



A Transfer Learning Framework for Automated Egg Damage Classification Using Pretrained EfficientNet

Suyahman Suyahman¹, Ardy Wicaksono², Irvan Setiadi Kartawiria³, Kukuh Madyaningrana⁴

^{1,2}Department of Computer Science, Universitas Sugeng Hartono, Sukoharjo, Indonesia

³Department of Food Technology, Universitas Sugeng Hartono, Sukoharjo, Indonesia

⁴Department of Nutrition, Universitas Sugeng Hartono, Sukoharjo, Indonesia

*suyahman.com@gmail.com

Abstract

Egg damage detection is important for maintaining food safety and operational efficiency in poultry production. Manual inspection is often inconsistent and difficult to scale, creating a need for accurate and computationally efficient automated methods. This study aims to develop and evaluate a transfer learning framework for automated egg damage classification using pretrained EfficientNet-B0. Three configurations were compared, namely Frozen Backbone Standard CE, Full Fine-Tuning Standard CE, and Full Fine-Tuning Weighted CE. The dataset was divided using a stratified 70:15:15 split, and the models were trained with AdamW, differential learning rates, early stopping, and image augmentation. Performance was assessed using accuracy, balanced accuracy, macro precision, macro recall, macro F1-score, weighted F1-score, confusion matrices, inference time, and Gradient-weighted Class Activation Mapping. The results show that Full Fine-Tuning Standard CE achieved the best overall performance, with an accuracy of 0.9917, balanced accuracy of 0.9947, macro F1-score of 0.9876, and weighted F1-score of 0.9917. The model correctly classified all non-damaged eggs and misclassified only one damaged egg, while maintaining an inference time of 29.53 ms per image. Weighted cross-entropy did not improve overall performance and increased false-negative predictions. Grad-CAM analysis showed that the model generally focused on relevant shell defects, although several errors were associated with background and contextual features. These findings demonstrate that full fine-tuning of pretrained EfficientNet-B0 provides an accurate, efficient, and interpretable solution for automated egg damage classification.

Keywords: computer vision; EfficientNet; egg damage classification; food quality inspection; transfer learning

How to Cite: [J. Suyahman, A. Wicaksono, I. Setiadi Kartawiria, and K. Madyaningrana, "A Transfer Learning Framework for Automated Egg Damage Classification Using Pretrained EfficientNet", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 10, no. 5, pp. 1504–1513, Oct. 2026, doi: [10.29207/resti.v10i5.7673](https://doi.org/10.29207/resti.v10i5.7673).

Permalink/DOI: <https://doi.org/10.29207/resti.v10i5.7673>

Received: March 28, 2026

Accepted: September 5, 2026

Available Online: October 16, 2026

This is an open-access article under the [CC BY 4.0 License](https://creativecommons.org/licenses/by/4.0/)

Published by [Ikatan Ahli Informatika Indonesia](https://www.iaii.or.id/)

1. Introduction

Eggs are an important source of animal protein and are consumed extensively across the world. For this reason, preserving egg quality is essential not only for food safety but also for the economic sustainability of poultry production systems [1-4]. Among the defects that commonly reduce egg quality, eggshell cracking is one of the most significant. Cracked eggs are more vulnerable to microbial contamination and tend to deteriorate rapidly during storage and transportation [5], [6]. Therefore, early identification of eggshell defects is necessary to prevent unsafe products from entering the market and to minimize financial losses throughout the supply chain.

In industrial settings, egg quality inspection is generally conducted through manual visual examination or candling techniques. Candling applies light to identify internal abnormalities or cracks in the eggshell [7]. However, the effectiveness of these methods relies strongly on the experience, attention, and consistency of the operator, making them prone to human error and subjective judgment [8]. In high-throughput production environments, where thousands of eggs may need to be inspected every hour, manual inspection is difficult to standardize and becomes increasingly inefficient. Beyond manual methods, industrial egg sorting systems also employ technologies such as acoustic analysis, spectroscopy, and machine vision. Although these technologies support high-speed inspection, their

effectiveness often depends on specialized hardware and predefined algorithms. Moreover, many existing systems have only been evaluated under laboratory conditions, which limits their robustness and applicability in real production environments. These constraints have encouraged the development of automated inspection systems based on computer vision and machine learning [9], [10].

Recent progress in deep learning has substantially enhanced the performance of automated visual inspection systems. Convolutional neural networks (CNNs) have been widely used for food quality evaluation and agricultural inspection applications [11-14]. Botta et al., for example, introduced a CNN-based method for eggshell crack detection using image patches and reported an accuracy of 95.38% [15]. Although the method achieved strong predictive performance, the use of image patches requires additional preprocessing steps and increases computational cost, which may reduce its suitability for real-time implementation.

Object detection approaches have also been investigated for eggshell crack identification. Caballero et al. compared several object detection models, including YOLOv4, YOLOv7, Faster R-CNN, SSD, and RetinaNet, for detecting cracks in duck eggs [16]. The study found that YOLOv7 produced the highest performance, with a mean average precision of 0.792. While object detection models provide accurate defect localization, they generally require detailed annotation procedures and greater computational resources than conventional classification models.

More recent studies have attempted to address variations across datasets and operational environments. Cheng et al. proposed a multi-domain training framework based on NSF-E-MMD to learn domain-invariant features from multiple datasets [17]. Their framework improved detection performance on previously unseen domains, obtaining mAP values of 86.6% and 88.8% with YOLOv5 and YOLOv8, respectively. Nevertheless, this method introduces additional complexity because it requires multi-domain datasets and domain alignment procedures during model development.

Despite these improvements, many existing methods still depend on complex model architectures, extensive preprocessing, detailed annotations, or domain adaptation mechanisms. These requirements may restrict their implementation in practical production environments. In applications where the main objective is limited to binary classification between damaged and non-damaged eggs, a more straightforward and computationally efficient method is preferable.

To address this limitation, this study proposes a transfer learning framework for automated egg damage classification using a pretrained EfficientNet model [18]. Transfer learning allows knowledge obtained from large-scale datasets to be adapted to a specific

classification task, even when the available task-specific dataset is limited [19]. EfficientNet was selected because it offers strong classification capability while maintaining parameter efficiency [20]. The proposed framework is intended to provide an accurate, lightweight, and practical solution for integration into real-world egg inspection systems.

