Indonesian Journal of Data and Science

Volume 7 Issue 2
ISSN 2715-9936
https://doi.org/10.56705/ijodas.v7i2.406

Research Article

Resin Code Classification on Plastic Packaging Using Few-Shot Learning

Alyani Noor Septalia[1]; Anindita Septiarini[2] ; Masna Wati[3]
[1] Universitas Mulawarman, Samarinda 75119, Indonesia, alyanins@student.unmul.ac.id
[2] Universitas Mulawarman, Samarinda 75119, Indonesia, anindita@unmul.ac.id
[3] Universitas Mulawarman, Samarinda 75119, Indonesia, masnawati@fkti.unmul.ac.id

Correspondence should be addressed to Anindita Septiarini; anindita@unmul.ac.id

Received 10 March 2026 Accepted 17 July 2026; Published 31 July 2026

Abstract:
Introduction: Accurate sorting of plastic waste using Resin Identification Codes (RICs) is essential for improving recycling quality. However, conventional deep learning approaches generally require large labeled datasets, which are difficult and costly to collect for small RIC symbols on plastic packaging. **Method**: This study employed a Few-Shot Learning approach based on Prototypical Networks using a 7-way 5-shot episodic training configuration. A self-collected dataset of 350 images, comprising 50 images for each of seven RIC categories—PETE, HDPE, PVC, LDPE, PP, PS, and OTHER—was used. Three backbone architectures, ConvNet4, ResNet-18, and EfficientNet-B2, were compared. An ablation study evaluated support-set augmentation, followed by supervised fine-tuning of the selected model. **Results and Discussion**: EfficientNet-B2 achieved the highest episodic accuracy of 93.36%, outperforming ResNet-18 at 88.71% and ConvNet4 at 54.14%. EfficientNet-B2 with light augmentation attained 85.71% accuracy on the fixed 42-image test set, with perfect recall for PVC, LDPE, and PP. Most errors involved visually similar HDPE and PP symbols. Fine-tuning corrected five of six misclassifications, increasing test accuracy to 90.48% and the F1-score from 0.857 to 0.903. **Conclusion**: Prototypical Networks with an EfficientNet-B2 backbone and cosine distance provide an effective solution for RIC classification under limited-data conditions and offer a practical foundation for automated plastic-waste sorting systems.

Keywords: Few-Shot Learning; Prototypical Network; RIC Classification; EfficientNet-B2; Plastic Waste Sorting.

1. Introduction

Plastic waste is one of the biggest environmental problems today. In Indonesia, plastic waste reaches 5.543 million tons every year, with each person using about 26.5 kg of plastic per year, but only 9% of it is successfully recycled even though production reaches 7 million tons per year [1], [2]. Sorting plastic based on the Resin Identification Code (RIC) is very important because mixing different resin types, such as PVC with HDPE or PP with LDPE, can lower the quality of recycled material by 40-60% [3], [4].

Convolutional Neural Networks (CNNs) have shown good results for general plastic waste recognition [5]-[7]. These models work well because of residual connections that make training deep networks easier [8], and compound scaling that balances network depth, width, and resolution [9]. However, most existing studies classify plastic by its shape (like bottles or bags) instead of its RIC symbol, even though the RIC symbol is the actual standard used in the recycling industry [10]. Classifying RIC symbols is difficult because the symbols are small, printing quality varies, packaging can be damaged, and it is expensive to build a large labeled dataset for seven categories that look similar to each other [8], [11].

Few-Shot Learning (FSL) is one way to solve the problem of limited data, since it trains models to learn from only a few labeled examples per class [12], [13]. FSFSL methods are usually grouped into metric-based [14], optimization-

based [15], and model-based approaches [16], [17]. For metric-based methods, previous research shows that the quality of the embedding is the most important factor for FSL performance, and that a good pretrained backbone can matter more than a complex model architecture [18]. Prototypical Networks [14], which is a metric-based FSL method, works by learning one prototype embedding for each class and then classifying new images based on the closest prototype. This method fits well with symbol classification tasks, where the images inside one class can look different from each other but the total amount of data is small [12], [13]. FSL has been used before for logo detection [19], symbols in engineering drawings [20], and document symbols [21], but it has not been used yet for RIC symbols on plastic packaging. Recently, some studies have also used Vision-Language Models for few-shot plastic waste classification [22], which shows that classifying waste with limited data is still an important and open research topic.

