Optimized image-based grouping of e-commerce products using deep hierarchical clustering

Yuliana Melita Pranoto [a,b,1], Anik Nur Handayani [a,2,*], Heru Wahyu Herwanto [a,3], Yosi Kristian [b,4]

[a] Department of Electrical Engineering and Informatics, Universitas Negeri Malang, 5 Semarang street, Malang 65145, Indonesia
[b] Department of Informatics, Institut Sains dan Teknologi Terpadu Surabaya, 73-77 Ngagel Jaya Tengah street, Surabaya 60284, Indonesia
[1] yuliana.melita.2205349@students.um.ac.id; [2] aniknur.ft@um.ac.id; [3] heru_wh@um.ac.id; [4] yosi@stts.edu
* corresponding author

ARTICLE INFO

Article history
Received March 29, 2024
Revised October 24, 2024
Accepted July 8, 2025
Available online July 15, 2025

Selected paper from The 2024 7th International Symposium on Advanced Intelligent Informatics (SAIN'24), Nanjing, China, November 12-14, 2024, http://sain.ijain.org. Peer-reviewed by SAIN'24 Scientific Committee and Editorial Team of IJAIN journal.

Keywords
Product grouping
E-commerce
Convolutional neural network
Agglomerative clustering
Normalized mutual information

ABSTRACT

Managing large and constantly evolving product catalogs is a significant challenge for e-commerce platforms, especially when visually similar products cannot be reliably distinguished using text-based methods. This study proposes a product grouping method that combines a fine-tuned EfficientNetV2M model with an adaptive Agglomerative Clustering strategy. Unlike conventional CNN-based approaches, which have limited scalability and a fixed number of clusters, the proposed method dynamically adjusts similarity thresholds and automatically forms clusters for unseen product variations. By linking deep visual feature extraction with adaptive clustering, the method enhances flexibility in handling product diversity. Experiments on the Shopee product image dataset show that it achieves a high Normalized Mutual Information (NMI) score of 0.924, outperforming standard baselines. These results demonstrate the method's effectiveness in automating catalog organization and offer a scalable solution for inventory management and personalized recommendations in e-commerce platforms.

1. Introduction

The COVID-19 pandemic has accelerated the shift towards online shopping and significantly fueled the global expansion of the e-commerce sector [1], [2]. However, this rapid growth introduces several challenges, including the widespread presence of duplicate or highly similar products across different online retailers, which complicates catalog management and may erode customer trust [3]. To mitigate this issue, various text-based duplicate detection methods have been proposed that rely on analyzing product titles and descriptions [4]–[6]. Although these approaches demonstrate some effectiveness, they frequently fail to capture essential visual attributes, such as color, pattern, and shape features, that are often crucial for accurately distinguishing between visually similar items.

To overcome the limitations of text-based methods, recent research has increasingly turned to image-based techniques for enhancing product identification and catalog organization. Consequently, numerous studies have applied image recognition methods to improve the accuracy of product grouping in online marketplaces [7]–[10]. Clustering algorithms have also been adopted to automatically group similar items, thereby facilitating improved catalog management and search efficiency [11], [12]. In parallel, advancements in deep learning, particularly in Convolutional Neural Network (CNN) architectures, have significantly enhanced visual feature extraction capabilities [13]–[18]. Furthermore, transfer learning has

proven effective in reducing the computational cost and data labeling burden by leveraging pre-trained models on large datasets [19]–[24].

Nevertheless, several limitations persist in the current literature. First, many studies rely on outdated CNN architectures such as VGG19, MobileNetV2, and ResNet-50 [25]–[27], which may lack the representational capacity to capture the nuanced visual features of contemporary product images. Second, existing clustering-based methods often assume a fixed number of clusters [28], [29], an assumption that is unrealistic in dynamic e-commerce environments where new product variants are introduced continuously.

To address these limitations, this study proposes an automated, image-driven product grouping framework that combines deep transfer learning with adaptive hierarchical clustering. The approach leverages a fine-tuned EfficientNetV2M model to extract rich and discriminative visual features from product images, outperforming conventional Convolutional Neural Networks in capturing nuanced image details. These extracted features are then processed using an adaptive agglomerative clustering algorithm, which eliminates the need to predefine the number of clusters by dynamically determining optimal grouping thresholds based on data distribution. To evaluate the effectiveness of the clustering results, the framework utilizes the Normalized Mutual Information (NMI) metric, which measures the degree of alignment between the predicted clusters and the ground-truth product categories.

2. Method

To enhance the efficiency and accuracy of product clustering, this study proposes a method that leverages transfer learning with Convolutional Neural Networks (CNNs). By fine tuning a pre-trained CNN and incorporating task-specific layers, the model is better equipped to adapt to the unique visual characteristics of the target dataset. Once product image features are extracted, Agglomerative Hierarchical Clustering is employed to group visually similar products without requiring a predefined number of clusters. This strategy improves system flexibility in accommodating new product variations and increases scalability across a wide range of product categories.

To illustrate the practical implementation of this approach, the following section details the end-to-end workflow of the proposed method. As shown in Fig. 1, the process begins with the collection of product images to construct the training dataset.

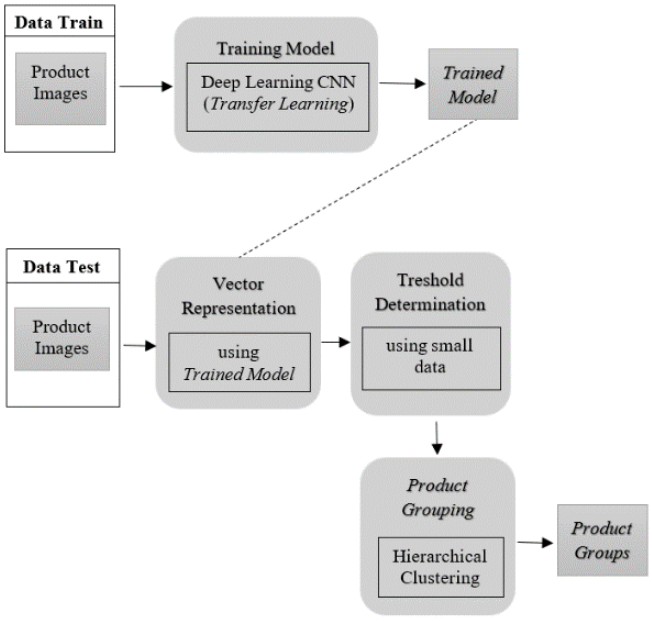


Fig. 1. Overall System Workflow Illustrating The Stages of Product Grouping Based on Image Features

These images are processed using the fine-tuned CNN model, resulting in vector representations (feature embeddings) for each image. During the testing phase, new product images undergo the same feature extraction process as existing ones. A threshold calibration step, based on a representative data subset, is then applied to determine grouping criteria. Finally, hierarchical clustering is performed on the resulting feature vectors, producing coherent product groups based on visual similarity..

2.1. Dataset

The dataset employed in this research was sourced from Kaggle [30], a platform widely recognized for facilitating knowledge exchange, learning, and data science competitions. Kaggle offers a broad array of high-quality datasets, enabling users to address real-world problems, evaluate machine learning models, and devise innovative solutions [31]. The dataset used in this study originates from Shopee, a prominent e-commerce platform in Indonesia and across Southeast Asia [32], [33].

To ensure model generalizability across diverse retail categories, the dataset encompasses a broad range of product types, including food, clothing, footwear, beauty products, baby supplies, kitchenware, health equipment, home goods, electronics, and others. Representative examples of product variations are depicted in Fig. 2. The dataset comprises 5,169 unique product entries, from which 4,634 products categorized into 281 product labels were allocated for training a deep learning model designed to extract image features. This model employs transfer learning based on a pre-trained CNN, as described in subsection 2.2.