2. Methods

This study proposes a transfer learning framework for automated egg damage classification. The pipeline includes dataset preparation, preprocessing and data splitting, model training, performance evaluation, and Grad-CAM interpretation. The complete workflow is shown in Figure 1.

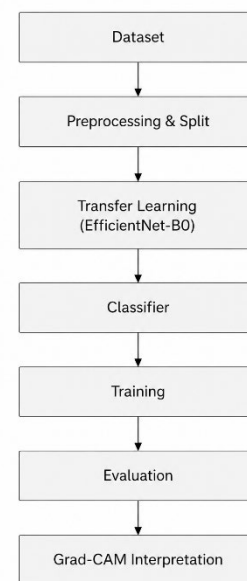


Figure 1. Process of the proposed framework

2.1 Hardware and Computational Environment

The experiments in this study were conducted using a cloud-based computational environment with the hardware specifications summarized in Table 1. The system is equipped with dual NVIDIA Tesla T4 GPUs, each providing 15 GB of memory, enabling efficient training and inference of deep learning models. In addition, a multi-core CPU and 30 GB of RAM support data preprocessing and model execution, while 57.6 GB of storage is utilized for dataset handling and model artifacts.

This configuration ensures sufficient computational resources to perform model training and evaluation effectively while maintaining a balance between performance and resource efficiency.

Table 1. Hardware Specifications

Component	Specification
GPU	NVIDIA Tesla T4 (2 units)
GPU Memory	15 GB per GPU
CPU	Multi-core CPU
RAM	30 GB
Storage	57.6 GB

2.2 Dataset

The dataset was obtained from Kaggle and contains egg images labeled as damaged or non-damaged (<https://www.kaggle.com/datasets/abdullahkhanuet22/eggs-images-classification-damaged-or-not>). The images vary in orientation, lighting, and crack visibility. Damaged eggs show visible cracks or shell defects, while non-damaged eggs remain intact. Representative samples are shown in Figure 2.

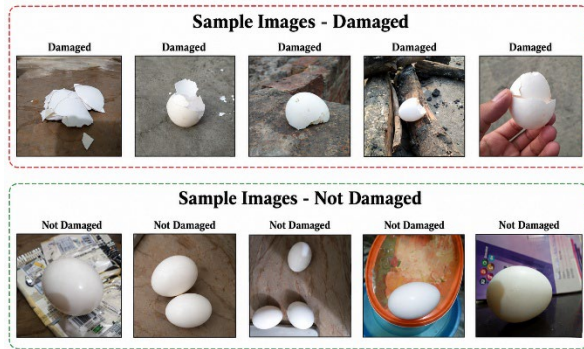


Figure 2. Samples from the dataset

Figure 3 shows an imbalanced class distribution, with more damaged than non-damaged egg images. The dataset was split using stratified sampling into 70% training, 15% validation, and 15% testing sets. A fixed random seed of 1234 ensured reproducibility, and the same partitions were used for all models. The training set was used for optimization, the validation set for model selection and early stopping, and the test set for final evaluation.

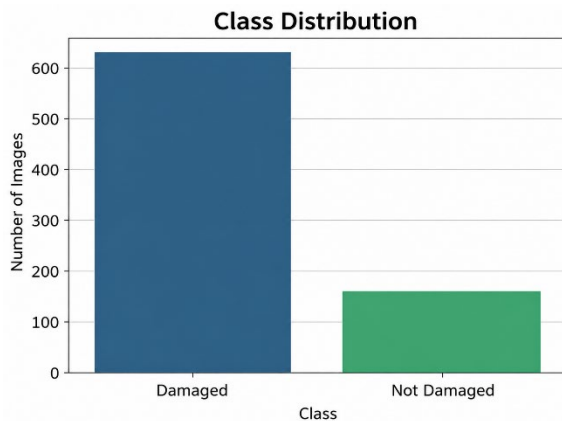


Figure 3. Class distribution

2.3 Preprocessing

All images underwent standardized preprocessing to ensure compatibility with the pretrained EfficientNet model [21]. First, the images were resized to the required input resolution to maintain uniform spatial dimensions. Second, the pixel values were normalized using the standard parameters associated with EfficientNet pretraining to align the input distribution with the ImageNet dataset [22], [23]. Third, the images were converted to RGB format and transformed into tensor representations. These steps reduced variability

caused by heterogeneous acquisition conditions and ensured consistent model inputs.

2.4 Transfer Learning Framework

EfficientNet-B0 is used as the backbone of the transfer learning framework for automated egg damage classification. The architecture provides strong feature extraction with relatively low computational complexity through compound scaling of network depth, width, and input resolution [24], [25]. Pretraining on ImageNet supplies general visual representations, including edges, textures, contours, and object shapes, which are subsequently adapted to distinguish damaged and non-damaged eggs.

The original classifier is replaced with a dropout layer of 0.4 followed by a fully connected layer with two outputs. Three configurations are evaluated, namely Frozen Backbone Standard CE, Full Fine-Tuning Standard CE, and Full Fine-Tuning Weighted CE. In the frozen-backbone configuration, only the classifier is trained. In both full fine-tuning configurations, the entire backbone and classifier are optimized. The overall architecture is illustrated in Figure 4.

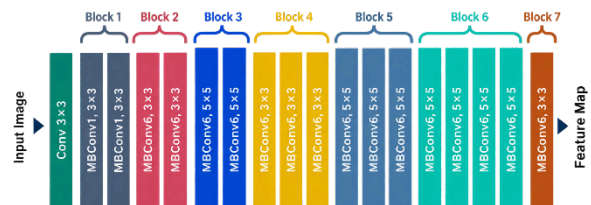


Figure 4. EfficientNet Architecture

Table 2 summarizes the differences among the three configurations.