This study has three main contributions: (1) it is the first FSL-based model made specifically for classifying RIC symbols using a self-collected close-up image dataset; (2) it compares three backbones (ConvNet4, ResNet-18, and EfficientNet-B2) inside a Prototypical Network for this task; and (3) it includes an ablation study on support-set augmentation and a fine-tuning step to fix classes that are often confused with each other. The main hypothesis is that a pretrained backbone combined with a Prototypical Network can classify RIC categories well using only 50 images per class, and that light augmentation can improve the model's robustness to real packaging conditions.

2. Method

A. Dataset and Preprocessing

A dataset of 350 close-up images was collected using a Samsung Galaxy A54 smartphone (50 MP, f/1.8, OIS) at a fixed distance of 10 cm from the packaging surface. The dataset covers all seven standard RIC categories: PETE (1), HDPE (2), PVC (3), LDPE (4), PP (5), PS (6), and OTHER (7), with 50 images for each class. The images include different conditions, such as different font styles, symbols with or without the polymer name written below, embossed versus printed symbols, and different lighting (**Figure 1**).

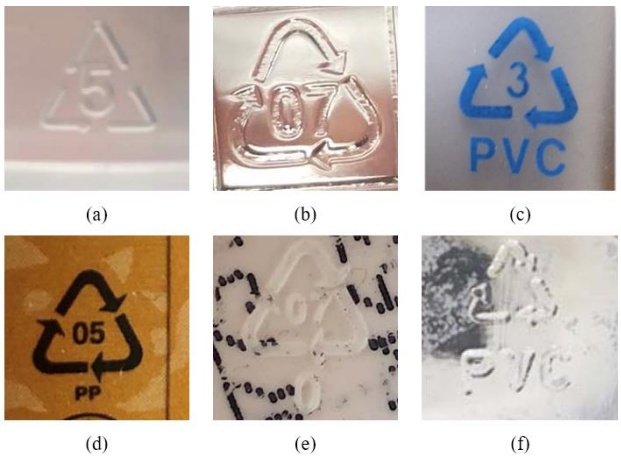

Figure 1. Examples of RIC symbol variations: (a) single digit with no label, (b) double digit with no label, (c) single digit with polymer name, (d) double digit with polymer name, (e) double digit with initials, (f) triangle with no digit.

To avoid data leakage, every image was taken from a different physical packaging item, so no packaging item appears more than once in the dataset or across different subsets. The dataset was split using stratified sampling into 80% training (40 images per class), 8% validation (4 images per class), and 12% test (6 images per class). All images were resized to match the input size needed for each backbone and normalized using the standard ImageNet mean and standard deviation (mean = [0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225]) so they would be compatible with the pretrained weights [23].

Light augmentation was applied only to the training set during episodic sampling, to add more visual variety without changing the shape of the RIC symbols too much. The augmentation steps used were RandomResizedCrop (scale 0.8-1.0), RandomHorizontalFlip, RandomRotation (±12°), and ColorJitter (brightness/contrast/saturation ±15%). No augmentation was applied to the validation or test sets, so the evaluation results stay fair. This approach follows previous work showing that augmenting the support set can help improve prototype diversity and generalization in few-shot image classification [24].

B. Few-Shot Learning Framework

This study uses the Prototypical Network [14] as the main FSL method. Training follows an episodic scheme, where each episode is like a small classification task made by randomly sampling a support set and a query set from the training data. This kind of training helps the model generalize better because it sees many different task combinations during training [25]. In each episode, a class prototype v_c is calculated as the average embedding of the support examples for that class:

$$v_c = \frac{1}{|S^c|} \sum_{(x_k, y_k) \in S^c} f_\phi (x_k) \tag{1}$$

Where f_ϕ (·) is the backbone CNN used as the embedding function. To classify a query image x, the model uses softmax over the cosine distance between the query and every class prototype:

$$P(y = c|x) = \frac{\exp(-d(f_\phi(x), v_c))}{\sum_{c' \epsilon C} \exp(-d(f_\phi(x), v_{c'}))} \tag{2}$$

Cosine distance was chosen instead of Euclidean distance because it is not affected by the size of the embedding vector, which makes the model more stable when there are lighting differences between images. If the embeddings are normalized using L2, cosine distance and squared Euclidean distance actually give similar results with only a constant difference in scale [17], so using cosine distance is still theoretically valid. All experiments used a 7-way 5-shot setup (N = 7 classes, K = 5 support images per class, Q = 15 query images per class per episode). The model was trained using the Adam optimizer with a cosine annealing learning-rate scheduler, and training stopped early based on validation accuracy.