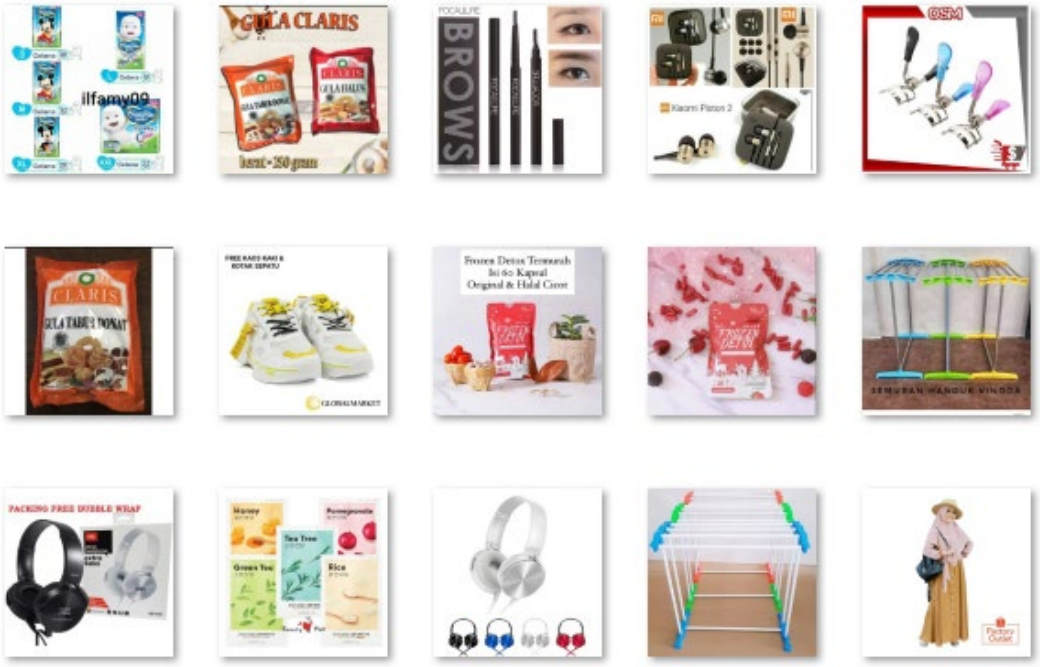


Fig. 2. Sample Product Images Across Various Categories in The Dataset

The remaining 535 products were reserved for the clustering stage. To determine appropriate parameter settings, a randomly selected subset of 103 samples was used for threshold calibration. The derived threshold was then applied to group the remaining 432 products, which were divided into four batches of 108 items each. Additional methodological specifics are provided in subsections 2.3 and 2.4.

To optimize model performance and training efficiency, the 4,634-image dataset was partitioned into training and validation sets using an 80:20 split, yielding 3,707 images for training and 927 for validation. All images were resized to 384×384 pixels to maintain a balance between computational efficiency and the preservation of visual detail. Each image was processed in RGB format, retaining three color channels (red, green, and blue). This resolution proved effective in supporting both accurate feature extraction and reliable model evaluation.

2.2. Modified Convolutional Neural Network Model

For this research, we employed a deep learning model provided by Keras Applications [34], [35]. Specifically, we utilized a modified version of EfficientNetV2M, a CNN architecture recognized for its compound scaling strategy, which simultaneously optimizes depth, width, and input resolution. This architectural advantage makes it particularly well-suited for large-scale product image classification tasks, where both high accuracy and computational efficiency are essential. Our previous study [36] identified EfficientNetV2M as the best-performing model among several alternatives, including VGG16 and MobileNetV2, thereby motivating its adoption in the present work.

To tailor the model for classifying diverse e-commerce product images, several architectural modifications were introduced. The original classification head was replaced with a fully connected output layer corresponding to the number of product categories. To reduce the risk of overfitting, dropout regularization and batch normalization were incorporated into the modified layers. A transfer learning strategy was applied by freezing the early convolutional layers, preserving general feature representations, while fine-tuning the later blocks and classification head to capture domain-specific patterns. The model was trained using the Adam optimizer with a manually tuned constant learning rate. Empirical results indicated that training over 20 epochs with a batch size of 200 yielded stable convergence without signs of overfitting.

The rationale behind these architectural and training choices is summarized in Table 1, which illustrates how the model was configured to strike a balance between efficiency and adaptability. This setup reflects a deliberate trade-off between leveraging pre-trained knowledge and adapting to domain-specific data. By freezing 684 out of 698 layers of the EfficientNetV2M architecture, the model retains the general visual representations learned from large-scale datasets, while allowing fine-tuning of only 14 layers to capture domain-specific nuances in e-commerce product images. This approach reduces computational cost and mitigates the risk of overfitting on a relatively limited training dataset. The decision to add four custom layers provides task-specific capacity without excessively increasing model complexity. Dropout was employed as a regularization technique to prevent overfitting, particularly important given the relatively small number of trainable parameters compared to the total model size. The choice of training the model for 20 epochs was based on empirical evaluation, ensuring sufficient convergence while avoiding over-training. Overall, this configuration was designed to maximize feature transfer efficiency, maintain model stability, and adapt the deep architecture to the visual diversity present in real-world e-commerce catalogs.

Table 1. Summary of Training Configuration and Model Setup

Description	Modified EfficientNetV2M
Number of Layers	698
Trainable layers	14
Frozen Layer	684
Total Parameters	108,531,269
Trainable Parameters	57,612,689
Added Layers	4
Overfitting Handler	Dropout
Number of Epochs	20

To validate the effectiveness of this setup, the model's classification performance was evaluated using standard metrics, as summarized in Table 2. The results show that the model achieved an accuracy of 84%, indicating a high level of agreement between the predicted labels and the ground truth. The weighted average F1-score, also at 84%, reflects a balanced performance across all classes, which is particularly important in the presence of class imbalance. A precision score of 86% suggests that the model effectively minimized false positives, while the recall value of 84% demonstrates its ability to capture the majority of actual positive cases. These outcomes indicate that the model performs reliably in general classification tasks and remains consistent even when distinguishing visually similar product categories, reinforcing its suitability for deployment in practical e-commerce scenarios.

Table 2. Evaluation Metrics of The Proposed Model

Description	Value
Accuracy	84%
Weighted average F1-Score	84%
Precision	86%
Recall	84%

These findings are consistent with prior research [37], [38], which demonstrates that CNN-based deep learning methods are highly effective in extracting meaningful visual features from product images. Furthermore, the use of transfer learning, which leverages pre-trained models trained on diverse image corpora, has been shown to improve classification performance across various domains significantly [39]–[41].

To further assess the model's discriminative capability in a multi-class setting, Fig. 3 presents a 3D visualization of the Receiver Operating Characteristic (ROC) curve, where the X-axis denotes the False Positive Rate (FPR), the Y-axis represents the True Positive Rate (TPR), and the Z-axis corresponds to the Area Under the Curve (AUC). This multi-dimensional perspective reveals the trade-offs between sensitivity and specificity across different thresholds. The visualization highlights the ten classes with the highest AUC scores, offering a focused view of the model's ability to distinguish between visually similar product categories.

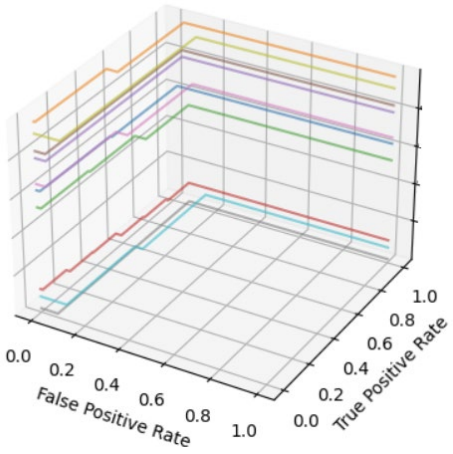


Fig. 3. 3D ROC Curve Visualization of The Top 10 Classes with Highest AUC Scores

To formally define the feature extraction process, Equation (1) provides a mathematical representation of the process:

$$Z_i^{img} = f_{img}(X_i^{img}; \theta_{cnn}) \quad (1)$$

where Z_i^{img} is image feature vector, f_{img} is CNN-based feature extraction function (EfficientNetV2M), X_i^{img} is Input image of the i-th product, and θ_{cnn} is Fine-tuned model parameters.

Each product image is modeled as a three-dimensional tensor of size $H \times W \times C$, where H, W, and C represent the height, width, and number of color channels (RGB), respectively. These tensors are passed through the CNN, pre-trained on a large-scale dataset and fine-tuned for the current classification task, undergoing successive layers of convolution and pooling. The final output is a compact, high-dimensional vector representation that serves as input for the downstream clustering operations discussed in the following section.