Frozen Backbone Standard CE serves as the baseline and uses standard cross-entropy loss. Full Fine-Tuning Standard CE applies standard cross-entropy while updating all network layers. Full Fine-Tuning Weighted CE uses the same full fine-tuning procedure but replaces the standard loss with class-weighted cross-entropy. The class weights are calculated from the training subset using inverse class frequency to increase the contribution of the underrepresented class during optimization.

For both full fine-tuning configurations, differential learning rates are applied, with 1×10^{-5} assigned to the pretrained backbone and 3×10^{-4} assigned to the classifier. All configurations use the AdamW optimizer, a weight decay of 1×10^{-4} , a batch size of 32, and a maximum of 30 epochs. Early stopping based on validation loss is applied to retain the model with the best validation performance. This experimental design isolates the effects of backbone adaptation and class-weighted optimization while maintaining identical architecture and training conditions.

2.5 Evaluation Metrics

Model performance is evaluated using accuracy, balanced accuracy, macro precision, macro recall,

macro F1-score, weighted F1-score, and inference time. Balanced accuracy is used as the primary metric because it calculates the average recall across classes, allowing damaged and non-damaged eggs to contribute equally despite the imbalanced class distribution.

A confusion matrix is used to summarize true-positive, true-negative, false-positive, and false-negative predictions [26-28]. This analysis helps identify class-

specific errors and systematic misclassification patterns.

Macro-averaged precision, recall, and F1-score assign equal importance to both classes, while weighted F1-score accounts for the number of samples in each class [29]. Accuracy measures overall prediction correctness, and inference time per image is reported to assess the computational suitability of each configuration for practical deployment.

Table 2. Comparison of Experimental Configurations

Aspect	Frozen Backbone Standard CE	Full Fine-Tuning Standard CE	Full Fine-Tuning Weighted CE
Pretrained Model	EfficientNet-B0	EfficientNet-B0	EfficientNet-B0
Backbone	Frozen	Fully trainable	Fully trainable
Trainable Layers	Classifier only	Backbone and classifier	Backbone and classifier
Classifier	Dropout (0.4) and fully connected layer	Dropout (0.4) and fully connected layer	Dropout (0.4) and fully connected layer
Loss Function	Standard CrossEntropyLoss	Standard CrossEntropyLoss	Weighted CrossEntropyLoss
Class Weighting	Not applied	Not applied	Inverse-frequency weights from training data
Optimizer	AdamW	AdamW	AdamW
Backbone Learning Rate	Not applicable	1×10^{-5}	1×10^{-5}
Classifier Learning Rate	3×10^{-4}	3×10^{-4}	3×10^{-4}
Weight Decay	1×10^{-4}	1×10^{-4}	1×10^{-4}
Batch Size	32	32	32
Maximum Epochs	30	30	30

2.6 Grad-CAM

Grad-CAM was applied to improve the interpretability of the image-level classification results. The method generates class-discriminative heatmaps that highlight image regions contributing most strongly to a prediction. Gradients from the predicted class were calculated with respect to the final convolutional feature maps of EfficientNet, and the resulting importance weights were used to produce an activation map.

The heatmaps were normalized, resized to the original image dimensions, and overlaid on the corresponding images. Representative true-positive, true-negative, false-positive, and false-negative samples from the testing subset were analyzed. This analysis was used to determine whether the model focused on relevant eggshell regions, such as visible cracks and structural defects, or on irrelevant patterns such as shadows, reflections, and background edges.

3. Results and Discussions

This study evaluates three EfficientNet-B0 transfer learning configurations for automated egg damage classification. The evaluated configurations are Frozen Backbone Standard CE, Full Fine-Tuning Standard CE, and Full Fine-Tuning Weighted CE. Their performance is examined through learning curves, balanced accuracy, confusion matrices, quantitative metrics, inference time, and Grad-CAM-based visual interpretation.

The learning patterns of the three evaluated configurations are shown in Figure 5. Figures 5(a) and 5(b) present the results of Frozen Backbone Standard CE. The training loss decreases from approximately

0.53 to 0.17, while the validation loss falls from about 0.41 to 0.17. The small gap between the two curves indicates stable optimization with limited overfitting. However, the final loss values remain higher than those obtained by the full fine-tuning configurations.

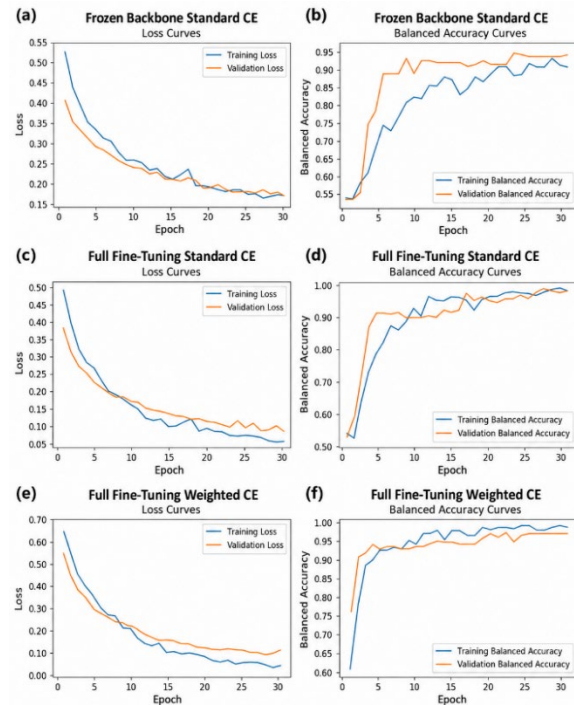


Figure 5. Learning behavior of the three EfficientNet-B0 configurations

This pattern is also reflected in the balanced accuracy curve shown in Figure 5(b). The frozen-backbone model improves gradually throughout training, with

validation balanced accuracy increasing rapidly during the early epochs before stabilizing between approximately 0.91 and 0.94. Training balanced accuracy reaches about 0.90 near the final epochs. The validation curve is often higher than the training curve because data augmentation is applied only to the training subset, making the augmented training images more challenging than the deterministic validation images. These results show that the frozen ImageNet backbone already provides useful general visual features. However, restricting optimization to the classification head limits the model's ability to learn eggshell-specific damage patterns.

