# JAST (Journal of Animal Science and Technology) TITLE PAGE

**Upload this completed form to website with submission**

| ARTICLE INFORMATION | Fill in information in each box below |
|---|---|
| **Article Type** | Research article |
| **Article Title (within 20 words without abbreviations)** | Egg fertility assessment system using deep learning technology |
| **Running Title (within 10 words)** | Egg fertility assessment system using deep learning technology |
| **Author** | Jin-Hyun Park[1†], See Hwan Sohn[2†], Hee-Mun Park[3] and Sang-Hyon Oh[4*] |
| **Affiliation** | [1]School of Mechatronics Engineering, Engineering College of Convergence Technology, Gyeongsang National University, Jinju 52725, Korea<br>[2]HenTech, Jinju 52725, Korea<br>[3]Department of Automation Machine Maintenance, Suncheon Campus of Korea Polytechnics, Suncheon 57975, Korea<br>[4]Division of Animal Science, College of Agriculture and Life Science, Gyeongsang National University, Jinju 52828, Korea |
| **ORCID (for more information, please visit https://orcid.org)** | Jin-Hyun Park https://orcid.org/0000-0002-7966-0014<br>Sea Hwan Sohn https://orcid.org/0000-0001-6735-9761<br>Hee-Mun Park https://orcid.org/0000-0001-5182-1739<br>Sang-Hyon Oh https://orcid.org/0000-0002-9696-9638 |
| **Competing interests** | No potential conflict of interest relevant to this article was reported. |
| **Funding sources**<br>State funding sources (grants, funding sources, equipment, and supplies). Include name and number of grant if available. | Not applicable. |
| **Acknowledgements** | Not applicable. |
| **Availability of data and material** | Upon reasonable request, the datasets of this study can be available from the corresponding author. |
| **Authors' contributions**<br>Please specify the authors' role using this form. | Conceptualization: Jin-Hyun Park, See Hwan Sohn, Sang-Hyon Oh<br>Data curation: Jin-Hyun Park, See Hwan Sohn, Hee-Mun Park, Sang-Hyon Oh<br>Formal analysis: Jin-Hyun Park, See Hwan Sohn, Hee-Mun Park, Sang-Hyon Oh<br>Methodology: Jin-Hyun Park, See Hwan Sohn, Hee-Mun Park, Sang-Hyon Oh<br>Software: Jin-Hyun Park, Hee-Mun Park, Sang-Hyon Oh<br>Validation: Jin-Hyun Park, See Hwan Sohn, Sang-Hyon Oh<br>Investigation: Jin-Hyun Park, See Hwan Sohn, Hee-Mun Park, Sang-Hyon Oh<br>Writing - original draft: Jin-Hyun Park, See Hwan Sohn, Sang-Hyon Oh<br>Writing - review & editing: Jin-Hyun Park, See Hwan Sohn, Sang-Hyon Oh |
| **Ethics approval and consent to participate** | All animal procedures such as ethical and animal welfare issues were approved by the Institutional Animal Care and Use Committee of Gyeongsang National University (approval number: 2018-7). |

**CORRESPONDING AUTHOR CONTACT INFORMATION**


| For the corresponding author (responsible for correspondence, proofreading, and reprints) | Fill in information in each box below |
|---|---|
| First name, middle initial, last name | Prof. Sang-Hyon OH |
| Email address – this is where your proofs will be sent | shoh@gnu.ac.kr |
| Secondary Email address | |
| Address | Division of Animal Science, Gyeongsang National University, Jinju 52828, Korea |
| Cell phone number | |
| Office phone number | +82-55-772-1955 |
| Fax number | +82-55- |

**Egg fertility assessment system using deep learning technology**


## Abstract

In the poultry industry, accurate distinction between fertilized, unfertilized, and hatching failure eggs is a key factor for improving hatching efficiency and reducing resource waste. This study comprehensively evaluated 12 CNN architectures for non-destructive egg fertility classification based on candling-style images captured on days 5–6 of incubation, under four experimental scenarios: Case A (image-level random split, three classes), Case B (egg-level separation, three classes), Case C (image-level random split, two classes), and Case D (egg-level separation, two classes). In Cases A and B, overall accuracy ranged from 93.65%–98.65% and 90.82%–95.79%, respectively. Per-class recall for fertilized eggs exceeded 98% across all models, while hatching failure egg recall was highly variable (5.88%–76.47% in Case A; 0%–27.78% in Case B), reflecting severe class imbalance and visual ambiguity. In Cases C and D (fertilized vs. unfertilized only), overall accuracy improved to 97.02%–99.80% and 95.84%–98.81%, respectively, with unfertilized egg recall ranging from 74.47%–97.87% (Case C) and 58.33%–87.50% (Case D). Egg-level evaluation (Cases B and D) consistently yielded lower accuracy than image-level splitting (Cases A and C), confirming that image-level splitting overestimates real-world generalization performance. CNN selection should be guided by the balance between computational resources and required accuracy: lightweight models (SqueezeNet: 4.7 MB; ShuffleNet: 5.5 MB) suit resource-constrained deployments, while larger models (Inception-v3, VGG-16, VGG-19) offer superior accuracy for high-performance applications. Classification of hatching failure eggs remains challenging due to their visual similarity to both fertilized and unfertilized eggs depending on the timing of developmental arrest, and represents an important direction for future research.

**Keywords** Egg, Fertility, Assessment, Deep Learning, CNN

## Introduction

The poultry industry has established itself as an important source of protein for humanity, and more efficient production management is essential due to the increasing demand driven by the growing global population [1, 2]. In the operation of this industry, the accurate distinction between fertilized and unfertilized eggs is one of the key factors for determining hatching efficiency [3].