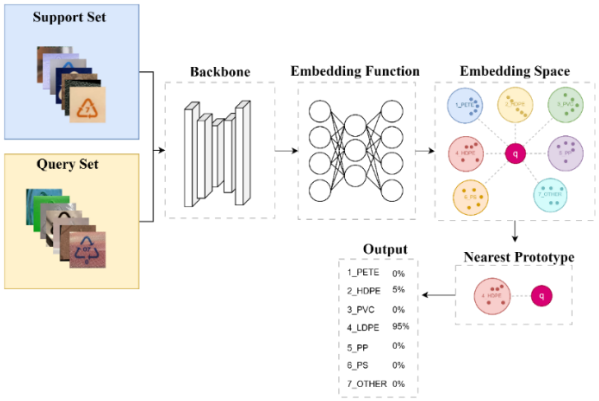

Figure 2. Episodic training scheme: the support set is used to build class prototypes, and query images are classified based on the nearest prototype using cosine distance.

Algorithm 1: Episodic Training with Prototypical Network

Input: Training set D, backbone f_ϕ, episodes T = 600, N = 7 ways, K = 5 shots, Q = 15 queries

Output: Trained embedding function f_ϕ, stored prototypes v_c

1 Initialize f_ϕ with ImageNet pretrained weights

2 **for** t = 1 to T do

Sample N classes $C_t \subset D$ at random
for each class c in C_t do
    Sample K support images → S_c, Q query images → Q_c
    Apply augmentation pipeline to S_c
end for
Compute prototype: $v_c \leftarrow$ mean $(f_\phi(x)$ for x in $S_c)$, for all c
for each query image x in $\cup Q_c$ do
    Compute cosine distance $d(f_\phi(x), v_c)$ or all c
    Compute softmax probability $P(y = c|x)$ via Eq. (2)
end for
Compute cross-entropy loss L over all query predictions
Update f_ϕ via Adam + cosine annealing scheduler
Evaluate on validation set; if no improvement for patience epochs → early stop
end for
Recompute final prototypes v_c from full training set embeddings
Export f_ϕ in TorchScript format for deployment

C. Backbone Architectures

Three CNN backbones were tested as feature extractors inside the Prototypical Network. This choice is based on the idea that embedding quality, which depends mainly on the backbone and its pretraining, is the most important factor in few-shot classification performance [18]. ConvNet4 was used as the baseline model from the original Prototypical Networks paper [14], made of four convolution blocks without residual connections. ResNet-18 was chosen because it has been shown to work well in few-shot classification tasks [8], using residual connections that make it easier to train deeper networks. EfficientNet-B2 was chosen because it has performed well in multi-class waste classification before [26]; its compound scaling method balances network depth, width, and input resolution [9], which is useful for detecting small RIC symbols with many visual variations.

D. Fine-Tuning Protocol

After looking at the results from episodic evaluation, some classes that look similar were still being misclassified, so a fine-tuning step was added using the full 280-image training set. Fine-tuning used cross-entropy loss with the AdamW optimizer (lr = 1e-4, weight decay = 1e-4) to adjust the embeddings so class boundaries become clearer. The test set was kept completely separate during training and fine-tuning, and it was only used at the very end for the final evaluation. Early stopping only checked validation accuracy, so the test set results did not affect model selection in any way. After fine-tuning, the class prototypes were recalculated using all training embeddings, and the model was exported in TorchScript format. Prediction confidence was measured using temperature-scaled softmax (T = 0.02) applied to the cosine similarity between query embeddings and the stored prototypes.

E. Evaluation Metrics

Model performance was measured using accuracy, precision, recall, and F1-score, all calculated from the confusion matrix [27]. For the backbone comparison and the ablation study, episodic accuracy averaged over 600 test episodes was used as the main metric, since this shows how well the model generalizes across many different randomly sampled tasks during model selection. For the final evaluation, precision, recall, and F1-score for each class were calculated on the fixed 42-image test set, where each image is only classified once using a fixed support set, giving a more realistic picture of how the model would perform in real use.

3. Result and Discussion

A. Backbone Comparison

Table I shows the episodic accuracy of the three backbones trained under the same conditions (7-way 5-shot, no augmentation). EfficientNet-B2 got the highest accuracy at 93.36%, much higher than ResNet-18 (88.71%) and ConvNet4 (54.14%). The big difference between ConvNet4 and the pretrained backbones shows that transfer learning from ImageNet is very important when there are only 40 training images per class, which matches earlier findings that pretrained embeddings are the main reason few-shot models perform well [18].