The effect of adapting the entire network becomes more evident in Figures 5(c) and 5(d), which present the learning behavior of Full Fine-Tuning Standard CE. The training loss declines from approximately 0.51 to below 0.04, whereas the validation loss decreases from around 0.38 to nearly 0.07. Both curves show a consistent downward trend, and the validation loss does not exhibit a sustained increase. This behavior indicates effective optimization without severe overfitting.

A similar improvement is observed in the balanced accuracy curves in Figure 5(d). Validation balanced accuracy exceeds 0.90 within approximately five epochs, followed shortly by training balanced accuracy. Both curves then remain between approximately 0.95 and 0.99. Their close alignment suggests that full fine-tuning enables the pretrained EfficientNet-B0 backbone to adapt effectively to shell cracks, broken contours, shell openings, and other damage-related structures.

To examine whether class weighting could further improve performance, Figures 5(e) and 5(f) present the learning behavior of Full Fine-Tuning Weighted CE. The training loss decreases from approximately 0.63 to 0.05, while the validation loss falls from around 0.55 to approximately 0.11. Although both curves indicate stable convergence, the final validation loss remains higher than that of Full Fine-Tuning Standard CE. The gap between the training and validation curves also becomes more visible during the later epochs.

This result is consistent with the balanced accuracy shown in Figure 5(f). Training balanced accuracy approaches 0.99, whereas validation balanced accuracy stabilizes between approximately 0.95 and 0.98. These findings suggest that weighted optimization increases class sensitivity but does not improve generalization beyond standard cross-entropy. Overall, Figure 5 shows that full fine-tuning produces stronger learning behavior than the frozen-backbone baseline. Among the three configurations, Full Fine-Tuning Standard CE achieves the best combination of low validation loss, high balanced accuracy, and stable convergence.

The differences observed during training are further supported by the confusion matrices presented in Figure 6. Figure 6(a) shows that Frozen Backbone Standard CE correctly classifies 92 damaged eggs and 22 non-damaged eggs. Three damaged eggs are misclassified as non-damaged, while three non-damaged eggs are predicted as damaged, resulting in six total errors. This finding confirms that pretrained EfficientNet-B0 features provide a strong baseline, although the fixed backbone remains insufficiently adapted to subtle eggshell damage patterns.

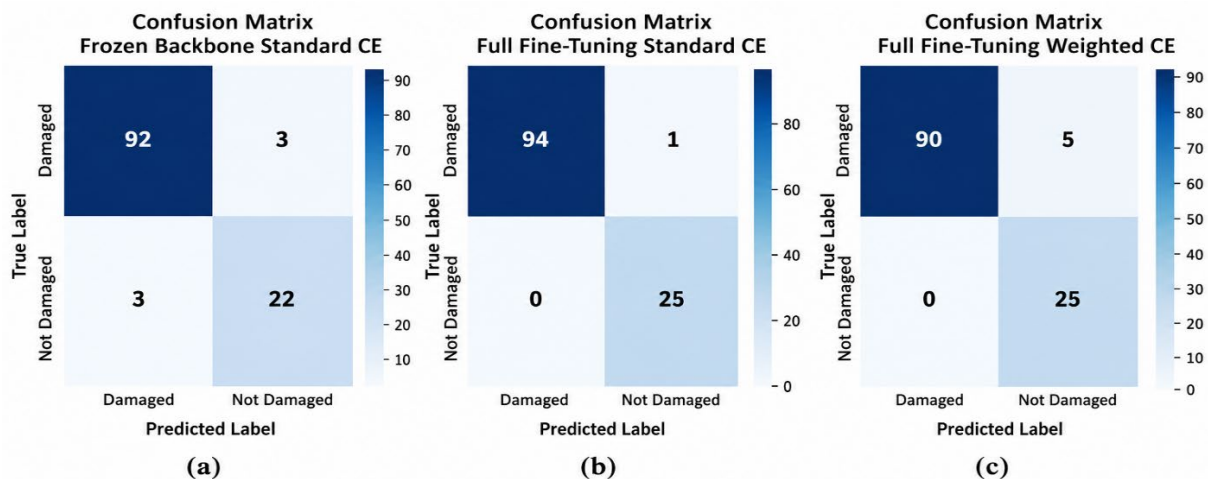


Figure 6. Confusion matrices of the three EfficientNet-B0 configurations on the testing subset

A clear improvement appears in Figure 6(b), which presents the confusion matrix of Full Fine-Tuning Standard CE. The model correctly classifies 94 of 95 damaged eggs and all 25 non-damaged eggs. Only one damaged egg is predicted as non-damaged, while no non-damaged egg is classified as damaged. When damaged eggs are treated as the positive class, this

single error is considered a false negative. This configuration therefore produces the strongest classification result among the three experiments.

In comparison, Figure 6(c) shows that Full Fine-Tuning Weighted CE correctly classifies 90 damaged eggs and all 25 non-damaged eggs. However, five damaged eggs are misclassified as non-damaged. Although this

configuration eliminates false positives, it produces more false negatives than Full Fine-Tuning Standard CE.

This error pattern is particularly important in practical egg inspection. A damaged egg classified as non-damaged may pose a greater food safety risk than an intact egg that is incorrectly rejected. Therefore, the higher number of false negatives produced by Full Fine-Tuning Weighted CE makes this configuration less

suitable for deployment than Full Fine-Tuning Standard CE, despite its relatively high balanced accuracy.

The quantitative performance of the three configurations is summarized in Table 3. Frozen Backbone Standard CE achieves an accuracy of 0.9500 and a balanced accuracy of 0.9242. The corresponding macro precision, macro recall, and macro F1-score are all 0.9242. These values confirm that the frozen backbone already performs well, although further task-specific adaptation is still required.