Fertilized eggs are essential resources for hatchery operations because they hatch into chicks. However, unfertilized eggs cannot hatch because they are not fertilized, and some fertilized eggs cease development during incubation. Both of these situations directly impact resource wastage and hatch rates within a hatchery. Therefore, accurately identifying and removing unfertilized and hatching failure eggs at an early stage is important for enhancing economic and operational efficiency [3, 4].

Traditionally, the differentiation between fertilized and unfertilized eggs has relied on visual inspection or simple candling methods [5, 6]. These traditional methods are highly dependent on the experience and skill level of the inspector. As a result, they can be subjective and have limited accuracy. Particularly in large-scale hatcheries, these traditional methods can result in human errors or inefficiencies that lead to economic losses. Therefore, there is a pressing need for the development and adoption of more objective and reliable automated methods [7-9].

Recent research using deep learning to differentiate between fertilized and unfertilized eggs can offer higher accuracy than traditional methods because deep learning can process images and recognize and learn complex patterns [10]. These methods are particularly advantageous for detecting subtle differences that are difficult to discern with the naked eye. Convolutional neural networks (CNNs) can be used to analyze images of eggs to help the model learn the subtle differences between fertilized, unfertilized, and hatching failure eggs and assist in differentiating them. Thus, using image processing with deep learning can automate the process of distinguishing between fertilized, unfertilized, and hatching failure eggs in large-scale production environments. This is highly effective in reducing human error and increasing the processing speed in large-scale production environments like hatcheries.

Deep learning models learn through large amounts of data. For egg differentiation, the model requires an initial dataset of images of fertilized, unfertilized, and hatching failure eggs to allow it to learn the characteristics of each class. Training and executing a deep learning model requires high-performance computing resources, and initial data collection and labeling can be expensive and time consuming. Therefore, research using deep learning should develop more efficient and accurate models through data collection and management strategies in order to maximize its significant advantages in making the differentiation of fertilized, unfertilized, and hatching failure eggs in the poultry industry more precise and efficient.

Previous studies on deep learning-based egg fertility classification have typically evaluated only a limited number of CNN architectures under a single experimental setting, and few have simultaneously addressed all three egg categories — fertilized, unfertilized, and hatching failure — under controlled imaging conditions [10]. Furthermore, the issue of data leakage arising from image-level train/test splitting, where multiple images of the same egg appear in both training and test sets, has been largely overlooked, potentially leading to an overestimation of classification accuracy.

To address these gaps, the present study makes the following novel contributions:

1. **Systematic benchmarking of 12 CNN architectures** across four carefully designed experimental scenarios (Cases A–D) that progressively control for data leakage and class composition, providing a comprehensive and fair comparison of model performance.
2. **Egg-level train/test separation** (Cases B and D) to prevent data leakage and ensure a more reliable estimate of real-world generalization performance.
3. **Evaluation under two distinct lighting conditions** (G-LED and W-LED) to simulate the variability encountered in real hatchery environments, thereby assessing the practical robustness of each CNN model.
4. **Practical deployment guidance** for the poultry industry by analyzing the trade-off between classification accuracy and computational resource requirements across all 12 models.

These contributions collectively advance the state of the art in automated egg fertility assessment and provide actionable guidelines for implementing deep learning systems in commercial hatcheries.

Specifically, this study acquires data for deep learning training, designs a deep learning system, and designs a recognition system for fertilized, unfertilized, and hatching failure eggs using the trained deep learning network. Through this, this study seeks to enhance hatch efficiency, reduce resource waste, and maximize economic benefits in the poultry industry. Furthermore, it aims to explore the potential for these technologies to be applied in fields other than the poultry industry and suggests future research directions. These technologies can be expanded into quality control and verification systems across the agriculture industry and play an important role in food safety and quality assurance. Therefore, this study will play a crucial role in laying the foundation for increasing the efficiency of poultry production and contributing to sustainable agricultural development.

## Materials and Methods

### System Overview

The algorithm flow is shown in Figure 1. First, 5$^{th}$ and 6$^{th}$ day eggs in the hatchery were placed in a darkroom. Then, the eggs were photographed inside the darkroom using a camera. The captured egg images went through basic image processing steps. Finally, trained CNN were used to classify the eggs as fertilized, unfertilized, or hatching failure.

### System Design

The system design is shown in Figure 2. G-LED (10W) or W-LED (80W) lights were installed to not only classify egg images in a laboratory setting but also to simulate shooting conditions that can occur in real livestock farms. The experimental data were created under these two different lighting conditions.

### Data Acquisition

As a preliminary study, eggs were placed in the incubator and photographed from the 1st to 12th day to determine the most distinct and visually distinguishable period, preferably within the shortest incubation period. In the case of fertilized eggs, four images were taken per day, collecting a total of 2,605 images (Figure 3). On May 13, 2024, 300 eggs were placed in the incubator, and 1,130 images were obtained on the 5th and 6th days using G-LED lighting. On July 15, 2024, another 200 eggs were placed in the incubator, and 1,475 images were obtained on the 5th and 6th days using W-LED lighting. The classification of fertilized, unfertilized, and hatching failure eggs was determined by breaking the eggs after 12 days of incubation. In the case of hatching failure eggs, the exact time at which development ceased cannot be determined.