2.3. Agglomerative Clustering

In this study, we implemented Agglomerative Hierarchical Clustering to group products based on their visual features. This method was chosen for its ability to produce dendrograms, which not only

reveal hierarchical relationships among items but also offer flexibility in selecting the desired level of granularity a significant advantage when dealing with heterogeneous product categories [28], [42], [43].

Unlike flat clustering techniques such as k-means, Agglomerative Clustering does not require the number of clusters to be predefined. Instead, it iteratively merges the most similar clusters based on the chosen distance metrics and linkage criteria. However, the effectiveness of this approach is highly sensitive to the choice of these parameters [44]. Building upon our earlier findings [45], we systematically evaluated various combinations of distance measures (Euclidean, Manhattan, and Cosine) and linkage methods (Ward, Complete, Single, and Average).

To identify optimal clustering configurations, we conducted a comparative evaluation of these combinations based on cluster coherence and external validation metrics. Our analysis revealed that two specific configurations consistently outperformed others: Cosine Similarity with Complete Linkage and Euclidean Distance with Ward Linkage. These findings corroborate our previous work [45] and were further supported by high NMI scores. Cosine Similarity was particularly effective in capturing directional relationships in high-dimensional vector spaces. In contrast, Euclidean Distance reflected spatial proximity, both of which are essential when assessing visual features encoded as embeddings [46].

To further examine the effect of feature quality on clustering outcomes, we compared visual representations extracted from two CNN models: the baseline Vanilla EfficientNetV2M (pre-trained on ImageNet) and the Modified EfficientNetV2M introduced in Section 2.2. A detailed comparison is provided in Table 3, which shows that the modified model yields more compact and semantically meaningful feature embeddings. These findings suggest that domain-specific fine-tuning improves the model's ability to distinguish visual patterns, ultimately enhancing the quality and interpretability of product groupings.

Table 3. Comparison of Image Feature Distances Using Vanilla and Modified EfficientNetV2M

Picture1	Picture2	Model	Euclidean Distance	Cosine Similarity
		Vanilla EfficientNetV2M	258.79	0.52
		Modified EfficientNetV2M	240.93	0.42
		Vanilla EfficientNetV2M	406.52	0.07
		Modified EfficientNetV2M	353.18	0.01
		Vanilla EfficientNetV2M	386.20	0.09
		Modified EfficientNetV2M	344.51	0.01

A more detailed analysis of these results is provided in Table 3, which compares the feature distances between image pairs using both model variants. The evaluation focuses on Euclidean distance and Cosine similarity to quantify the quality of the visual representations. In all three image pairs, the Modified EfficientNetV2M consistently produced lower values across both metrics when compared to the Vanilla version. These results indicate that the modified model is more effective at encoding meaningful visual distinctions, particularly in differentiating dissimilar product categories. For example, in the second and third image pairs comparing baby diapers with headphones the Modified EfficientNetV2M achieved notably lower Cosine similarity scores (0.01) compared to the Vanilla model (0.07 and 0.09, respectively),

signifying more apparent separation between unrelated items. Likewise, the reduced Euclidean distances for similar items, such as different diaper packages, reflect tighter intra-class clustering. Overall, these findings reinforce the benefit of domain-specific fine-tuning in enhancing the discriminative power of learned features and improving the reliability of subsequent clustering operations.

2.4. Setting the Distance Threshold Parameters

To determine the most suitable parameters for grouping products into clusters, we utilized a randomly selected subset of 103 product samples from the test dataset. The image features of these samples were extracted using the previously described Modified EfficientNetV2M model. These feature vectors were then used to construct a dendrogram using the Agglomerative Hierarchical Clustering method.

This calibration step served as the basis for identifying a clustering threshold that reflects the inherent structure of the data. The threshold was established according to the known number of product groups within the subset and corresponds to a horizontal cut in the dendrogram that captures the natural separation between clusters (as indicated by the yellow dashed line in Fig. 4 and Fig. 5). This thresholding approach ensures that the resulting clusters align closely with the semantic structure of the dataset.

Table 4 summarizes the optimal clustering thresholds identified for each combination of distance metric and linkage method across both the Vanilla and Modified EfficientNetV2M models. The results show that for both models, the best-performing linkage configurations were Cosine Similarity with Complete Linkage and Euclidean Distance with Ward Linkage. Notably, the Modified EfficientNetV2M required higher threshold values compared to the Vanilla model for both metrics, 0.975 versus 0.915 for Cosine Similarity, and 430 versus 450 for Euclidean Distance. This shift suggests that the modified model generates more compact and semantically coherent feature embeddings, allowing for tighter intra-cluster grouping without compromising inter-cluster separability. The identification of these thresholds is critical for determining where to cut the dendrogram during agglomerative clustering, as they directly influence the granularity and accuracy of the resulting product groupings. These findings further support the effectiveness of the modified architecture in improving visual feature representation and its impact on downstream clustering performance.

Table 4. Best Clustering Thresholds for Each Distance Metric and Linkage Combination

Model	Metric	Linkage	Best Threshold
Vanilla EfficientNetV2M	Cosine Similarity	Complete Linkage	0.915
	Euclidean Distance	Ward Linkage	450
Modified EfficientNetV2M	Cosine Similarity	Complete Linkage	0.975
	Euclidean Distance	Ward Linkage	430

To formalize the segmentation process, Equation (2) presents the mathematical formulation for cutting the dendrogram to obtain discrete cluster assignments:

$$C = cut_{tree}(D, t) \tag{2}$$

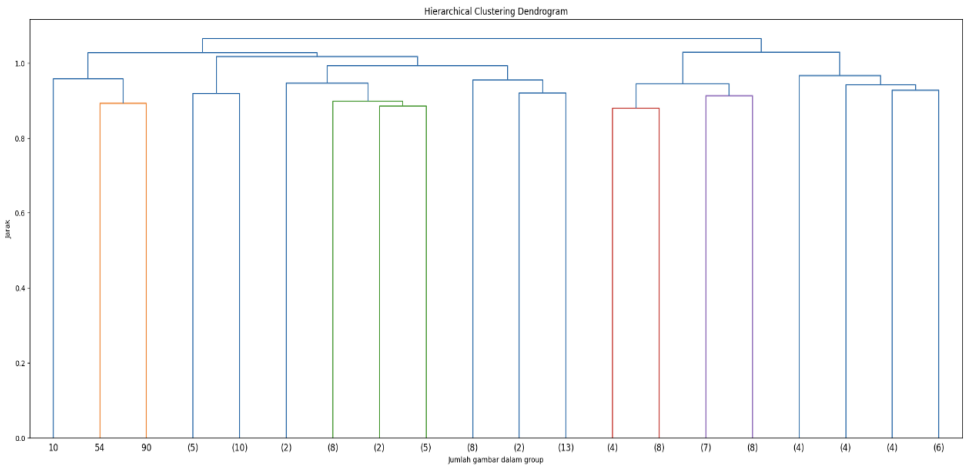
where C is resulting set of clusters, D is pairwise distance matrix between product images, and t is distance threshold used to segment the dendrogram.

This expression encapsulates the fundamental principle of Agglomerative Clustering: initially treating each product as an individual cluster and progressively merging the most similar pairs according to a chosen distance metric (e.g., Euclidean, Manhattan, Cosine) and linkage criterion (e.g., Complete, Average, Ward), until a complete dendrogram is constructed. By applying a cut at a specified threshold t, the resulting clusters reflect underlying visual similarity without requiring a predefined number of groups. This threshold-based segmentation approach offers flexibility and adaptability, making it particularly suitable for dynamic product datasets where the number and type of items may vary over

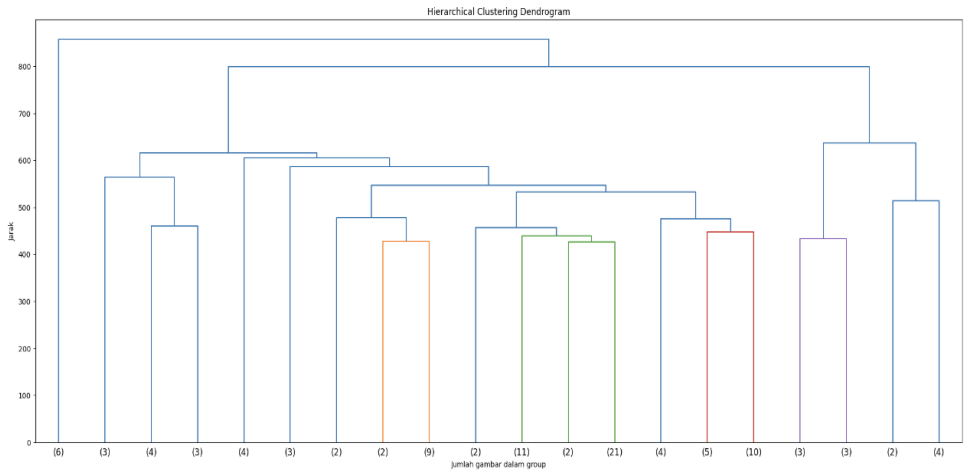
time. Moreover, its data-driven nature enables the system to adapt to evolving inventories, making it ideal for practical e-commerce deployments where scalability and automation are essential. As such, the interpretability of dendrograms becomes a critical asset in evaluating both the structure and validity of the resulting clusters.