Table 3. Performance Comparison of the EfficientNet-B0 Configurations

Model Configuration	Accuracy	Balanced Accuracy	Macro Precision	Macro Recall	Macro F1-score	Weighted F1-score	Inference Time (ms/image)
Frozen Backbone Standard CE	0.95	0.9242	0.9242	0.9242	0.9242	0.95	31.05
Full Fine-Tuning Standard CE	0.9917	0.9947	0.9808	0.9947	0.9876	0.9917	29.53
Full Fine-Tuning Weighted CE	0.9583	0.9737	0.9167	0.9737	0.941	0.9597	29.27

Full Fine-Tuning Standard CE produces the best overall result. It achieves an accuracy of 0.9917, balanced accuracy of 0.9947, macro precision of 0.9808, macro recall of 0.9947, macro F1-score of 0.9876, and weighted F1-score of 0.9917. Compared with Frozen Backbone Standard CE, this configuration improves accuracy by 4.17 percentage points, balanced accuracy by 7.05 percentage points, and macro F1-score by 6.34 percentage points.

These improvements indicate that updating the entire EfficientNet-B0 backbone enables the network to learn more discriminative and task-specific representations. The model becomes more responsive to shell cracks, openings, broken surfaces, and structural irregularities. The balanced accuracy is slightly higher than the overall accuracy because recall is very strong across both classes, as also reflected in Figure 6(b).

Full Fine-Tuning Weighted CE achieves an accuracy of 0.9583, balanced accuracy of 0.9737, and macro F1-score of 0.9410. Although this configuration improves balanced accuracy relative to the frozen-backbone baseline, it does not outperform standard full fine-tuning. Its macro recall remains high at 0.9737, but macro precision decreases to 0.9167.

This difference shows that class weighting changes the class-specific decision boundary but does not improve the overall balance between precision and recall. In the current dataset, the damaged class contains more samples than the non-damaged class. Inverse-frequency weighting therefore gives greater importance to the non-damaged class. As a result, the model becomes more conservative when predicting damage and produces more damaged eggs classified as non-damaged.

The inference times of the three configurations are comparable. They range from 29.27 to 31.05 ms per image. Full Fine-Tuning Weighted CE records the lowest inference time, although the difference from the other configurations is less than two milliseconds. Since

all three models use the same EfficientNet-B0 architecture during inference, these minor differences are more likely caused by runtime variation than by structural differences.

Full Fine-Tuning Standard CE therefore provides the strongest balance between predictive performance and computational efficiency. It achieves the highest accuracy, balanced accuracy, and F1-based metrics while maintaining an inference time of approximately 29.53 ms per image.

Grad-CAM is used to examine the image regions that contribute to model predictions. The visual explanations are presented in Figure 7. Figure 7(a) shows a correctly classified damaged egg with a confidence score of 0.9990. The strongest activation is concentrated around the visible shell defect. This indicates that the model uses a meaningful damage-related region when producing its prediction.

Figure 7(d) presents another damaged egg incorrectly classified as non-damaged with a confidence score of 0.7418. The activation is directed mainly toward the background, hand, and object contour rather than the damaged shell. This result shows that the model may rely on contextual or spurious visual features when the image composition is complex.

Because the true class in Figures 7(c) and 7(d) is damaged and the predicted class is non-damaged, both examples are false negatives. These visual results complement the confusion matrix analysis and show that the remaining errors are associated with insufficient attention to the most relevant damage regions.

Figure 7(b) presents a correctly classified non-damaged example with a confidence score of 0.8835. In this case, the activation is distributed more broadly around the eggs and surrounding background. This suggests that the model may recognize non-damaged eggs partly through the absence of localized crack-related evidence and partly through broader contextual patterns.

Figure 7(c) shows a damaged egg incorrectly classified as non-damaged with a confidence score of 0.5409. The Grad-CAM heatmap is broadly distributed around the egg and background, while the damaged shell region receives relatively weak activation. This indicates that the model fails to treat the broken shell as the dominant discriminative feature.

Overall, the Grad-CAM analysis confirms that the model learns meaningful defect-related representations when damage is clear and visually distinct. However, the false-negative examples reveal sensitivity to background, illumination, hand position, egg orientation, and scene composition. These findings support the use of egg segmentation, object-centered cropping, background normalization, and more diverse environmental augmentation in future work.

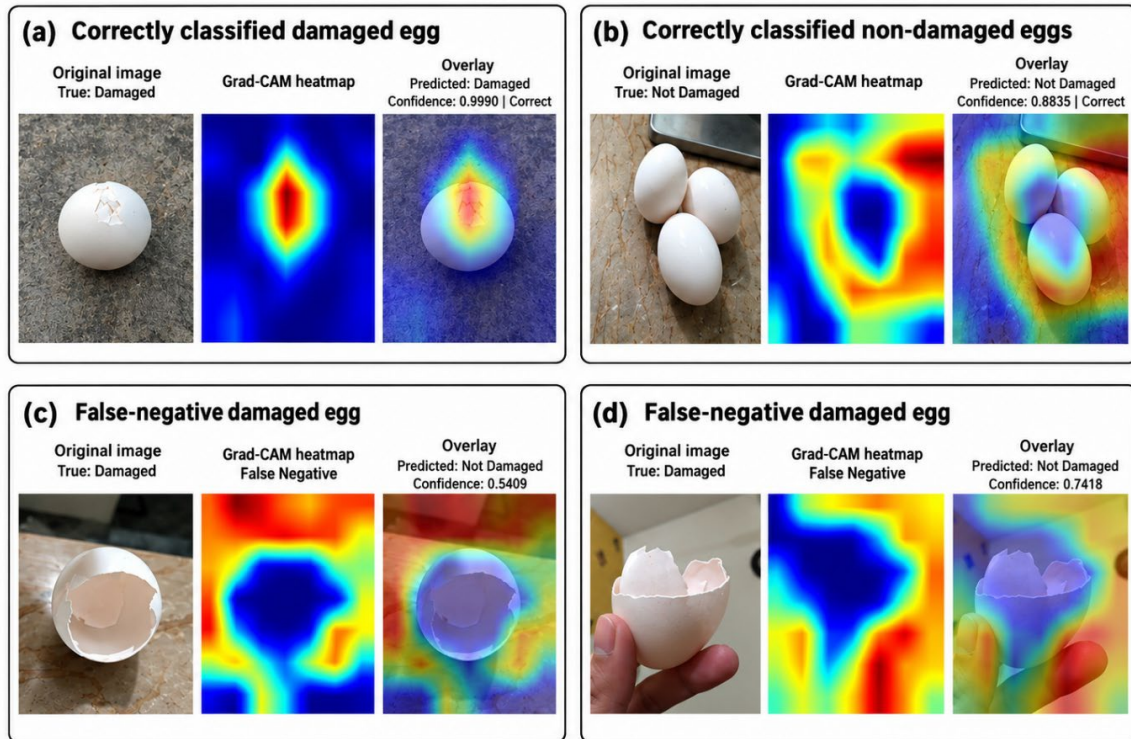


Figure 7. Grad-CAM visualizations of representative classification outcomes.