TABLE I. Backbone comparison (7-way 5-shot, no augmentation)

Backbone	Episodic Accuracy (%)
ConvNet4	54.14
ResNet-18	88.71
EfficientNet-B2	**93.36**

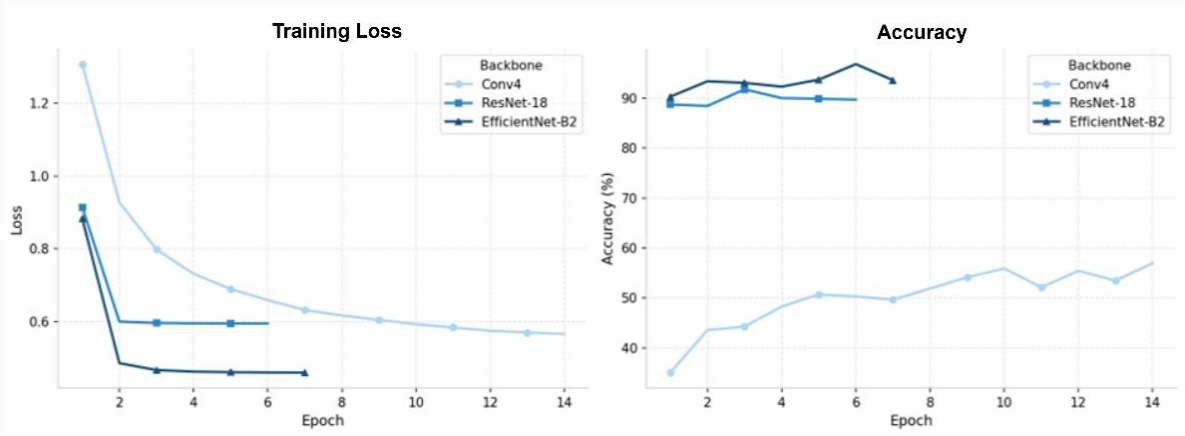

Figure 3. Training progress: (a) episodic accuracy and (b) training loss for ConvNet4, ResNet-18, and EfficientNet-B2.

The training curves in **Figure 3** show that EfficientNet-B2 learned the fastest and reached the highest accuracy overall, while ConvNet4 had an unstable training curve, which matches its simpler design. ResNet-18 trained steadily but stayed about 4.7 percentage points below EfficientNet-B2, which suggests that compound scaling gives a real advantage for learning RIC symbol embeddings when data is limited. This matches other studies showing that richer embeddings usually lead to better few-shot classification results [16], [28].

B. Ablation Study: Data Augmentation

Using EfficientNet-B2 as the fixed backbone, two conditions were compared: training with and without light augmentation on the support set. **Table 2** shows the results.

Table 2. Ablation study: effect of support-set augmentation

Configuration	Episodic Acc. (%)	F1-Score
EfficientNet-B2 (no aug.)	85.25	0.851
EfficientNet-B2 + **aug.**	**87.50**	**0.878**

Adding augmentation improved episodic accuracy by 2.25 percentage points and F1-score by 0.027. This suggests that augmenting the support set makes the prototype embeddings more varied, which helps the model handle changes in lighting and camera angle in the query images [24]. Because it consistently improved the results, the augmented EfficientNet-B2 setup was chosen as the final model.