If development stops early in the incubation process, the characteristics are similar to those of unfertilized eggs, whereas if development stops several days after incubation, the characteristics resemble those of fertilized eggs. Through all of this preliminary research, it was found that collecting images on the 5th or 6th day after placing eggs in the incubator is optimal, so all images were collected on the 5th or 6th days after loading eggs into the hatchery (Figure 4). The resulting dataset was severely imbalanced across the three egg categories: fertilized eggs accounted for 87.6% of total training images (n=1,826), while unfertilized eggs represented 9.1% (n=190) and hatching failure eggs only 3.4% (n=70) (Figure 5). In the test sets, hatching failure eggs represented only 3.3% of total test images (17–18 out of 520–523 in Cases A and B). This imbalance is an inherent challenge in this domain, as hatching failure eggs are rare in practice and their exact developmental arrest point is unknown, making systematic collection and labeling difficult. Although images were collected under two distinct lighting conditions (G-LED and W-LED) to simulate the variability encountered in real hatchery environments, the effect of lighting condition on CNN classification performance was not explicitly analyzed in this study. The training and test datasets used in all four experimental cases (Cases A–D) comprised images from both lighting conditions without differentiation, meaning that the reported classification performance reflects the combined effect of both conditions. A systematic analysis comparing model performance across the two lighting conditions was not conducted due to the difference in egg sample sizes between the two collection sessions (300 eggs under G-LED vs. 200 eggs under W-LED) and the potential confounding effect of seasonal variation between the May and July collection dates. This is acknowledged as a limitation of the current study, and future work should investigate the effect of lighting condition on classification performance using a balanced, controlled experimental design.

## CNN Training and Testing Data

The dataset was organized into four experimental cases (Cases A–D) to systematically evaluate CNN performance under different data splitting strategies and class compositions. Each case was designed with a specific rationale to progressively address potential sources of bias and reliability concerns, as summarized in Table 1.

**Case A** was designed as a baseline evaluation. Data were randomly selected from the total dataset (2,605 images) in an 8:2 ratio for training and testing, respectively. This conventional random split provides a reference point for comparing the performance of subsequent experimental designs. **Case B** was designed to address a critical limitation of Case A. Since four images per day were captured for each fertilized egg, a random image-level split (as in Case A) may result in images from the same egg appearing in both training and test sets. This data leakage could lead to an overestimation of classification accuracy. Therefore, in Case B, eggs to be excluded from training were randomly pre-selected at the egg level prior to data splitting, and all images of these selected eggs were reserved exclusively as test data. This egg-level separation ensures that the test set contains no images related to eggs seen during training, thereby providing a more reliable estimate of generalization performance. **Case C** was designed to evaluate CNN performance under a simplified binary classification scenario. Hatching failure eggs were excluded from the dataset due to their severely limited sample size (n=70), which was insufficient for reliable CNN training and introduced ambiguity in classification boundaries. Only fertilized and unfertilized eggs (total: 2,536 images) were retained, and data were split using the same random 8:2 image-level ratio as in Case A. **Case D** combined the egg-level data separation strategy of Case B with the binary class setting of Case C. Hatching failure eggs were excluded from both training and testing, and 20% of eggs were pre-selected at the egg level as test data prior to training. This design provides the most rigorous evaluation of the model's ability to distinguish fertilized from unfertilized eggs under realistic hatchery conditions, where data leakage is prevented and classification ambiguity from hatching failure eggs is eliminated.

**CNN Training**

The CNNs used for learning were SqueezeNet [11], ShuffleNet [12], MobileNet-v2 [13], NASNet-Mobile [14], EfficientNet-b0 [15], GoogLeNet [16], Inception-v3 [17], ResNet-18 [18], ResNet-50 [18], AlexNet [19], VGG-16 [20], and VGG-19 [20]. Figure 6 shows the structure of a general CNNs applied in this study. The parameter memory and number of parameters of the CNN are as shown in Table 2. The hardware required for training the proposed CNN consisted of an Intel i9-12900K CPU and 64GB RAM as the central processing unit and an NVIDIA RTX A6000 graphics processing unit. The simulation was performed using the deep learning toolbox embedded in MathWorks' MATLAB® [21]. The parameters required for training were set to the same values for all networks used in the simulations to ensure fairness in performance evaluation.

The training parameters were an initial learning rate of 0.0001, a maximum of 200 iterations, and a mini-batch size of 256. The training data were randomly shuffled for each iteration. The network's optimization algorithm employed stochastic gradient descent (SGD) with momentum. The resulting dataset was severely imbalanced: fertilized eggs accounted for 87.6% of total training images (n=1,826), compared to 9.1% for unfertilized eggs (n=190) and only 3.4% for hatching failure eggs (n=70). In the test sets, hatching failure eggs represented only 3.3% of total test images (17–18 out of 520–523 in Cases A and B), making overall accuracy a potentially misleading metric in this experimental setting. The class imbalance in the dataset was not explicitly addressed during training in this study, and the same training parameters were applied uniformly across all classes. This is acknowledged as an important limitation that likely contributed to the lower classification performance observed for unfertilized and hatching failure eggs. Under standard cross-entropy loss without class weighting, the network optimization is dominated by the majority class (fertilized eggs), which may cause the model to underlearn the discriminative features of the minority classes. Future work should incorporate one or more of the following strategies to mitigate this effect:

- **Data augmentation for minority classes:** Applying augmentation techniques such as random flipping, rotation, and brightness adjustment exclusively to unfertilized and hatching failure egg images to increase their effective training sample size.
- **Class-weighted loss function:** Assigning higher loss weights to minority classes (inversely proportional to class frequency) during training to penalize misclassification of underrepresented categories more heavily.
- **Oversampling/undersampling:** Applying oversampling of minority class images (e.g., SMOTE or random oversampling) or undersampling of the majority class to create a more balanced training distribution.

## Results

Prior to presenting the classification results, it is important to note that overall accuracy is used as a comparative metric in this study for consistency with prior literature. However, given the severe class imbalance in the dataset — fertilized eggs accounting for 87.6% of total training images, compared to 9.1% for unfertilized eggs and 3.4% for hatching failure eggs — overall accuracy should be interpreted with caution, as it is heavily influenced by the majority class. A model could theoretically achieve over 87% accuracy by predicting all eggs as fertilized, without correctly identifying a single unfertilized or hatching failure egg. Per-class recall values are therefore reported alongside overall accuracy in Tables 3~6 to provide a more informative assessment of model performance for each egg category. The test images were taken on different dates and were unrelated to the data used for network training. Compared to the experimental results of Case A, the overall accuracy of the CNNs was slightly lower (Figure 7). Nevertheless, the overall accuracy exceeded 91%.