Table 4 compares the three configurations proposed in this study with previous egg damage detection methods. The comparison includes the applied method, task type, key technique, reported performance, and availability of Grad-CAM or other visual explanation methods.

Table 4 compares the proposed EfficientNet-B0 framework with previous studies on egg crack and damage detection. Botta et al. employed a CNN-based

classification approach with patch extraction and reported an accuracy of 95.38%. This result is comparable to the 95.00% accuracy obtained by the Frozen Backbone Standard CE configuration in the present study. However, the earlier approach requires an additional patch extraction stage, whereas the proposed framework performs direct image-level classification.

Table 4. Comparison with Previous Egg Damage Detection Studies

Ref	Method	Task Type	Key Technique	Performance	Grad-CAM or Visual Explanation
[15]	CNN-based crack detection	Classification	Patch extraction with CNN	Accuracy 95.38%	Not reported
[16]	YOLOv4, YOLOv7, Faster R-CNN, SSD, and RetinaNet	Object detection	CLAHE preprocessing and multi-model comparison	Best result obtained by YOLOv7 with mAP 0.792	Bounding-box localization; Grad-CAM not reported
[17]	YOLOv5 and YOLOv8 with NSFE-MMD	Object detection	Multi-domain training and domain adaptation	YOLOv5 mAP 86.6% and YOLOv8 mAP 88.8%	Detection localization; Grad-CAM not reported
Proposed Study	Pretrained EfficientNet-B0	Classification	Comparative evaluation of Frozen Backbone Standard CE, Full Fine-Tuning Standard CE, and Full Fine-Tuning Weighted CE	Accuracy of 95.00%, 99.17%, and 95.83%, respectively; Full Fine-Tuning Standard CE achieved the best result	Grad-CAM applied to representative correct and false-negative predictions

Caballero et al. evaluated several object detection architectures and reported the best performance using YOLOv7 with an mAP of 0.792. Their approach provides explicit defect localization through bounding boxes, but it requires spatial annotations and a more complex training pipeline. Cheng et al. also used object detection models and incorporated NSF-MMD to improve robustness across different domains. Their method achieved mAP values of 86.6% with YOLOv5 and 88.8% with YOLOv8. Although this approach improves cross-domain generalization, it depends on multi-domain datasets and additional domain adaptation procedures.

The proposed study evaluates three transfer learning configurations within a single EfficientNet-B0 framework. Frozen Backbone Standard CE achieved an accuracy of 95.00%, Full Fine-Tuning Standard CE achieved 99.17%, and Full Fine-Tuning Weighted CE achieved 95.83%. The results show that full fine-tuning with standard cross-entropy provides the strongest performance, indicating that adapting the complete pretrained backbone is more effective than training only the classifier or applying inverse-frequency class weighting.

Direct numerical comparison between the proposed classification framework and object detection studies should be interpreted cautiously because accuracy and mAP evaluate different tasks. Accuracy measures image-level classification correctness, whereas mAP evaluates both localization and classification. Therefore, the comparison primarily highlights differences in annotation requirements, methodological complexity, and interpretability rather than direct numerical superiority.

The previous classification study did not report Grad-CAM-based interpretation. The object detection studies provide spatial localization through bounding boxes, but Grad-CAM was not reported. In contrast, the proposed study applies Grad-CAM to representative correct and false-negative predictions. This additional analysis provides qualitative evidence of whether the model focuses on relevant eggshell regions or is influenced by background and contextual features.

Several limitations should be acknowledged when interpreting the findings. First, the experiments rely on a single publicly available dataset, which may not capture the full variability of industrial egg inspection environments. Differences in shell color, illumination, camera position, background conditions, and conveyor systems could affect model performance. In addition, the testing subset contains only 120 images, making external validation on larger and more diverse datasets necessary before broader deployment.

The reported results are also based on one fixed stratified data split. Although this split preserves the class distribution, repeated experiments or stratified cross-validation would provide a more reliable estimate of performance stability and variability. Grad-CAM

improves model interpretability by highlighting influential image regions, but its explanations remain qualitative and spatially coarse. Therefore, it should not be considered a substitute for precise localization methods based on object detection or segmentation.

Future research should focus on expanding the dataset, conducting repeated validation, and evaluating the model under more realistic operational conditions. Object-centered preprocessing, background removal, egg segmentation, and targeted augmentation may help reduce contextual bias and improve robustness. Real-time deployment testing should also be conducted, while lightweight detection or segmentation models may be explored when exact crack localization is required.

4. Conclusions

This study developed and evaluated a transfer learning framework for automated egg damage classification using pretrained EfficientNet-B0. Three configurations were compared, including Frozen Backbone Standard CE, Full Fine-Tuning Standard CE, and Full Fine-Tuning Weighted CE, to examine the effects of backbone adaptation and imbalance-aware optimization. The results show that Full Fine-Tuning Standard CE provides the strongest overall performance, achieving the highest accuracy, balanced accuracy, macro F1-score, and weighted F1-score while maintaining low inference latency. The confusion matrix further confirms that this configuration produces only one false-negative prediction and no false positives, demonstrating strong reliability across both damaged and non-damaged classes. In contrast, the frozen-backbone configuration offers a competitive baseline but remains limited by its fixed feature representations, while weighted cross-entropy improves sensitivity to the non-damaged class at the cost of additional damaged eggs being misclassified as intact. These findings indicate that full adaptation of the pretrained EfficientNet-B0 backbone is more effective than training only the classifier or applying inverse-frequency weighting for the present dataset. Grad-CAM analysis also provides qualitative evidence that the model can focus on relevant shell defects when damage is visually clear and spatially distinct. However, several incorrect predictions reveal residual sensitivity to background, illumination, object orientation, and contextual features. The original contribution of this work lies in presenting a unified comparison of frozen-backbone learning, standard full fine-tuning, and weighted full fine-tuning within a lightweight EfficientNet-B0 framework, supported by post hoc visual interpretation. From a practical perspective, the framework offers a simple image-level classification solution that avoids bounding-box annotation while retaining strong predictive performance and near-real-time inference capability. This makes it potentially suitable for automated screening in poultry production and food quality inspection systems. Future work should validate the framework using larger and more

diverse datasets, repeated experimental runs, and real industrial acquisition conditions. Additional object-centered preprocessing, segmentation, and lightweight localization methods may further improve robustness and support precise defect identification.