The structure and behavior of the clustering process are visually represented in Fig. 4 and Fig. 5. In these dendrograms, the vertical axis indicates inter-cluster distances, while the horizontal axis corresponds to merged nodes, with each node representing the size of the resulting cluster. Default dendrogram links are shown in blue, and branches of the same color represent the final merged clusters. These visualizations highlight how varying parameter configurations influence clustering outcomes, as detailed in Table 4.

To deepen the analysis, a comparative evaluation of clustering structures was conducted across both the baseline and modified models, focusing on how specific metric–linkage combinations affect the formation and separation of product groups. By visualizing the dendrograms produced under different parameter settings, the study aims to assess not only quantitative outcomes, such as threshold values and NMI scores, but also qualitative differences in cluster compactness and clarity. These visual comparisons offer valuable insights into the practical implications of model design choices, particularly in distinguishing fine-grained product variations within high-dimensional visual feature spaces.



(a)



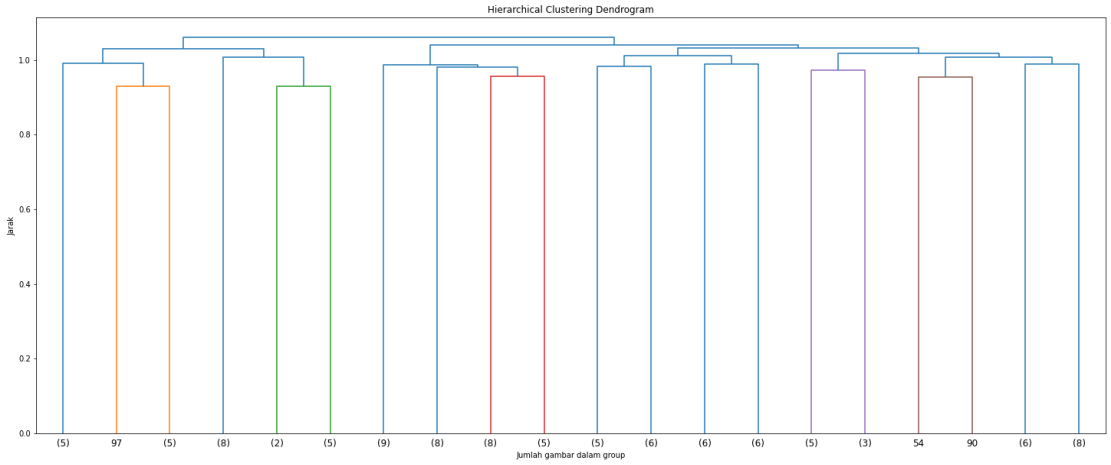
(b)

Fig. 4. Dendrogram of Vanilla EfficientNetV2M for 103 Test Samples : (a) Using Metric Cosine Similarity and Complete Linkage (b) Using Metric Euclidean Distance and Ward Linkage

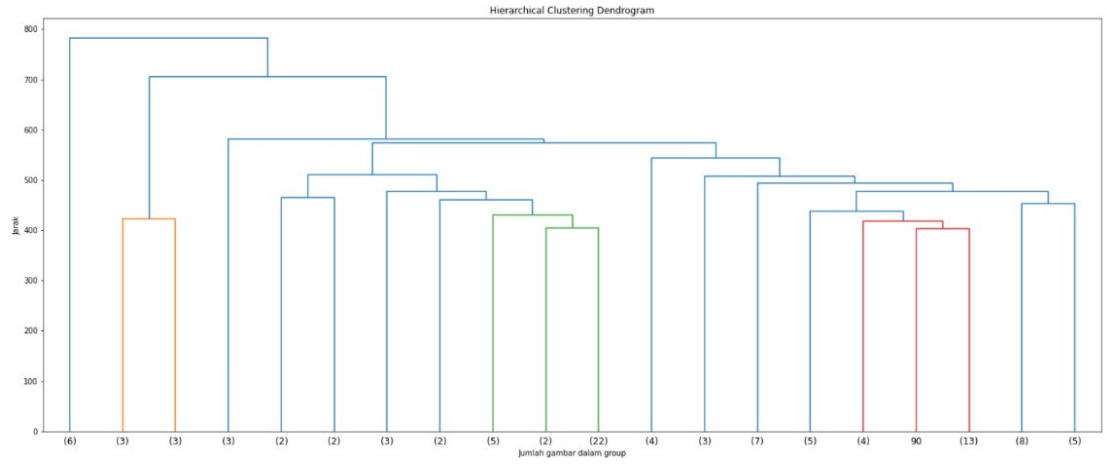
Fig. 4 illustrates the clustering behavior of the Vanilla EfficientNetV2M model using two configurations: (a) Cosine Similarity with Complete Linkage, and (b) Euclidean Distance with Ward

Linkage. These dendrograms visualize the hierarchical relationships among the 103 test samples, with the orange dashed line marking the threshold level determined during calibration. In subfigure (a), the clustering structure is relatively shallow, suggesting moderate intra-class compactness under cosine-based similarity. In contrast, subfigure (b) exhibits deeper and more differentiated branches, indicating improved separation between clusters when using Euclidean distance and Ward linkage. This visual contrast reinforces the impact of the distance metric and linkage selection on clustering behavior, supporting the numerical differences in optimal threshold values previously shown.

While Fig. 4 provides insight into the baseline model's ability to structure data hierarchically, it also reveals certain limitations in capturing fine-grained visual distinctions among product categories. The relatively less compact cluster formations and broader inter-cluster spacing suggest that the Vanilla EfficientNetV2M, though pre-trained on large-scale datasets, lacks the specialization needed for high-resolution discrimination in domain-specific contexts such as e-commerce. These observations raise an important question regarding the extent to which feature refinement, through model adaptation, can enhance clustering performance. To explore this, Fig. 5 presents the results of applying the same clustering procedures to embeddings generated by the Modified EfficientNetV2M model, thereby enabling a direct comparison between generic and fine-tuned representations.



(a)



(b)

Fig. 5. Dendrogram of Modified EfficientNetV2M for 103 Test Samples: (a) Using Metric Cosine Similarity

Fig. 5 displays the clustering results obtained using the Modified EfficientNetV2M model, offering a visual representation of how its fine-tuned feature embeddings influence the grouping structure. In subfigure (a), which uses Cosine Similarity and Complete Linkage, the resulting dendrogram shows a more precise separation between clusters, with more balanced branch heights and minimal overlap near

the threshold. This improved clustering structure suggests that the modified model produces representations that are not only compact within clusters but also better aligned across semantically similar items. Subfigure (b), which applies Euclidean Distance with Ward Linkage, reveals a more refined hierarchy compared to its Vanilla counterpart in Fig. 4, with multiple clusters forming below the threshold line, indicating early consolidation of well-separated groups. Notably, significant and consistent clusters, such as the one containing 22 samples, emerge more distinctly, highlighting the model's improved ability to capture intra-class consistency. These patterns reinforce the earlier quantitative findings and illustrate how domain-specific fine-tuning enhances the clustering structure, making it more reflective of underlying semantic categories in e-commerce product data.

2.5. Calculating Normalized Mutual Information Values

To evaluate the quality of clustering results, we employed the NMI score, a widely used metric in cluster analysis for measuring the similarity between predicted clusters and ground truth labels [47]–[49]. NMI quantifies the amount of shared information between two clusterings, producing values ranging from 0 to 1. A score of 0 indicates no mutual information, implying entirely dissimilar clusters, while a score of 1 denotes perfect agreement between the clusterings.