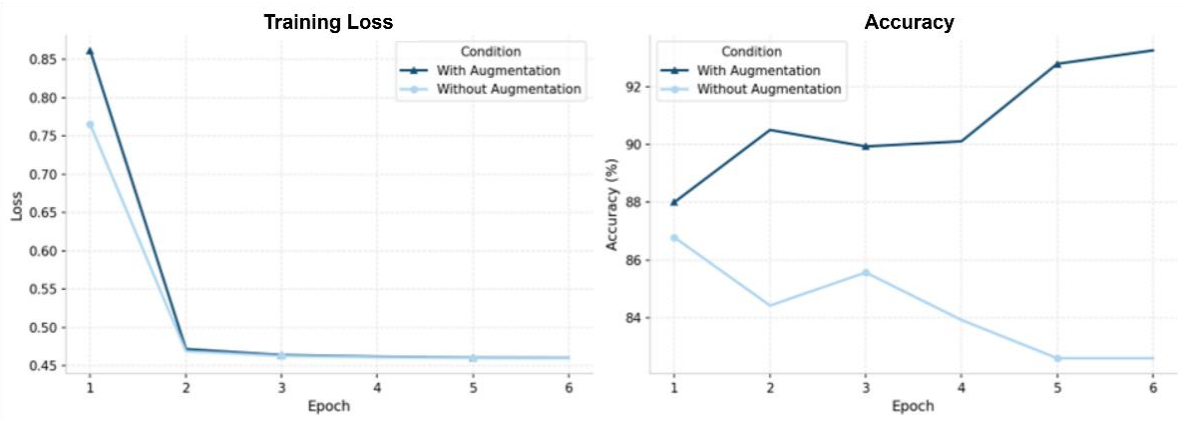

Figure 4. Training progress in the ablation study: the augmented model reaches higher accuracy.

C. Final Model Evaluation

Sections 3A and 3B report episodic accuracy averaged over 600 test episodes, which reflects how well the model generalizes across many different sampled tasks during model selection. This section instead uses the fixed 42-image test set to measure real deployment performance. These two evaluations measure different things: episodic accuracy shows meta-learning generalization, while fixed test accuracy shows how the model performs in a single, realistic classification pass. The final model (EfficientNet-B2 with augmentation) was evaluated on the fixed 42-image test set. **Table 3** shows the per-class results.

Table 3. Per-class classification ceport (final model, before fine-tuning)

Class	Precision	Recall	F1-Score	Support
1_PETE	1.000	0.833	0.909	6
2_HDPE	1.000	0.500	0.667	6
3_PVC	1.000	1.000	1.000	6
4_LDPE	0.750	1.000	0.857	6
5_PP	0.600	1.000	0.750	6
6_PS	1.000	0.833	0.909	6
7_OTHER	1.000	0.833	0.909	6
Accuracy			**0.857**	**42**

Overall test accuracy reached 85.71%. Because the test set is small (42 images, 6 per class), one wrong prediction changes the accuracy by about 2.38%. Using bootstrap resampling (n = 1,000), the 95% confidence interval for the accuracy is [73.81%, 95.24%], which shows that even with a small test set, the result is still reasonably reliable. Three classes reached perfect recall (PVC, LDPE, and PP), meaning no images from these classes were missed, but LDPE and PP had lower precision because some images from other classes were wrongly predicted as LDPE or PP. Most errors came from the HDPE-PP pair, three out of six HDPE images were wrongly predicted as PP, with high confidence (92.65%, 99.87%, and 99.92%). This is likely because the numbers "2" and "5" look similar when printed or embossed on certain packaging materials. One PETE image was predicted as LDPE, one PS image as PP, and one OTHER image as LDPE, which explains the rest of the errors.

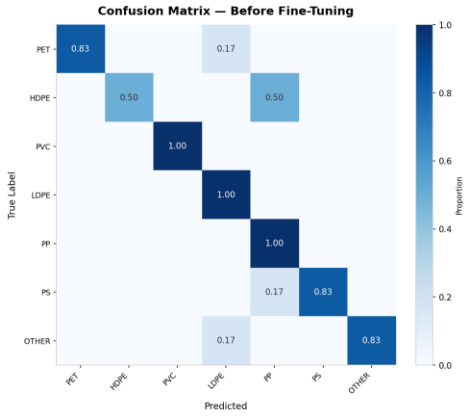

Figure 5. Confusion matrix (rows = actual class, columns = predicted class). Most errors come from HDPE being predicted as PP.

D. Fine-Tuning Results

Fine-tuning was done to fix the HDPE-PP confusion. The backbone was trained further on all 280 training images using cross-entropy loss and AdamW (lr = 1e-4, weight decay = 1e-4), with early stopping based on validation accuracy. Table 4 compares the F1-scores before and after fine-tuning.

Table 4. F1-Score Before and After Fine-Tuning

Class	F1-Score Before	F1-Score After	ΔF1
1_PETE	0.909	0.909	0.000
2_HDPE	0.667	0.857	+0.190
3_PVC	1.000	0.923	−0.077
4_LDPE	0.857	0.800	−0.057
5_PP	0.750	0.909	+0.159
6_PS	0.909	1.000	+0.091
7_OTHER	0.909	0.923	+0.014
Weighted avg	0.857	0.903	+0.046

Fine-tuning raised HDPE recall from 0.500 to 1.000 and improved the weighted F1-score from 0.857 to 0.903, showing that normal supervised fine-tuning can fix embedding overlap between similar classes. The drop in F1 for PVC (-0.077) and LDPE (-0.057) shows a trade-off between precision and recall, fine-tuning improved LDPE precision from 0.750 to 1.000 by removing false positives, but recall dropped from 1.000 to 0.667 because some LDPE images were now classified into neighboring classes. This trade-off is acceptable since the main goal was to fix the bigger HDPE-PP problem.