Declarations

AI Usage Statement

The authors used Google Gemini, ChatGPT, and QuillBot solely to improve the readability and language of the manuscript; all scientific content, analysis, and conclusions are the authors' own.

Author Contributions

S.S.: supervision, conceptualization, methodology, software, formal analysis, writing – original draft; A.W.: data curation, validation, writing – review and editing; I.S.K.: investigation, resources, writing – review and editing; K.M.: formal analysis, writing – review and editing.

Funding Statement

This research received no external funding.

Conflict of Interest Statement

The authors declare no conflict of interest.

Data and Code Availability

The dataset used in this study is publicly available at <https://www.kaggle.com/datasets/abdullahkhanuet22/eggs-images-classification-damaged-or-not>.

References

- [1] S. Molnár and L. Szöllösi, "Sustainability and Quality Aspects of Different Table Egg Production Systems: A Literature Review," *Sustainability*, vol. 12, no. 19, p. 7884, Sep. 2020, doi: [10.3390/su12197884](https://doi.org/10.3390/su12197884)
- [2] F. S. Nassar and A. O. Abbas, "Crisis and risk management in poultry production: Preparing future leaders for sustainability and food security," *Poultry Science*, vol. 104, no. 12, p. 105903, Dec. 2025, doi: [10.1016/j.psj.2025.105903](https://doi.org/10.1016/j.psj.2025.105903)
- [3] M. S. Istiak and A. Khaliduzzaman, "Poultry and Egg Production: An Overview," in *Informatics in Poultry Production*, A. Khaliduzzaman, Ed., Singapore: Springer Nature Singapore, 2022, pp. 3–12. doi: [10.1007/978-981-19-2556-6_1](https://doi.org/10.1007/978-981-19-2556-6_1)
- [4] F. S. Nassar, "Strategic role of poultry production sciences in shaping the future of global food security and strengthen sustainability," *Poultry Science*, vol. 105, no. 5, p. 106617, May 2026, doi: [10.1016/j.psj.2026.106617](https://doi.org/10.1016/j.psj.2026.106617)
- [5] A. Damena, A. Mikru, M. Adane, and B. Dobo, "Microbial Profile and Safety of Chicken Eggs from a Poultry Farm and Small-Scale Vendors in Hawassa, Southern Ethiopia," *Journal of Food Quality*, vol. 2022, pp. 1–16, May 2022, doi: [10.1155/2022/7483253](https://doi.org/10.1155/2022/7483253)
- [6] Y. Madhumala and V. S. Soraganvi, "Microbiology of Food Spoilage," in *Frontiers in Food Biotechnology*, J. S. Yaradoddi, B. S. Meti, S. B. Mudgulkar, and D. Agsar, Eds., Singapore: Springer Nature Singapore, 2024, pp. 67–80. doi: [10.1007/978-981-97-3261-6_5](https://doi.org/10.1007/978-981-97-3261-6_5)
- [7] P. Yu, C. Shen, J. Cheng, X. Yin, C. Liu, and Z. Yu, "Advances in Nondestructive Technologies for External Eggshell Quality Evaluation," *Sensors*, vol. 25, no. 18, p. 5796, Sep. 2025, doi: [10.3390/s25185796](https://doi.org/10.3390/s25185796)
- [8] M. W. Ahmed, S. J. Hossainy, A. Khaliduzzaman, J. L. Emmert, and M. Kamruzzaman, "Non-destructive optical sensing technologies for advancing the egg industry toward Industry 4.0: A review," *Comp Rev Food Sci Food Safe*, vol. 22, no. 6, pp. 4378–4403, Nov. 2023, doi: [10.1111/1541-4337.13227](https://doi.org/10.1111/1541-4337.13227)
- [9] X. Yang, R. B. Bist, S. Subedi, and L. Chai, "A Computer Vision-Based Automatic System for Egg Grading and Defect Detection," *Animals*, vol. 13, no. 14, p. 2354, Jul. 2023, doi: [10.3390/ani13142354](https://doi.org/10.3390/ani13142354)
- [10] Z. Wu, H. Zhang, and C. Fang, "Research on machine vision online monitoring system for egg production and quality in cage environment," *Poultry Science*, vol. 104, no. 1, p. 104552, Jan. 2025, doi: [10.1016/j.psj.2024.104552](https://doi.org/10.1016/j.psj.2024.104552)
- [11] H. Ding, H. Hou, L. Wang, X. Cui, W. Yu, and D. I. Wilson, "Application of Convolutional Neural Networks and Recurrent Neural Networks in Food Safety," *Foods*, vol. 14, no. 2, p. 247, Jan. 2025, doi: [10.3390/foods14020247](https://doi.org/10.3390/foods14020247)
- [12] X. Wang et al., "Enhancing optical non-destructive methods for food quality and safety assessments with machine learning techniques: A survey," *Journal of Agriculture and Food Research*, vol. 19, p. 101734, Mar. 2025, doi: [10.1016/j.jafr.2025.101734](https://doi.org/10.1016/j.jafr.2025.101734)
- [13] C. Wang et al., "Application of Convolutional Neural Network-Based Detection Methods in Fresh Fruit Production: A Comprehensive Review," *Front. Plant Sci.*, vol. 13, p. 868745, May 2022, doi: [10.3389/fpls.2022.868745](https://doi.org/10.3389/fpls.2022.868745)
- [14] S. Xu, H. Wang, X. Liang, and H. Lu, "Research Progress on Methods for Improving the Stability of Non-Destructive Testing of Agricultural Product Quality," *Foods*, vol. 13, no. 23, p. 3917, Dec. 2024, doi: [10.3390/foods13233917](https://doi.org/10.3390/foods13233917)
- [15] B. Botta, S. S. R. Gattam, and A. K. Datta, "Eggshell crack detection using deep convolutional neural networks," *Journal of Food Engineering*, vol. 315, p. 110798, Feb. 2022, doi: [10.1016/j.jfoodeng.2021.110798](https://doi.org/10.1016/j.jfoodeng.2021.110798)
- [16] R. G. Caballero, J. R. Gucor, R. D. P. Talledo, J. J. M. Montemayor, and M. M. Maulana, "Crack Detection on Duck Eggs Using CNN," *Procedia Computer Science*, vol. 257, pp. 631–636, 2025, doi: [10.1016/j.procs.2025.03.081](https://doi.org/10.1016/j.procs.2025.03.081)
- [17] Y. Cheng, Y. Huang, J. Zhang, X. Zhang, Q. Wang, and W. Fan, "Robust Detection of Cracked Eggs Using a Multi-Domain Training Method for Practical Egg Production," *Foods*, vol. 13, no. 15, p. 2313, Jul. 2024, doi: [10.3390/foods13152313](https://doi.org/10.3390/foods13152313)
- [18] Md. A. Talukder, Md. A. Layek, M. Kazi, Md. A. Uddin, and S. Aryal, "Empowering COVID-19 detection: Optimizing performance through fine-tuned EfficientNet deep learning architecture," *Computers in Biology and Medicine*, vol. 168, p. 107789, Jan. 2024, doi: [10.1016/j.combiomed.2023.107789](https://doi.org/10.1016/j.combiomed.2023.107789)
- [19] J. Gupta, S. Pathak, and G. Kumar, "Deep Learning (CNN) and Transfer Learning: A Review," *J. Phys.: Conf. Ser.*, vol. 2273, no. 1, p. 012029, May 2022, doi: [10.1088/1742-6596/2273/1/012029](https://doi.org/10.1088/1742-6596/2273/1/012029)
- [20] A. Shamila Ebenezer, S. Deepa Kanmani, M. Sivakumar, and S. Jeba Priya, "Effect of image transformation on EfficientNet model for COVID-19 CT image classification," *Materials Today: Proceedings*, vol. 51, pp. 2512–2519, 2022, doi: [10.1016/j.matpr.2021.12.121](https://doi.org/10.1016/j.matpr.2021.12.121)
- [21] A. Rafay and W. Hussain, "EfficientSkinDis: An EfficientNet-based classification model for a large manually curated dataset of 31 skin diseases," *Biomedical Signal Processing and Control*, vol. 85, p. 104869, Aug. 2023, doi: [10.1016/j.bspc.2023.104869](https://doi.org/10.1016/j.bspc.2023.104869)
- [22] Mst. A. Orin, A. H. Shakil, M. Rahman, and Z. I. A. Khalib, "Preprocessing-enhanced EfficientNetB7-based Computer-aided Diagnosis (CAD) Framework for Skin Lesion Classification," in *2025 IEEE 7th International Conference on Sustainable Technologies For Industry 5.0 (STI)*, Dhaka, Bangladesh: IEEE, Dec. 2025, pp. 1–6. doi: [10.1109/STI69347.2025.11367584](https://doi.org/10.1109/STI69347.2025.11367584)
- [23] M. Salvi, F. Branciforti, F. Molinari, and K. M. Meiburger, "Generative models for color normalization in digital pathology and dermatology: Advancing the learning paradigm," *Expert Systems with Applications*, vol. 245, p. 123105, Jul. 2024, doi: [10.1016/j.eswa.2023.123105](https://doi.org/10.1016/j.eswa.2023.123105)
- [24] R. Chaganti, V. Ravi, and T. D. Pham, "Image-based malware representation approach with EfficientNet convolutional neural networks for effective malware classification," *Journal*