This metric was selected due to its effectiveness in quantifying clustering performance across varying parameter configurations. In this study, NMI values were used to assess the consistency and accuracy of clustering outcomes under different distance and linkage combinations. Higher NMI scores suggest that the clustering algorithm effectively grouped similar data points and captured the latent structure in the dataset [50], [51]. Compared to alternative evaluation metrics, NMI has demonstrated greater reliability in assessing clustering accuracy and resolving cluster labeling challenges [52], [53].

The mathematical formulation used to compute the NMI score is provided in Equation (3)

$$NMI = \frac{2xMI(C,Y)}{H(C)+H(Y)} \tag{3}$$

Where;

$MI(C,Y)$ = Denotes the Mutual Information between the predicted clusters C and the ground truth clusters Y.

$H(C)$ = Is the entropy of the predicted clusters

$H(Y)$ = Is the entropy of the ground truth clusters

In this Equation, Mutual Information (MI) reflects the shared information between the predicted and actual clusterings, while entropy measures the degree of uncertainty within each clustering. The NMI value reaches its upper bound of 1 when the clustering output perfectly matches the ground truth and falls to 0 when no overlap exists between the two distributions.

Table 5 presents the NMI scores for various combinations of models, distance metrics, and linkage methods, providing a quantitative assessment of clustering quality. The Modified EfficientNetV2M consistently outperforms its Vanilla counterpart across both configurations, with the highest NMI score of 0.938 achieved when using Cosine Similarity and Complete Linkage. This score indicates a strong alignment between predicted clusters and ground truth labels, confirming the effectiveness of domain-specific fine-tuning in enhancing feature discrimination. In contrast, the Vanilla EfficientNetV2M exhibits considerably lower NMI values, 0.752 with Cosine Similarity and 0.653 with Euclidean Distance, indicating that its general-purpose features are less effective in accurately grouping visually similar e-commerce products. Notably, even within the same model architecture, Cosine Similarity paired with Complete Linkage consistently yields better performance than Euclidean Distance with Ward Linkage. This pattern implies that orientation-based similarity (as captured by cosine metrics) is more effective for high-dimensional feature spaces generated by deep CNNs. The results reinforce earlier visual and threshold-based findings, validating the superiority of the modified model and highlighting the critical role of metric–linkage pairing in optimizing clustering outcomes.

Table 5. NMI Scores Across Different Models and Clustering Parameter Configurations

Model	Metric	Linkage	NMI Score
Vanilla EfficientNetV2M	Cosine Similarity	Complete Linkage	0.752
	Euclidean Distance	Ward Linkage	0.653
Modified EfficientNetV2M	Cosine Similarity	Complete Linkage	0.938
	Euclidean Distance	Ward Linkage	0.767

3. Results and Discussion

The experimental results demonstrate that the proposed method, combining the Modified EfficientNetV2M model for feature extraction with Agglomerative Hierarchical Clustering, delivers strong performance in organizing visually similar product images. As shown in Table 5, the best configuration, using Cosine Similarity and Complete Linkage, achieved an NMI score of 0.938, indicating a high degree of alignment between the predicted clusters and the actual product labels. Notably, the Modified EfficientNetV2M model significantly outperformed its Vanilla counterpart, highlighting its capacity to generate semantically richer feature embeddings. These findings suggest that fine-tuning the CNN on domain-specific data enhances the quality of visual representation, ultimately improving the effectiveness of clustering in unsupervised settings.

To further assess the robustness and scalability of the system, we evaluated its performance under incremental data expansion, an essential consideration for real-world deployment. Specifically, we simulated a dynamic e-commerce scenario by progressively introducing new product data in four separate batches. As reported in Table 6, clustering performance improved consistently with each batch, culminating in an NMI score of 0.970 on Batch 4 using the Modified EfficientNetV2M model. This upward trend indicates that the system not only generalizes well to previously unseen data but also benefits from increased diversity, which enriches the learned feature space. Such adaptability is particularly valuable for e-commerce platforms where product catalogs evolve continuously. A detailed breakdown of the NMI scores for each configuration across the batches is provided in Table 6.

Table 6. NMI Scores for Each Clustering Batch Using The Top Two Model Parameter Configurations

Model	Metric	Linkage	Data Batch	NMI Score
Vanilla EfficientNetV2M	Cosine Similarity	Complete Linkage	Batch-1	0.760
			Batch-2	0.805
			Batch-3	0.747
			Batch-4	0.836
	Euclidean Distance	Ward Linkage	Batch-1	0.809
			Batch-2	0.899
			Batch-3	0.886
			Batch-4	0.931
Modified EfficientNetV2M	Cosine Similarity	Complete Linkage	Batch-1	0.911
			Batch-2	0.921
			Batch-3	0.939
			Batch-4	0.925
	Euclidean Distance	Ward Linkage	Batch-1	0.812
			Batch-2	0.920
			Batch-3	0.883
			Batch-4	**0.970**

Table 6 presents the NMI scores across four successive clustering batches for the top two model–parameter configurations. The results demonstrate the superior performance and scalability of the Modified EfficientNetV2M model over its Vanilla counterpart. Notably, the Modified model, paired with Cosine Similarity and Complete Linkage, consistently outperformed all other configurations,

achieving remarkably high NMI scores, ranging from 0.911 in Batch 1 to 0.939 in Batch 3. Although there is a slight dip in Batch-4 (0.925), the model maintained strong consistency across all iterations, indicating stable and reliable clustering performance even as the dataset grew.

Equally significant is the Modified model using Euclidean Distance with Ward Linkage, which achieved the highest single score of 0.970 in Batch-4. This peak score suggests that as more data is introduced, the model benefits from increased diversity and becomes better at capturing the latent structure of the visual feature space. In contrast, the Vanilla EfficientNetV2M yielded lower and more fluctuating scores in both configurations, with Cosine–Complete ranging between 0.747 and 0.836, and Euclidean–Ward from 0.809 to 0.931. While the Vanilla model still showed improved clustering as batches progressed, its gains were less pronounced, highlighting the limited capacity of non-fine-tuned architectures to adapt to domain-specific visual patterns.

Overall, these results reinforce the advantage of domain-adapted models in scalable clustering tasks, particularly in dynamic e-commerce environments where product inventories are frequently updated. The steady improvement in NMI across batches for the Modified model also suggests its robustness in generalizing to unseen product images and maintaining high intra-class consistency without supervision.

Table 7 presents the average NMI scores across all four clustering batches, providing a holistic view of each model's performance under different parameter configurations. The Modified EfficientNetV2M model consistently outperformed the Vanilla version, achieving an average NMI of 0.924 with Cosine Similarity and Complete Linkage, the highest overall score. This configuration demonstrated both robustness and adaptability, reflecting the model's enhanced capacity to capture consistent semantic groupings across diverse and expanding data subsets. Notably, even with Euclidean Distance and Ward Linkage, the modified model maintained strong performance (NMI average: 0.896), further highlighting its ability to generate compact yet discriminative visual embeddings suitable for hierarchical clustering.

Table 7. Average NMI Scores Across All Batches by Model and Clustering Parameter

Model	Metric	Linkage	NMI Average
Vanilla EfficientNetV2M	Cosine Similarity	Complete Linkage	0.787
	Euclidean Distance	Ward Linkage	0.881
Modified EfficientNetV2M	Cosine Similarity	Complete Linkage	0.924
	Euclidean Distance	Ward Linkage	0.896

The strong alignment between this clustering configuration and the nature of e-commerce data is particularly noteworthy. Product images on e-commerce platforms often exhibit high intra-class variation (e.g., different angles, lighting conditions, or packaging updates) but low inter-class margins, making orientation-sensitive similarity measures, such as Cosine Similarity, especially advantageous. Combined with Complete Linkage, which emphasizes maximal pairwise distance within clusters, this method helps preserve cluster tightness and avoids premature merging of dissimilar items. Meanwhile, the Modified EfficientNetV2M, having been fine-tuned on domain-specific data, captures nuanced visual patterns more effectively than the generic Vanilla version. The match between the data and methods helps explain why the Cosine–Complete combination performs so well and consistently. In contrast, the Vanilla EfficientNetV2M model produced lower average NMI scores under both configurations. Although the Euclidean–Ward pairing yielded relatively strong results (0.881), it still lagged behind the modified model, underscoring the limitations of relying solely on pre-trained features without domain adaptation. The lowest performance was observed in the Vanilla model using Cosine–Complete (0.787), suggesting that without specialized tuning, this configuration lacks the necessary discriminative power to handle the complexity of real-world product data. Overall, these findings confirm that both the model architecture and clustering design must be carefully matched to the intrinsic structure of the dataset to enable scalable and reliable product grouping in dynamic e-commerce environments.