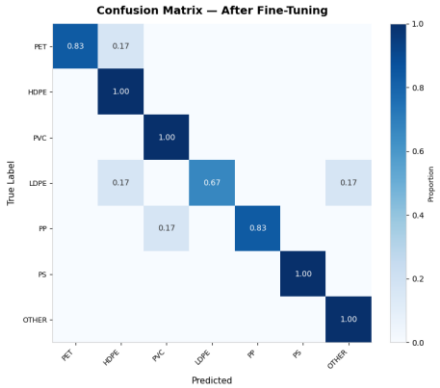

Figure 6. Confusion matrix after fine-tuning: HDPE and PP are no longer confused with each other.

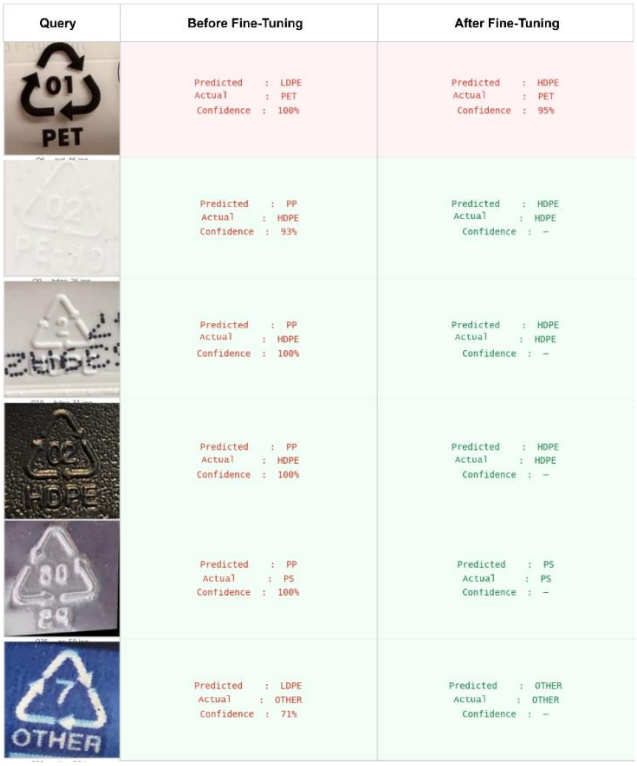

Figure 7. Prediction confidence for an HDPE image: before fine-tuning (PP: 99.92%) vs. after (HDPE: 100%).

E. Discussion

These results show that Prototypical Networks with cosine distance work well for classifying RIC symbols even when data is limited. EfficientNet-B2's compound scaling [9] gave it a clear advantage in embedding quality over the other backbones, which matches its good performance in other waste classification studies [26] and supports the idea that embedding quality is the main factor for few-shot performance [18]. The ablation study shows that even simple augmentation (±12° rotation, ±15% color jitter) can improve results without changing the shape of the RIC symbols too much, which agrees with other feature-augmentation studies in FSL [24].

Compared to earlier RIC classification research, this study reaches competitive accuracy using only 50 images per class, while Listyalina et al. [10] needed 685 images with DenseNet-121 to reach 85.51%. Agarwal et al. reached 99.79% using one-shot learning on the WaDaBa dataset, but that dataset was collected in a controlled lab setting, while our dataset represents more realistic packaging conditions, so the two results are not directly comparable. Ranjbar et al. [22] recently used Vision-Language Models for few-shot plastic waste classification with multimodal prompting; even though their method is very different from episodic metric learning, both studies show that classifying waste with limited data is still a real and active research problem.

The ongoing HDPE-PP confusion shows a real challenge, the numbers "2" and "5" can look almost the same on embossed packaging when viewed up close. Future work could try text-aware features, such as combining OCR results with visual features as done in [29], to help solve this. Making Prototypical Networks work well across different domains is also still an open research problem [30], which shows why more diverse datasets are needed for future testing. The one case that stayed unresolved, a PETE image predicted as LDPE with 99.99% confidence, is likely caused by very similar embeddings between the two classes, and might need extra PETE samples with different printing styles or hard-negative training in the future.