When looking at only fertilized eggs, all CNNs showed accuracies of over 97%. In the case of unfertilized and hatching failure eggs, all networks showed lower accuracies compared to fertilized eggs, as in Case A. The lower classification performance observed for unfertilized and hatching failure eggs can be attributed to three compounding factors. First, the training dataset was severely imbalanced, with hatching failure eggs (n=70) and unfertilized eggs (n=190) representing only 3.2% and 8.7% of the total training data, respectively, compared to fertilized eggs (n=1,826). Second, the visual characteristics of hatching failure eggs are inherently ambiguous: eggs whose development arrested early in incubation closely resemble unfertilized eggs due to the absence of visible vasculature, while those that arrested later resemble fertilized eggs due to partial vascular network development. This visual ambiguity makes it fundamentally difficult for CNN models to learn consistent discriminative features for hatching failure eggs. Third, the limited diversity of hatching failure egg images further constrained the model's ability to generalize to unseen examples in the test set.

To provide a more comprehensive evaluation beyond overall accuracy, per-class recall values were computed for all CNN models and are summarized in Table 3. The results reveal a substantial disparity in recall across egg categories: while fertilized egg recall exceeded 98% for all models, hatching failure egg recall ranged widely from 5.88% (MobileNet-v2, EfficientNet-b0) to 76.47% (Inception-v3, VGG-19), and unfertilized egg recall ranged from 63.83% (MobileNet-v2) to 93.62% (Inception-v3). These per-class recall values highlight that overall accuracy alone is insufficient to characterize model performance under class imbalance, as high overall accuracy is largely driven by the dominant fertilized egg class.

A descriptive comparison of overall accuracy across the 12 CNN models in Case A revealed a performance range of 5.00 percentage points (from 93.65% for MobileNet-v2 and ResNet-50 to 98.65% for Inception-v3). Models were grouped into three performance tiers:

- **High performance (>97%):** Inception-v3 (98.65%), AlexNet (98.27%), VGG-19 (98.08%), VGG-16 (97.69%), GoogLeNet (96.92%)
- **Mid performance (95–97%):** SqueezeNet (95.77%), NASNet-Mobile (95.58%), EfficientNet-b0 (95.00%)
- **Lower performance (<95%):** ShuffleNet (94.62%), ResNet-18 (94.04%), MobileNet-v2 (93.65%), ResNet-50 (93.65%)

In Case B, the overall performance results of the CNNs showed that the more network training parameters there are, the higher the test performance (Figure 8). Per-class recall analysis for Case B (Table 4) revealed a critical finding: the recall for hatching failure eggs dropped to 0% for six out of twelve CNN models (SqueezeNet, ShuffleNet, EfficientNet-b0, GoogLeNet, AlexNet, and VGG-16 partial), indicating a complete failure to detect this class under egg-level data separation. Even the best-performing model (NASNet-Mobile) achieved only 27.78% recall for hatching failure eggs. This result demonstrates that overall accuracy is highly misleading in this setting, as models achieving over 90% overall accuracy may simultaneously fail entirely at classifying the minority class.

In Case B, overall accuracy ranged from 90.82% (NASNet-Mobile) to 95.79% (Inception-v3 and VGG-16), a spread of 4.97 percentage points. Compared to Case A, all models showed a consistent decrease in overall accuracy under egg-level data separation, confirming that image-level splitting in Case A led to an overestimation of classification performance. The performance ranking among models remained largely consistent between Cases A and B, with Inception-v3, VGG-16, and VGG-19 maintaining top positions and NASNet-Mobile showing the largest performance drop (95.58% → 90.82%).

Case C used only fertilized and unfertilized eggs from the total data (2,536), and randomly selected them in an 8:2 ratio for training and testing data. The overall classification performance of the CNNs was excellent, with over 97.02%. Particularly, for fertilized eggs, all networks showed outstanding performance evaluation results of more than 99.34% (Figure 9).

For unfertilized eggs, all networks showed classification performances of over 82.98% except for MobileNet-v2. Looking at the performance of each network, SqueezeNet, ShuffleNet, MobileNet-v2, and NASNet-Mobile were efficient for application in resource-limited environments due to their small parameter memory and fewer parameters resulting in fast training times. However, their accuracy was significantly lower compared to larger models. EfficientNet-b0 offered balanced accuracy compared to its relatively low parameter size but required longer training times than other network models. GoogLeNet, Inception-v3, and ResNet-18 had high accuracy and moderate parameter sizes and training times. Inception-v3 showed the highest accuracy among the analyzed models but required long training times due to its very high parameter memory and large input size (299×299). ResNet-50, with its deep architecture, provided good accuracy but required high parameter memory and long training times. AlexNet, with its relatively simple architecture, had high accuracy but used very high parameter memory. VGG-16 and VGG-19 showed extremely high accuracy but required long training times due to their extremely large parameter sizes and memory. Per-class recall values for Case C (Table 5) showed substantial improvement over Cases A and B, particularly for unfertilized eggs. Recall for unfertilized eggs ranged from 74.47% (MobileNet-v2) to 97.87% (Inception-v3), with seven out of twelve models exceeding 85%. These results confirm that excluding hatching failure eggs from the classification task significantly improves model performance for unfertilized egg detection, likely due to the reduction of visual ambiguity in the training data.