Beyond the quantitative metrics, the architectural advantages of the proposed approach further underscore its practical value. EfficientNetV2M's compound scaling strategy, coupled with its greater representational capacity, enables it to capture fine-grained visual nuances that are often overlooked by shallower CNNs. When combined with transfer learning, the model adapts efficiently to domain-specific imagery with minimal supervision, an essential characteristic for deployment in large-scale, heterogeneous environments where labeled data is limited or incomplete. In contrast to conventional CNN models such as VGG16 or MobileNetV2, which were evaluated in our prior work [36], the Modified EfficientNetV2M consistently yielded more robust and generalizable representations, remarkably when fine-tuned on the target dataset. While earlier architectures may suffice for smaller, well-curated datasets, they often lack the depth and flexibility required to address the complexities of real-world e-commerce data.

Moreover, our approach provides significant advantages over fully supervised models designed for static, closed-set classification. For instance, [54] reports a six-layer CNN trained from scratch on 1,050 labeled product images for a fixed-label classification task, achieving an accuracy of 91.37%. While effective in constrained settings, such methods lack scalability in open-world environments, where new and unlabeled products are frequently introduced. In contrast, our unsupervised clustering framework accommodates these variations without requiring exhaustive labeling or prior knowledge of the number of classes, making it a more practical solution for dynamic, large-scale retail systems. From an application perspective, the proposed system delivers several tangible benefits for e-commerce operations. First, it facilitates catalog consolidation by automatically grouping visually similar or duplicate products, thereby reducing redundancy and simplifying inventory management. Additionally, the clustering of related items enhances product discovery, resulting in more relevant search results and a broader range of recommendations for users. Furthermore, the system significantly lowers annotation costs, as it relies on unsupervised clustering rather than fully labeled datasets, making it scalable and cost-effective for large and constantly evolving product catalogs.

Table 8 and Table 9 present qualitative examples of clustering errors encountered by the Vanilla and Modified EfficientNetV2M models under their respective optimal parameter configurations. These misclassifications provide insight into the challenges of relying solely on visual similarity for unsupervised product grouping. As shown in Table 8, the Vanilla model often failed to distinguish semantically distinct items that share superficial visual characteristics. For instance, an electric stove was grouped with egg beaters due to similar metallic surfaces and rounded contours. At the same time, a liquid lipstick was mistakenly grouped with baby diapers, likely due to a similarity in color scheme or packaging structure. In some cases, such as antiseptic gel, both tested configurations failed to differentiate it from unrelated product categories, suggesting insufficient feature discrimination in the baseline model.

Table 9 demonstrates that even the Modified EfficientNetV2M, while yielding stronger performance overall, was not immune to errors. Misclustering often occurred in categories with overlapping packaging or ambiguous visual forms. For example, a frozen detox drink was incorrectly grouped with sugar-dust coating products, possibly due to similar red-and-white packaging themes. Likewise, olive oil bottles were misclustered with clotheslines, indicating confusion caused by vertical alignments and shared structural layouts. In another instance, headsets and eyelash curlers, both comprising compact, curved forms, were misclassified under both linkage configurations, reflecting persistent difficulties in distinguishing geometrically similar but semantically unrelated objects.

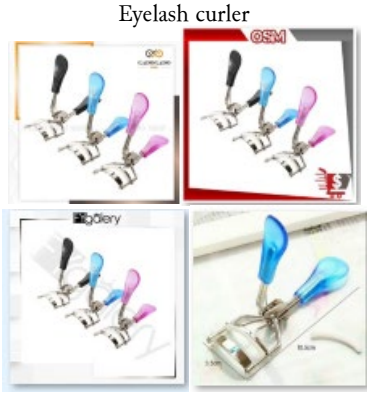
Despite these advances, the system exhibits apparent limitations, particularly in edge cases where visual ambiguity leads to semantically inconsistent clusters. These issues often arise in product domains characterized by subtle inter-class distinctions or high intra-class variability. Further analysis attributes these failures not only to model constraints but also to data-related factors. Low image quality, such as poor resolution, inconsistent lighting, and visually cluttered backgrounds, can obscure discriminative cues, impairing the CNN's ability to encode meaningful representations. In other scenarios, class overlap or vague labeling contributed to model confusion, especially among consumer goods such as beauty accessories and kitchen utensils, where visual and functional boundaries are less well-defined.

Table 8. Clustering Errors Using Vanilla EfficientNetV2M with Top Two Parameter Settings

Types of Failure	Miss Clustered Product Image	Actual Group Cluster	Result Group Cluster
Recognition failed using Cosine Similarity - Complete Linkage.		Electric Stove 	Egg Beater 
Recognition failed using Euclidean Distance - Ward Linkage.		Liquid Lipstick 	Baby Diapers 
Recognition Failed using Cosine Similarity - Complete Linkage and Euclidean Distance - Ward Linkage		Antiseptic Gel 	Baby Diapers 

These findings highlight a fundamental limitation of vision-only clustering systems: their limited resilience in uncontrolled, real-world environments, such as user-generated e-commerce listings. In such contexts, image features alone may fail to convey category-relevant semantics, particularly when appearance-based similarity does not align with product function or taxonomy. To mitigate these issues, future work should consider multimodal strategies that incorporate auxiliary metadata, such as product names, category tags, or descriptions, into the clustering pipeline. Complementary techniques, including image enhancement, background subtraction, and outlier detection, can further enhance clustering robustness and aid in identifying ambiguous instances that require human verification. Such measures are essential for enhancing the system's scalability and reliability in dynamic online retail environments.

Table 9. Clustering Errors Using Modified EfficientNetV2M with Top Two Parameter Settings

Types of Failure	Miss Clustered Product Image	Actual Group Cluster	Result Group Cluster
Recognition failed using Cosine Similarity - Complete Linkage.		Frozen Detox 	Sugar-dust Coating 
Recognition failed using Euclidean Distance - Ward Linkage.		Olive Oil 	Clothesline 
Recognition Failed using Cosine Similarity - Complete Linkage and Euclidean Distance - Ward Linkage		Headset 	Eyelash curler 

4. Conclusion

Based on the findings of this research, several key conclusions can be drawn. The Modified EfficientNetV2M architecture demonstrated superior performance in extracting discriminative visual features compared to the Vanilla version, underscoring the importance of domain-specific fine-tuning in improving the quality of image representations for clustering tasks. Among the various clustering configurations explored, the combinations of Cosine Similarity with Complete Linkage and Euclidean Distance with Ward Linkage consistently produced the most coherent and semantically meaningful clusters. After determining the optimal clustering parameters, the model's scalability was further validated by incrementally adding new product data. The results confirmed that newly introduced items could be successfully integrated into existing clusters or allocated to new ones, highlighting the system's adaptability in dynamic inventory environments. The highest clustering performance was achieved by pairing the Modified EfficientNetV2M with Cosine Similarity and Complete Linkage, yielding a peak NMI score of 0.924 and outperforming all other configurations across evaluation batches. In practical e-

commerce applications, the proposed approach offers tangible operational advantages. By automatically grouping visually similar or duplicate products, the system can enhance catalog organization and inventory management while reducing redundancy. From a user experience perspective, such clustering contributes to improved personalization, search relevance, and navigation, ultimately increasing user satisfaction by presenting more visually relevant alternatives. Additionally, the approach supports more effective marketing strategies by identifying product groupings that facilitate targeted promotions and more accurate customer segmentation, leading to enhanced engagement and higher conversion rates. Nonetheless, the consistency of clustering performance may vary across product categories, particularly in visually diverse domains such as fashion, where intra-class variability is high. To address these limitations, future research should explore the expansion of dataset size and diversity, the refinement of image feature extraction techniques, the integration of hybrid models, and the incorporation of auxiliary metadata such as product titles and descriptions to provide deeper semantic context and improve clustering accuracy.