4. Conclusion

This study shows that a Prototypical Network using an EfficientNet-B2 backbone and cosine distance can reach 85.71% test accuracy for classifying seven RIC classes using only 50 training images per class, improving to 90.3% weighted F1-score after fine-tuning. EfficientNet-B2's compound scaling made it clearly better than the ResNet-18 and ConvNet4 baselines within this FSL setup, which supports the idea that embedding quality is the main factor driving few-shot performance [18]. Support-set augmentation consistently helped the model generalize better, and fine-tuning successfully fixed most of the HDPE-PP confusion.

This study also has some limitations. The 350 images were all taken with one device (Samsung Galaxy A54) under fairly controlled lighting and a fixed 10 cm distance, so the model's performance might change with different cameras or more varied real-world conditions. The small test set (42 images, 6 per class) also adds some uncertainty to the results, shown by the 95% bootstrap confidence interval of [73.81%, 95.24%]. In addition, all packaging samples came from one geographic area, and printing styles can be different across manufacturers and countries. Testing how well Prototypical Networks generalize across different domains is still an open question [30]. Future work should use more devices, more packaging sources, and more varied environments to check if the model still performs well.

Overall, this study presents a working FSL pipeline for automatic RIC-based plastic waste sorting that only needs a small amount of labeled data, making it practical for recycling facilities that do not have many labeled samples. Future work could expand the dataset with more real-world images, try combining visual and text features as shown in recent VLM-based studies [22], and test the model on larger FSL benchmark datasets such as Meta-Dataset [17] to check generalizability further.

Acknowledgment

The authors thank the Informatics Study Program, Faculty of Engineering, Universitas Mulawarman, for institutional support.

References

[1] Sustainable Waste Indonesia, “Collection and Recycling Rate Index of Plastic Waste in Indonesia,” 2025.

[2] OECD, *Regional Plastics Outlook for Southeast and East Asia*. OECD Publishing, 2025.

[3] J. Vaucher, A. Demongeot, V. Michaud, and Y. Leterrier, “Recycling of Bottle Grade PET: Influence of HDPE Contamination on the Microstructure and Mechanical Performance of 3D Printed Parts,” *Polymers*, vol. 14, no. 24, p. 5507, Dec. 2022, https://doi.org/10.3390/polym14245507.

[4] T. Werner, I. Taha, and D. Aschenbrenner, “An overview of polymer identification techniques in recycling plants with focus on current and future challenges,” *Procedia CIRP*, vol. 120, pp. 1381-1386, 2023, https://doi.org/10.1016/j.procir.2023.09.180.

[5] J. Bobulski and M. Kubanek, “Deep Learning for Plastic Waste Classification System,” *Applied Computational Intelligence and Soft Computing*, vol. 2021, pp. 1-7, May 2021, https://doi.org/10.1155/2021/6626948.

[6] A. A. P. Chazhoor, E. S. L. Ho, B. Gao, and W. L. Woo, “Deep transfer learning benchmark for plastic waste classification,” *Intelligence & Robotics*, 2022, https://doi.org/10.20517/ir.2021.15.

[7] E. Gothai, R. Thamilselvan, P. Natesan, M. Keerthivasan, K. Kabinesh, and D. K. Ruban, “Plastic Waste Classification Using CNN for Supporting 3R’s Principle,” in *2022 International Conference on Computer Communication and Informatics (ICCCI)*, Jan. 2022, pp. 01-07, https://doi.org/10.1109/ICCCI54379.2022.9740902.

[8] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770-778, https://doi.org/10.1109/CVPR.2016.90.

[9] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks."

[10] L. Listyalina, Y. Yudianingsih, A. W. Soedjono, E. L. Utari, and D. A. Dharmawan, "Deep-RIC: Plastic Waste Classification using Deep Learning and Resin Identification Codes (RIC)," *Telematika*, vol. 19, no. 2, p. 215, June 2022, https://doi.org/10.31315/telematika.v19i2.7419.

[11] S. Agarwal, R. Gudi, and P. Saxena, "Image Classification Approaches for Segregation of Plastic Waste Based on Resin Identification Code," *Transactions of the Indian National Academy of Engineering*, vol. 7, no. 3, pp. 739-751, Sept. 2022, https://doi.org/10.1007/s41403-022-00324-4.

[12] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a Few Examples: A Survey on Few-shot Learning," *ACM Computing Surveys*, vol. 53, no. 3, pp. 1-34, May 2021, https://doi.org/10.1145/3386252.