- of Information Security and Applications, vol. 69, p. 103306, Sep. 2022, doi: [10.1016/j.jisa.2022.103306](https://doi.org/10.1016/j.jisa.2022.103306)
- [25] M. Behzadpour, B. L. Ortiz, E. Azizi, and K. Wu, "Breast Tumor Classification Using EfficientNet Deep Learning Model," 2024, arXiv. doi: [10.48550/ARXIV.2411.17870](https://doi.org/10.48550/ARXIV.2411.17870)
- [26] T. A. Cengel, B. Gencturk, E. T. Yasin, M. B. Yildiz, I. Cinar, and M. Koklu, "Automating egg damage detection for improved quality control in the food industry using deep learning," Journal of Food Science, vol. 90, no. 1, p. e17553, Jan. 2025, doi: [10.1111/1750-3841.17553](https://doi.org/10.1111/1750-3841.17553)
- [27] Suyahman, O. Saut Parulian, D. Prasetyo, and M. Anwar Fauzi, "Traditional Batik Pattern Recognition with MobileNetV2 and Sampling-Based Hyperparameter Optimization," Jurnal Ilmu Komputer dan Informasi, vol. 19, no. 1, pp. 53–61, Mar. 2026, doi: [10.21609/jiki.v19i1.1597](https://doi.org/10.21609/jiki.v19i1.1597)
- [28] S. Suyahman and A. Hapsari, "VGG-Based Feature Extraction for Classifying Traditional Batik Motifs Using Machine Learning Models," Preservation, Digital Technology & Culture, vol. 54, no. 3, pp. 215–222, Oct. 2025, doi: [10.1515/pdte-2025-0009](https://doi.org/10.1515/pdte-2025-0009)
- [29] A. S. Rathor et al., "Deep convolutional network for plant monitoring in precision agriculture," Sustainable Computing: Informatics and Systems, vol. 50, p. 101302, Jun. 2026, doi: [10.1016/j.suscom.2026.101302](https://doi.org/10.1016/j.suscom.2026.101302)