Acknowledgment

The authors thank Universitas Negeri Malang, Indonesia, and Institut Sains dan Teknologi Terpadu Surabaya, Indonesia, for their support of this work.

Declarations

Author contribution. Yuliana Melita Pranoto: conceptualization, methodology, writing, data curation, data analysis, software, and data evaluation. Anik Nur Handayani, Heru Wahyu Herwanto, and Yosi Kristian: supervision, conceptualization, and writing review
Funding statement. This research is self-funded
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

References

[1] G. Dionysiou, K. Fouskas, and D. Karamitros, "The Impact of Covid-19 in E-Commerce. Effects on Consumer Purchase Behavior," *Springer Proc. Bus. Econ.*, pp. 199–210, 2021, doi: 10.1007/978-3-030-66154-0_22.

[2] W. Chmielarz, M. Zborowski, J. Xuetao, M. Atasever, and J. Szpakowska, "Covid-19 Pandemic as Sustainability Determinant of e-Commerce in the Creation of Information Society," *Procedia Comput. Sci.*, vol. 207, pp. 4378–4389, 2022, doi: 10.1016/j.procs.2022.09.501.

[3] N. Valstar, F. Frasincar, and G. Brauwers, "APFA: Automated product feature alignment for duplicate detection," *Expert Systems with Applications*, vol. 174. Elsevier, 2021, doi: 10.1016/j.eswa.2021.114759.

[4] Z. Zhang and X. Song, "An Exploratory Study on Utilising the Web of Linked Data for Product Data Mining," *SN Comput. Sci.*, vol. 4, no. 1, p. 15, 2023, doi: 10.1007/s42979-022-01415-3.

[5] R. A. Asmara *et al.*, "YOLO-based object detection performance evaluation for automatic target aimbot in first-person shooter games," *Bull. Electr. Eng. Informatics*, vol. 13, no. 4, pp. 2456–2470, 2024, doi: 10.11591/eei.v13i4.6895.

[6] L. Renaningtyas, P. Dwitasari, and N. Ramadhani, "Implementing The Use of AI for Analysis and Prediction in the Fashion Industry," *Acad. Res. Community Publ.*, vol. 7, no. 1, 2023, doi: 10.21625/archive.v7i1.928.

[7] T. Widiyaningtyas, D. Dwi Prasetya, and H. W. Herwanto, "Time Loss Function-based Collaborative Filtering in Movie Recommender System," *Int. J. Intell. Eng. Syst.*, vol. 16, no. 6, pp. 1021–1030, Dec. 2023, doi: 10.22266/ijies2023.1231.84.

[8] R. A. Harianto, Y. M. Pranoto, and T. P. Gunawan, "Data Augmentation and Faster RCNN Improve Vehicle Detection and Recognition," in *3rd 2021 East Indonesia Conference on Computer and Information Technology, EIConCIT 2021*, 2021, pp. 128–133, doi: 10.1109/EIConCIT50028.2021.9431863.

[9] M. N. Mohammad, C. U. Kumari, A. S. D. Murthy, B. O. L. Jagan, and K. Saikumar, "Implementation of online and offline product selection system using FCNN deep learning: Product analysis," *Mater. Today Proc.*, vol. 45, pp. 2171–2178, 2021, doi: 10.1016/j.matpr.2020.10.072.

[10] M. Mousavizadeh, M. Koohikamali, M. Salehan, and D. J. Kim, "An Investigation of Peripheral and Central Cues of Online Customer Review Voting and Helpfulness through the Lens of Elaboration Likelihood Model," *Inf. Syst. Front.*, vol. 24, no. 1, pp. 211–231, 2022, doi: 10.1007/s10796-020-10069-6.

[11] N. Chaudhuri, G. Gupta, V. Vamsi, and I. Bose, "On the platform but will they buy? Predicting customers' purchase behavior using deep learning," *Decis. Support Syst.*, vol. 149, 2021, doi: 10.1016/j.dss.2021.113622.

[12] C. Wang, "Efficient customer segmentation in digital marketing using deep learning with swarm intelligence approach," *Inf. Process. Manag.*, vol. 59, no. 6, p. 103085, 2022, doi: 10.1016/j.ipm.2022.103085.

[13] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognit. Lett.*, vol. 141, pp. 61–67, 2021, doi: 10.1016/j.patrec.2020.07.042.

[14] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, p. 53, Mar. 2021, doi: 10.1186/s40537-021-00444-8.

[15] A. T. Hermawan, I. A. E. Zaeni, A. P. Wibawa, Gunawan, W. H. Hendrawan, and Y. Kristian, "A Multi Representation Deep Learning Approach for Epileptic Seizure Detection," *J. Robot. Control*, vol. 5, no. 1, pp. 187–204, 2024, doi: 10.18196/jrc.v5i1.20870.

[16] S. Srivastava, A. V. Divekar, C. Anilkumar, I. Naik, V. Kulkarni, and V. Pattabiraman, "Comparative analysis of deep learning image detection algorithms," *J. Big Data*, vol. 8, no. 1, p. 66, Dec. 2021, doi: 10.1186/s40537-021-00434-w.

[17] D. Widjojo, E. Setyati, and Y. Kristian, "Integrated Deep Learning System for Car Damage Detection and Classification Using Deep Transfer Learning," in *Proceeding - IEEE 8th Information Technology International Seminar, ITIS 2022*, 2022, pp. 21–26, doi: 10.1109/ITIS57155.2022.10010292.

[18] A. T. Hermawan, I. A. E. Zaeni, A. P. Wibawa, Gunawan, N. Hartono, and Y. Kristian, "EEG-Based Lie Detection Using Autoencoder Deep Learning with Muse II Brain Sensing," *Int. J. Robot. Control Syst.*, vol. 4, no. 3, pp. 1403–1428, 2024, doi: 10.31763/ijrcs.v4i3.1497.

[19] I. Dagher and D. Barbara, "Facial age estimation using pre-trained CNN and transfer learning," *Multimed. Tools Appl.*, vol. 80, no. 13, pp. 20369–20380, 2021, doi: 10.1007/s11042-021-10739-w.

[20] I. H. Sarker, "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Comput. Sci.*, vol. 2, no. 6, p. 420, 2021, doi: 10.1007/s42979-021-00815-1.

[21] A. Mathew, P. Amudha, and S. Sivakumari, "Deep learning techniques: an overview," *Adv. Intell. Syst. Comput.*, vol. 1141, pp. 599–608, 2021, doi: 10.1007/978-981-15-3383-9_54.

[22] A. Hosna, E. Merry, J. Gyalmo, Z. Alom, Z. Aung, and M. A. Azim, "Transfer learning: a friendly introduction," *Journal of Big Data*, vol. 9, no. 1. Springer, 2022, doi: 10.1186/s40537-022-00652-w.

[23] J. Gupta, S. Pathak, and G. Kumar, "Deep Learning (CNN) and Transfer Learning: A Review," *J. Phys. Conf. Ser.*, vol. 2273, no. 1, 2022, doi: 10.1088/1742-6596/2273/1/012029.

[24] Y. Kristian, L. Zaman, M. Tenoyo, and A. Jodhinata, "Advancing Guitar Chord Recognition: A Visual Method Based on Deep Convolutional Neural Networks and Deep Transfer Learning," *ECTI Transactions on Computer and Information Technology*, vol. 18, no. 2. researchgate.net, pp. 235–249, 2024. [Online]. Available at: https://www.researchgate.net/publication/380532263.

[25] Y. Li, J. Ma, and Y. Zhang, "Image retrieval from remote sensing big data: A survey," *Inf. Fusion*, vol. 67, pp. 94–115, Mar. 2021, doi: 10.1016/J.INFFUS.2020.10.008.

[26] Z. Fathima and S. Shariff, "Product Matching for E-commerce Platform based on Text and Image Similarity using Deep Neural Network Architecture." Dublin, National College of Ireland, pp. 1-22, 2022. [Online]. Available at: https://norma.ncirl.ie/6292/.

[27] X. Zhang, F. Guo, T. Chen, L. Pan, G. Beliakov, and J. Wu, "A Brief Survey of Machine Learning and Deep Learning Techniques for E-Commerce Research," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 18, no. 4. mdpi.com, pp. 2188–2216, 2023, doi: 10.3390/jtaer18040110.

