023, doi: 10.1016/j.iswa.2023.200275.

[12] S. nosuke Ishikawa *et al.*, "Example-based explainable AI and its application for remote sensing image classification," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 118, no. November 2022, p. 103215, 2023, doi: 10.1016/j.jag.2023.103215.

[13] N. Gozzi, L. Malandri, F. Mercorio, and A. Pedrocchi, "XAI for myo-controlled prosthesis: Explaining EMG data for hand gesture classification," *Knowledge-Based Syst.*, vol. 240, p. 108053, 2022, doi: 10.1016/j.knosys.2021.108053.

[14] C.-K. Yeh, C.-Y. Hsieh, A. S. Suggala, D. I. Inouye, and P. Ravikumar, "On the (In)fidelity and Sensitivity for Explanations," *Adv. Neural Inf. Process. Syst.*, vol. 32, no. NeurIPS, Jan. 2019, [Online]. Available: http://arxiv.org/abs/1901.09392

[15] D. Alvarez-Melis and T. S. Jaakkola, "Towards robust interpretability with self-explaining neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 2018-Decem, no. NeurIPS, pp. 7775–7784, 2018.