In Case C, overall accuracy ranged from 97.02% (MobileNet-v2) to 99.80% (Inception-v3), a spread of 2.78 percentage points — notably narrower than Cases A and B. This reduced performance gap between models suggests that the exclusion of hatching failure eggs substantially simplified the classification task for all networks. Models were grouped into performance tiers:

- **High performance (>99%):** Inception-v3 (99.80%), EfficientNet-b0 (99.80%), NASNet-Mobile (99.01%), GoogLeNet (99.40%), AlexNet (99.40%), VGG-16 (99.60%), VGG-19 (99.60%)
- **Mid performance (98–99%):** SqueezeNet (98.21%), ShuffleNet (98.61%), ResNet-18 (98.41%), ResNet-50 (98.61%)
- **Lower performance (<98%):** MobileNet-v2 (97.02%)

Case D determined the test images by randomly pre-selecting 20% of the eggs, similar to Case B, except hatching failure eggs were excluded from training. After training the networks, the images of the selected eggs were used as test data to ensure the networks' reliability. Therefore, the test images were unrelated to the data used for network training because they were different images taken on different dates. The overall classification performance of the CNNs was excellent, with over 95.84% (Table 6 and Figure 10). Particularly for fertilized eggs, all networks showed outstanding performance evaluation results of more than 99.12%. Under the most rigorous experimental setting (Case D), unfertilized egg recall ranged from 58.33% (SqueezeNet) to 87.50% (Inception-v3, AlexNet, VGG-16, VGG-19). Compared to Case C, the egg-level separation in Case D resulted in a consistent decrease in unfertilized egg recall across all models, reflecting the more challenging nature of this evaluation protocol. These findings suggest that image-level splitting (Cases A and C) tends to overestimate real-world classification performance, and that egg-level evaluation (Cases B and D) provides a more conservative and reliable estimate of generalization capability in hatchery deployment scenarios.

In Case D, overall accuracy ranged from 95.84% (SqueezeNet) to 98.81% (VGG-19), a spread of 2.97 percentage points. Consistent with the pattern observed between Cases A and B, the egg-level separation in Case D resulted in lower overall accuracy compared to Case C for all models, with the largest drops observed for SqueezeNet (98.21% → 95.84%) and MobileNet-v2 (97.02% → 96.44%). This consistent pattern across all four cases strongly suggests that image-level data splitting systematically overestimates classification performance, and that egg-level evaluation provides a more reliable and conservative estimate of real-world generalization.

In the case of unfertilized eggs, SqueezeNet and MobileNet-v2 showed slightly lower performance, but the results were still reasonable. While Inception-v3 and VGG models offered high accuracy, they required significant computational resources, whereas GoogLeNet and EfficientNet-b0 presented a good balance between accuracy and efficiency. MobileNet-v2 and ShuffleNet were suitable for mobile and resource-constrained applications, and ResNet-18 and ResNet-50 provided a good balance of depth and accuracy. The choice of network should be determined by the balance between available computational resources, desired speed, and accuracy. Table 7 compares the unfertilized egg recall rate (%) across cases A–D for all 12 CNN models.

## Discussion

In recent years, advancements in science and technology have opened up new possibilities for solving previously unsolvable problems. In particular, optical techniques and advanced image analysis methods offer the potential to nondestructively and quickly distinguish between fertilized and unfertilized eggs. For example, analyses using near-infrared spectroscopy (NIRS) or hyperspectral imaging technology can detect subtle chemical and structural differences inside eggs. These technologies not only provide high accuracy but are also suitable for large-scale processing, making them highly applicable for industrial use. However, NIRS generates complex spectral data so it requires sophisticated data interpretation using advanced statistical methods and machine learning algorithms. Additionally, standardization is challenging because results can vary with environmental conditions such as temperature and humidity.

Hyperspectral imaging technology also requires high-speed computing resources and sophisticated analysis algorithms for processing and analyzing large amounts of data. This makes real-time processing difficult in large-scale production environments. Moreover, these optical analysis methods are expensive, and the high maintenance costs can be a financial burden for poultry farms.

Besides optical methods, there are techniques that utilize biological markers. Fertilized eggs exhibit specific biochemical or hormonal changes post-fertilization, which can be detected to identify fertilized eggs. This method allows for a more precise analysis of the egg's internal state and can resolve the limitations of traditional visual inspection methods. Additionally, methods based on biological markers are nondestructive so they allow for real-time monitoring without affecting the incubation process.

However, these methods require biological markers that are sufficiently specific and sensitive to distinguish between fertilized and unfertilized eggs. Biological variability means that the concentration of markers may differ between individual eggs under the same conditions, potentially compromising result consistency. Detecting these biological markers also requires complex experimental procedures and expensive equipment. These methods take considerable time to distinguish between fertilized and unfertilized eggs and are challenging to apply in mass production environments outside the laboratory setting.

Recently, research using deep learning to differentiate between fertilized and unfertilized eggs has shown that image processing and the ability to recognize and learn complex patterns can provide higher accuracy than traditional methods. These methods are particularly advantageous for detecting subtle differences that are difficult to discern visually. This study comprehensively evaluated the performances of CNNs to distinguish between fertilized, unfertilized, and hatching failure eggs. Table 8 concisely describes the advantages and characteristics of each network.

In particular, this study aimed to verify the applicability of deep learning in the poultry field by focusing on 12 widely researched and well-known CNN algorithms. All CNNs exhibited an overall classification performance accuracy of over 93.65%. Specifically, all networks were able to classify fertilized eggs with more than 98% accuracy. In contrast, unfertilized and hatching failure eggs showed lower accuracy compared to fertilized eggs. The misclassification patterns observed in this study can be explained by the biological characteristics of each egg category at the imaging timepoint (days 5–6 of incubation). Fertilized eggs consistently showed high classification recall (>96%) across all models and cases, which can be attributed to the well-developed vascular network visible under candling-style illumination at this stage — a visually salient feature that CNN models can readily learn. In contrast, hatching failure eggs presented a fundamentally ambiguous classification target: as noted in the Data Acquisition section, eggs whose development arrested early during incubation closely resemble unfertilized eggs in appearance, while those that arrested later retain partial vascular structures similar to fertilized eggs. This biological continuum of visual appearance — ranging from unfertilized-like to fertilized-like depending on the timing of developmental arrest — means that hatching failure eggs do not form a visually coherent category, making it inherently difficult for CNN models to learn consistent discriminative features for this class. This biological explanation is consistent with the extremely variable and generally low recall observed for hatching failure eggs across all models (0%–76.47% in Cases A and B) and the comparatively lower but more stable recall for unfertilized eggs (58.33%–93.62% across cases).