[28] P. Shetty and S. Singh, "Hierarchical Clustering: A Survey," *Int. J. Appl. Res.*, vol. 7, no. 4, pp. 178–181, Apr. 2021, doi: 10.22271/allresearch.2021.v7.i4c.8484.

[29] M. R. Kumar, S. Vishnu, G. Roshen, D. N. Kumar, P. Revathi, and D. R. L. Baster, "Product Recommendation Using Collaborative Filtering and K-Means Clustering," *Proc. - Int. Conf. Comput. Power, Commun. Technol. IC2PCT 2024*, pp. 1722–1728, 2024, doi: 10.1109/IC2PCT60090.2024.10486625.

[30] Suresh, "Shopee Train Images WithLabels Dataset". Retrieved June 24, 2022. [Online]. Available at: https://www.kaggle.com/datasets/dharmiksv/shopee-train-images-withlabels.

[31] L. Massaron, *The Kaggle book : data analysis and machine learning for competitive data science*. Packt Publishing Ltd, p. 534, 2022. [Online]. Available at: https://books.google.co.id/books/about/The_Kaggle_Book.html?id=GAVsEAAAQBAJ&redir_esc=y.

[32] N. Farliana, W. Rahmaningtyas, and R. Widhiastuti, "Development of E-commerce Management and Policy in Indonesia," *Am. J. Humanit. Soc. Sci. Res.*, vol. 06, no. 01, pp. 155–160, 2022. [Online]. Available at: https://www.ajhssr.com/wp-content/uploads/2022/01/N22601155160.pdf.

[33] S. Bamansoor *et al.*, "Efficient online shopping platforms in Southeast Asia," in *2021 2nd International Conference on Smart Computing and Electronic Enterprise: Ubiquitous, Adaptive, and Sustainable Computing Solutions for New Normal, ICSCEE 2021*, 2021, pp. 164–168, doi: 10.1109/ICSCEE50312.2021.9497901.

[34] Z. Wang, L. Li, C. Zeng, S. Dong, and J. Sun, "SLBDetection-Net: Towards closed-set and open-set student learning behavior detection in smart classroom of K-12 education," *Expert Syst. Appl.*, vol. 260, Jan. 2025, doi: 10.1016/J.ESWA.2024.125392.

[35] A. Kapoor, A. Gulli, S. Pal, and F. Chollet, *Deep learning with TensorFlow and Keras*. books.google.com, p. 698, 2022. [Online]. Available at: https://ieeexplore.ieee.org/document/10162595.

[36] Y. M. Pranoto, A. N. Handayani, and Y. Kristian, "Marketplace Product Image Grouping Using Transfer Learning of Deep Convolutional Neural Network in COVID-19 Post-Pandemic Situation," in *The Spirit of Recovery*, CRC Press, 2023, pp. 55–63, doi: 10.1201/9781003331674-4.

[37] P. Desai, J. Pujari, C. Sujatha, A. Kamble, and A. Kambli, "Hybrid Approach for Content-Based Image Retrieval using VGG16 Layered Architecture and SVM: An Application of Deep Learning," *SN Comput. Sci.*, vol. 2, no. 3, 2021, doi: 10.1007/s42979-021-00529-4.

[38] A. A. Elngar, M. Arafa, A. Fathy, B. Moustafa, and O. Mahmoud, "Image Classification Based On CNN: A Survey," *Journal of Cybersecurity and Information Management*. academia.edu, p. PP. 18-50, 2021, doi: 10.54216/jcim.060102.

[39] J. S. Kumar, S. Anuar, and N. H. Hassan, "Transfer Learning based Performance Comparison of the Pre-Trained Deep Neural Networks," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 1. eprints.utm.my, pp. 797–805, 2022, doi: 10.14569/IJACSA.2022.0130193.

[40] N. Abou Baker, N. Zengeler, and U. Handmann, "A Transfer Learning Evaluation of Deep Neural Networks for Image Classification," *Machine Learning and Knowledge Extraction*, vol. 4, no. 1. mdpi.com, pp. 22–41, 2022, doi: 10.3390/make4010002.

[41] C. Öztürk, M. Taşyürek, and M. U. Türkdamar, "Transfer learning and fine-tuned transfer learning methods' effectiveness analyse in the CNN-based deep learning models," *Concurr. Comput. Pract. Exp.*, vol. 35, no. 4, 2023, doi: 10.1002/cpe.7542.

[42] T. Li, A. Rezaeipanah, and E. S. M. Tag El Din, "An ensemble agglomerative hierarchical clustering algorithm based on clusters clustering technique and the novel similarity measurement," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 6, pp. 3828–3842, 2022, doi: 10.1016/j.jksuci.2022.04.010.

[43] X. Ran, Y. Xi, Y. Lu, X. Wang, and Z. Lu, "Comprehensive survey on hierarchical clustering algorithms and the recent developments," *Artif. Intell. Rev.*, vol. 56, no. 8, pp. 8219–8264, 2023, doi: 10.1007/s10462-022-10366-3.

[44] L. Ramos Emmendorfer and A. M. de Paula Canuto, "A generalized average linkage criterion for Hierarchical Agglomerative Clustering," *Appl. Soft Comput.*, vol. 100, 2021, doi: 10.1016/j.asoc.2020.106990.

[45] Y. M. Pranoto, A. N. Handayani, H. W. Herwanto, and Y. Kristian, "Optimizing Product Matching in E-Commerce with DOC2VEC: Leveraging Hierarchical Clustering Parameters Based on Product Titles," *ECTI Trans. Comput. Inf. Technol.*, vol. 18, no. 3, pp. 396–405, 2024, doi: 10.37936/ecti-cit.2024183.256164.

[46] I. K. Salman Al-Tameemi, M. R. Feizi-Derakhshi, S. Pashazadeh, and M. Asadpour, "Multi-Model Fusion Framework Using Deep Learning for Visual-Textual Sentiment Classification," *Comput. Mater. Contin.*, vol. 76, no. 2, pp. 2145–2177, Aug. 2023, doi: 10.32604/CMC.2023.040997.

[47] D. Zhang *et al.*, "Supporting Clustering with Contrastive Learning," *NAACL-HLT 2021 - 2021 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. Proc. Conf.*, pp. 5419–5430, 2021, doi: 10.18653/v1/2021.naacl-main.427.

[48] J. AlShaqsi, W. Wang, O. Drogham, and R. S. Alkhawaldeh, "Quantitative and qualitative similarity measure for data clustering analysis," *Cluster Comput.*, pp. 14977–15002, 2024, doi: 10.1007/s10586-024-04664-4.

[49] H. Gong, Y. Li, J. Zhang, B. Zhang, and X. Wang, "A new filter feature selection algorithm for classification task by ensembling pearson correlation coefficient and mutual information," *Eng. Appl. Artif. Intell.*, vol. 131, p. 107865, 2024, doi: 10.1016/j.engappai.2024.107865.

[50] H. O. Velesaca, G. Bastidas, M. Rouhani, and A. D. Sappa, "Multimodal image registration techniques: a comprehensive survey," *Multimed. Tools Appl.*, vol. 83, no. 23, pp. 63919–63947, 2024, doi: 10.1007/s11042-023-17991-2.

[51] H. Zhou, X. Wang, and Y. Zhang, "Feature selection based on weighted conditional mutual information," *Appl. Comput. Informatics*, vol. 20, no. 1–2, pp. 55–68, 2024, doi: 10.1016/j.aci.2019.12.003.

[52] X. Yang, J. Yan, Y. Cheng, and Y. Zhang, "Learning deep generative clustering via mutual information maximization," *IEEE Trans. Neural Networks Learn. Syst.*, pp. 6263 - 6275, 2022, doi: 10.1109/TNNLS.2021.3135375.

[53] M. Rahmanian and E. G. Mansoori, "An unsupervised gene selection method based on multivariate normalized mutual information of genes," *Chemom. Intell. Lab. Syst.*, vol. 222, p. 104512, 2022, doi: 10.1016/j.chemolab.2022.104512.

[54] I. M. Wiryana, S. Harmanto, A. Fauzi, I. Bil Qisthi, and Z. Nadya Utami, "Store product classification using convolutional neural network," *IAES Int. J. Artif. Intell.*, vol. 12, no. 3, p. 1439, Sep. 2023, doi: 10.11591/ijai.v12.i3.pp1439-1447.