[13] Y. Song, T. Wang, P. Cai, S. K. Mondal, and J. P. Sahoo, "A Comprehensive Survey of Few-shot Learning: Evolution, Applications, Challenges, and Opportunities," *ACM Computing Surveys*, vol. 55, no. 13s, pp. 1-40, Dec. 2023, https://doi.org/10.1145/3582688.

[14] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical Networks for Few-shot Learning." arXiv, June 19, 2017, Accessed: Feb. 15, 2024, http://arxiv.org/abs/1703.05175.

[15] C. Finn, P. Abbeel, and S. Levine, "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks." arXiv, July 18, 2017, https://doi.org/10.48550/arXiv.1703.03400.

[16] J. Y. Lim, K. M. Lim, C. P. Lee, and Y. X. Tan, "A review of few-shot image classification: Approaches, datasets and research trends," *Neurocomputing*, vol. 649, p. 130774, Oct. 2025, https://doi.org/10.1016/j.neucom.2025.130774.

[17] E. Triantafillou *et al.*, "Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples." arXiv, Apr. 08, 2020, https://doi.org/10.48550/arXiv.1903.03096.

[18] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, and P. Isola, "Rethinking Few-Shot Image Classification: A Good Embedding is All You Need?," in *Computer Vision - ECCV 2020*, vol. 12359, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer International Publishing, 2020, pp. 266-282.

[19] S. Hou, W. Liu, A. Karim, Z. Jia, W. Jia, and Y. Zheng, "Few-shot logo detection," *IET Computer Vision*, vol. 17, no. 5, pp. 586-598, Aug. 2023, https://doi.org/10.1049/cvi2.12205.

[20] L. Jamieson, E. Elyan, and C. F. Moreno-García, "Few-Shot Symbol Detection in Engineering Drawings," *Applied Artificial Intelligence*, vol. 38, no. 1, p. 2406712, Dec. 2024, https://doi.org/10.1080/08839514.2024.2406712.

[21] M. Alfaro-Contreras, A. Ríos-Vila, J. J. Valero-Mas, and J. Calvo-Zaragoza, "Few-shot symbol classification via self-supervised learning and nearest neighbor," *Pattern Recognition Letters*, vol. 167, pp. 1-8, Mar. 2023, https://doi.org/10.1016/j.patrec.2023.01.014.

[22] I. Ranjbar, Y. Ventikos, and M. Arashpour, "Zero-shot and few-shot multimodal plastic waste classification with vision-language models," *Waste Management*, vol. 202, p. 114815, July 2025, https://doi.org/10.1016/j.wasman.2025.114815.

[23] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211-252, Dec. 2015, https://doi.org/10.1007/s11263-015-0816-y.

[24] H. Wang, S. Tian, Y. Fu, J. Zhou, J. Liu, and D. Chen, "Feature augmentation based on information fusion rectification for few-shot image classification," *Scientific Reports*, vol. 13, no. 1, p. 3607, Mar. 2023, https://doi.org/10.1038/s41598-023-30398-1.

[25] P. Zhu, Z. Zhu, Y. Wang, J. Zhang, and S. Zhao, "Multi-granularity episodic contrastive learning for few-shot learning," *Pattern Recognition*, vol. 131, p. 108820, Nov. 2022, https://doi.org/10.1016/j.patcog.2022.108820.

[26] S. Agustiani, A. Junaidi, R. Aryanti, and A. A. B. Kamil, "Waste Classification using EfficientNetB3-Based Deep Learning for Supporting Sustainable Waste Management," vol. 4, no. 1, 2025.

[27] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168-192, Jan. 2021, https://doi.org/10.1016/j.aci.2018.08.003.

[28] X. Li, X. Yang, Z. Ma, and J.-H. Xue, "Deep metric learning for few-shot image classification: A Review of recent developments," *Pattern Recognition*, vol. 138, p. 109381, June 2023, https://doi.org/10.1016/j.patcog.2023.109381.

[29] V. O. Kuzevanov and D. V. Tikhomirova, "Deep Learning for Effective Visualization and Classification of Recyclable Material Labels," *Scientific Visualization*, vol. 16, no. 5, pp. 179-196, Dec. 2024, https://doi.org/10.26583/sv.16.5.12.

[30] W. Li, L. Wang, J. Xu, J. Huo, Y. Gao, and J. Luo, "Revisiting Local Descriptor Based Image-To-Class Measure for Few-Shot Learning," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 7253-7260, https://doi.org/10.1109/CVPR.2019.00743.