The CNN training for hatching failure and unfertilized eggs utilized only 70 and 190 images, respectively, which was insufficient for effective network training. This data scarcity, combined with severe class imbalance, led to a systematic bias toward the majority class (fertilized eggs) during training. As a result, the models tended to misclassify minority class eggs as fertilized eggs, particularly under the egg-level evaluation protocol (Cases B and D) where test images were completely unseen during training. The per-class recall analysis (Tables 3-6) confirms this pattern: hatching failure egg recall dropped to 0% for multiple models in Case B, and unfertilized egg recall was consistently lower in egg-level evaluation cases (B and D) compared to image-level cases (A and C). These findings highlight a critical limitation of the current study and underscore the need for more balanced data collection strategies in future work. Specifically, targeted collection of hatching failure egg images across different developmental arrest time points would help the model learn the full spectrum of visual variation within this class. Until sufficient data can be collected, techniques such as class-weighted loss functions, synthetic data augmentation (e.g., generative adversarial networks), or transfer learning from related domains may offer viable mitigation strategies. It should be noted that no class imbalance handling technique was applied during the training phase of this study. The uniform application of standard cross-entropy loss across all classes means that the gradient updates during training were disproportionately influenced by the fertilized egg class, potentially suppressing the model's ability to learn discriminative features for unfertilized and hatching failure eggs. This methodological limitation should be addressed in future studies through the application of class-weighted loss functions or data augmentation strategies targeting the minority classes.

A cross-case descriptive analysis of CNN performance reveals several consistent patterns. First, there is a general positive relationship between the number of model parameters and classification accuracy: larger models such as Inception-v3 (23.9M parameters), VGG-16 (138M), and VGG-19 (144M) consistently ranked among the top performers across all four cases, while lightweight models such as SqueezeNet (1.24M), ShuffleNet (1.40M), and MobileNet-v2 (3.50M) consistently ranked lower. However, this relationship is not strictly monotonic: AlexNet (61M parameters) achieved performance comparable to or exceeding that of ResNet-50 (25.6M) across multiple cases, suggesting that architectural design — rather than parameter count alone — plays an important role in determining classification performance.

Second, the performance gap between models was consistently larger for minority classes (unfertilized and hatching failure eggs) than for the majority class (fertilized eggs). For fertilized eggs, all 12 models achieved recall above 96% across all cases, indicating that this class is easily learnable regardless of model complexity. In contrast, for unfertilized eggs in Case A, recall ranged from 63.83% (MobileNet-v2) to 93.62% (Inception-v3) — a gap of nearly 30 percentage points — highlighting that model selection has a substantially greater impact on minority class performance than on majority class performance.

Third, the performance spread across models was consistently narrower in Cases C and D (excluding hatching failure eggs) than in Cases A and B (including hatching failure eggs). The range of overall accuracy across models was 5.00 percentage points in Case A, 4.97 in Case B, 2.78 in Case C, and 2.97 in Case D. This convergence in Cases C and D suggests that the presence of hatching failure eggs in the dataset introduces differential difficulty across models, particularly disadvantaging lightweight architectures that lack the capacity to capture the subtle visual features distinguishing hatching failure eggs from the other two categories. The misleading nature of overall accuracy under class imbalance is particularly pronounced in Cases A and B, where hatching failure eggs represent only 3.3% of the test set (17–18 images out of 520–523 total). In this setting, a model that completely fails to identify any hatching failure egg — as observed for six models in Case B (0% recall for SqueezeNet, ShuffleNet, EfficientNet-b0, GoogLeNet, AlexNet, and VGG-16) — still achieves overall accuracy exceeding 92%, solely due to correct classification of the majority fertilized egg class. This finding reinforces the importance of reporting per-class performance metrics rather than relying solely on overall accuracy when evaluating classification systems with severely imbalanced datasets.

GoogLeNet showed balanced performance with high accuracy for both fertilized and unfertilized eggs. Particularly, networks like Inception-v3, AlexNet, Vgg16, and Vgg18 showed favorable accuracy results compared to other networks in many cases due to their network depth and number of parameters. However, network training time tends to lengthen as the number of learning parameters increases [22-24]. Inceptionv3 delivered top performance in classification but required a significant amount of training time, necessitating a suitable balance between memory usage and training time.

The per-class recall analysis reveals an important limitation of relying solely on overall accuracy for performance evaluation in imbalanced datasets. As shown in Table 5, while all CNN models achieved high recall for fertilized eggs (>97% across all cases), the recall for hatching failure eggs was markedly lower and highly variable across models, ranging from 0% to 76.47% in Cases A and B. This extreme variability suggests that overall accuracy is heavily influenced by the majority class (fertilized eggs) and does not adequately reflect the model's ability to correctly identify minority classes. Future studies should therefore report per-class recall, precision, and F1-score as standard evaluation metrics, and should prioritize model selection based on minority class performance rather than overall accuracy alone.

The practical applicability of the proposed system is supported by several key design features of this study. First, the image acquisition system described in this study — comprising a darkroom enclosure, G-LED or W-LED illumination, and a standard camera — is deliberately designed to be simple, low-cost, and replicable in real hatchery environments. Unlike optical analysis methods such as near-infrared spectroscopy (NIRS) or hyperspectral imaging, which require expensive specialized equipment and complex data processing pipelines, the proposed system relies on standard RGB imaging and CNN-based classification, making it accessible to hatcheries with limited technical resources.

Second, the optimal imaging timepoint identified in this study - days 5–6 of incubation - represents an early enough stage to allow timely removal of non-viable eggs before significant incubator space and energy resources are consumed, while also providing sufficient visual contrast between egg categories for reliable CNN-based classification. Early removal of unfertilized eggs at this stage can directly improve hatchery efficiency by freeing incubator capacity for viable eggs and reducing energy consumption associated with maintaining non-viable eggs throughout the full incubation period.

Third, the range of CNN models evaluated in this study provides hatchery operators with flexible deployment options based on available computational resources. Lightweight models such as SqueezeNet (4.7 MB, 8 min training) and ShuffleNet (5.5 MB, 26 min training) are suitable for deployment on mobile or embedded devices with limited processing power, making them applicable in small-scale or resource-constrained farm settings. In contrast, higher-accuracy models such as Inception-v3, VGG-16, and VGG-19, while requiring greater computational resources, are better suited for large-scale commercial hatcheries where maximizing classification accuracy — particularly for unfertilized egg detection — is the primary operational priority. The choice of CNN model should therefore be guided by the specific operational requirements and computational infrastructure available at each hatchery facility.

## Conclusions

The distinction between fertilized and unfertilized eggs from images on the 5th to 6th day after incubation was found to be a highly effective and efficient method for all 12 CNN algorithms. Thus, the selection of CNNs should be based on available computational resources and the desired speed and accuracy. However, network training can be difficult due to the challenge of acquiring data for hatching failure eggs. The exact point of hatching failure is unclear, which makes the distinction between fertilized and unfertilized eggs ambiguous. Nonetheless, the proposed system offers a practical and accessible solution for enhancing hatching efficiency in real hatchery environments. By enabling early and reliable identification of unfertilized eggs at days 5–6 of incubation using standard RGB imaging and CNN-based classification, the system can reduce resource waste associated with maintaining non-viable eggs throughout the full incubation period, optimize incubator space utilization, and contribute to improved economic outcomes in hatchery operations. The flexibility of the proposed framework — supporting deployment across a range of CNN architectures from lightweight mobile models to high-accuracy deep networks — allows hatchery operators to select the most appropriate model based on their specific computational resources and accuracy requirements. Additionally, these technologies can be applied in other fields beyond the poultry industry. These technologies can be expanded to quality control and verification systems across the agriculture sector and play an important role in food safety and quality assurance.

## Acknowledgments

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

## References


1. Banday MT, Adil S, Sheikh I, Hamadani H, Qadri FI, Sahfi ME, Sait HS, El-Mageed TA A, Salem H, Taha A, El-Saadony M, El-Hack MAA. The use of silkworm pupae (Bombyx mori) meal as an alternative protein source for poultry. World's Poult Sci J. 2023;79:119-34. https://doi.org/10.1080/00439339.2023.2163955

2. Burt D. Applications of biotechnology in the poultry industry. World's Poult Sci J. 2002;58:13-15. https://doi.org/10.1079/WPS20020002

3. Ori AMK. Review of the Factors That Influence Egg Fertility and Hatchabilty in Poultry. Int J Poult Sci. 2011;10:483-92. https://doi.org/10.3923/ijps.2011.483.492

4. So JH, Joe SY, Hwang SH, Hong SJ, Lee SH. Current advances in detection of abnormal egg: a review. J Anim Sci Technol. 2022;64:813-29. https://doi.org/10.5187/jast.2022.e56

5. Adegbenjo AO, Liu L, Ngadi M. Non-Destructive Assessment of Chicken Egg Fertility. Sensors. 2020:20:5546. https://doi.org/10.3390/s20195546

6. Ketelaere BD, Bamelis F, Kemps B, Decuypere E, Baerdemaeker JD. Non-destructive measurements of the egg quality. World's Poult Sci J. 2004:60:289-302. https://doi.org/10.1079/WPS200417

7. Yang X, Bist RB, Subedi S, Chai L. A Computer Vision-Based Automatic System for Egg Grading and Defect Detection. Animals. 2023:13(14):2354. https://doi.org/10.3390/ani13142354

8. Saifullah S, Suryotomo AP. Identification of chicken egg fertility using SVM classifier based on first-order statistical feature extraction. J Ilmu Komun. 2021;13:285-93. https://doi.org/10.33096/ilkom.v13i3.937.285-293

9. Saifullah S, Permadi VA. Comparison of Egg Fertility Identification based on GLCM Feature Extraction using Backpropagation and K-means Clustering Algorithms. In: Proceedings of the 2019 5th International Conference on Science and Technology (ICST); 2019. p.140-45. https://doi.org/10.1109/ICSITech46713.2019.8987496

10. Çevik, KK, Koçer HE, Boğa M. Deep Learning Based Egg Fertility Detection. Vet Sci. 2022;9(10):574. https://doi.org/10.3390/cetsci9100574

11. Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5 MB model size. arXiv.

2016;1602.07360. https://doi.org/10.48550/arXiv.1602.07360

12. Zhang X, Zhou X, Lin M, Sun J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2018. p. 6848-56. https://doi.org/10.1109/CVPR.2018.00716

13. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. Mobilenetv2:Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR); 2018. p. 4510-20. https://doi.org/10.1109/CVPR.2018.00474

14. Zoph B, Vasudevan V, Shlens J, Le QV. Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2018. p. 8697-710. https://doi.org/10.48550/arXiv.1707.07012

15. Tan M, Le QV. EfficientNET: Rethinking model scaling for convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2018. p. 8697-710. https://doi.org/10.48550/arXiv.1905.11946

16. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015. p. 1-9. https://doi.org/10.48550/arXiv.1409.4842

17. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 2818-26. https://doi.org/10.48550/arXiv.1512.00567

18. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. p.770-8. https://doi.org/10.48550/arXiv.1512.03385

19. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Commun ACM. 2017;60:84-90. https://doi.org/10.1145/3065386

20. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014. https://doi.org/10.48550/arXiv.1409.1556

21. MathWorks. Deep Learning Toolbox documentation Internet. MathWorks Inc. [cited

2025 Jun 12] https://mathworks.com/help/deeplearning/index.html

22. Saifullah S, Dreżewski R, Yudhana A, Pranolo A, Kaswijianti W, Suryotomo AP, Putra SAJ, Khaliduzzaman A, Prabuwono AS, Japkowicz N. Nondestructive chicken egg fertility detection using CNN-transfer learning algorithms. J Ilm Tek Elektro Komput Inform. 2023;9:854-71. https://doi.org/10.26555/jiteki.v9i3.26722

23. Swarup C, Singh KU, Kumar A, Pandey SK, Varshney N, Singh T. Brain tumor detection using CNN, AlexNet & GoogLeNet ensembling learning approaches. Electron Res Arch. 2023;31:2900-24. https://doi.org/10.3934/era.2023146

24. Lin CJ, Lin CH, Sun CC, Wang SH. Evolutionary-fuzzy-integral-based convolutional neural networks for facial image classification. Electronics. 2019;8:997. https://doi.org/10.3390/electronics8090997

Table 1. Data organization for deep learning training

| Case | A | B | C | D |
|---|---|---|---|---|
| Total number of images | 2606 | 2606 | 2536 | 2536 |
| Random selection ratio (Training:Test) | 8:2 | Randomly select eggs early and choose images of the selected eggs as test data | 8:2 | Randomly select eggs early and choose images of the selected eggs as test data |
| Training data | 2086 | 2083 | 2016 | 2014 |
| Fertilized | 1826 | 1825 | 1826 | 1825 |
| Unfertilized | 190 | 189 | 190 | 189 |
| Hatching failure | 70 | 69 | - | - |
| Test data | 520 | 523 | 503 | 505 |
| Fertilized | 456 | 457 | 456 | 457 |
| Unfertilized | 47 | 48 | 47 | 48 |
| Hatching failure | 17 | 18 | - | - |

Table 2. Parameter memory and number of parameters in the network [11]

| Networks | Memory of Parameter(Unit:Mega) | No. of Parameter (Unit:Mega) | Image input size |
|---|---|---|---|
| SqueezeNet [1] | 4.7 | 1.24 | 227×227×3 |
| ShuffleNet [2] | 5.5 | 1.40 | 224×224×3 |
| MobileNet-v2 [3] | 14 | 3.50 | 224×224×3 |
| NASNet-Mobile [4] | 20 | 5.30 | 224×224×3 |
| EfficientNet-b0 [5] | 20 | 5.30 | 224×224×3 |
| GoogLeNet [6] | 27 | 7.00 | 224×224×3 |
| Inception-v3 [7] | 91 | 23.90 | 299×299×3 |
| ResNet-18 [8] | 45 | 11.70 | 224×224×3 |
| ResNet-50 [8] | 98 | 25.60 | 224×224×3 |
| AlexNet [9] | 233 | 61.00 | 227×227×3 |
| VGG-16 [10] | 528 | 138.00 | 224×224×3 |
| VGG-19 [10] | 548 | 144.00 | 224×224×3 |

Table 3. Per-class recall (%) of 12 CNN models for Case A. (Fertilized: n=456, Hatching failure: n=17, Unfertilized: n=47)

| Network | Fertilized Recall (%) | Hatching Failure Recall (%) | Unfertilized Recall (%) | Overall Accuracy (%) |
|---|---|---|---|---|
| SqueezeNet [1] | 99.56 | 41.18 | 78.72 | 95.77 |
| ShuffleNet [2] | 99.78 | 23.53 | 70.21 | 94.62 |
| MobileNet-v2 [3] | 100.00 | 5.88 | 63.83 | 93.65 |
| NASNet-Mobile [4] | 98.25 | 47.06 | 87.23 | 95.58 |
| EfficientNet-b0 [5] | 98.90 | 5.88 | 89.36 | 95.00 |
| GoogLeNet [6] | 99.78 | 41.18 | 89.36 | 96.92 |
| Inception-v3 [7] | 100.00 | 76.47 | 93.62 | 98.65 |
| ResNet-18 [8] | 98.46 | 35.29 | 72.34 | 94.04 |
| ResNet-50 [8] | 99.12 | 23.53 | 65.96 | 93.65 |
| AlexNet [9] | 100.00 | 70.59 | 91.49 | 98.27 |
| VGG-16 [10] | 99.78 | 64.71 | 89.36 | 97.69 |
| VGG-19 [10] | 100.00 | 76.47 | 87.23 | 98.08 |

Table 4. Per-class recall (%) of 12 CNN models for Case B. (Fertilized: n=457, Hatching failure: n=18, Unfertilized: n=48)

| **Network** | **Fertilized Recall (%)** | **Hatching Failure Recall (%)** | **Unfertilized Recall (%)** | **Overall Accuracy (%)** |
|---|---|---|---|---|
| SqueezeNet [1] | 99.78 | 0.00 | 70.83 | 93.69 |
| ShuffleNet [2] | 99.56 | 0.00 | 68.75 | 93.31 |
| MobileNet-v2 [3] | 99.56 | 16.67 | 58.33 | 92.93 |
| NASNet-Mobile [4] | 96.28 | 27.78 | 62.50 | 90.82 |
| EfficientNet-b0 [5] | 98.47 | 0.00 | 87.50 | 94.07 |
| GoogLeNet [6] | 99.78 | 0.00 | 83.33 | 94.84 |
| Inception-v3 [7] | 99.78 | 11.11 | 89.58 | 95.79 |
| ResNet-18 [8] | 97.59 | 11.11 | 79.17 | 92.93 |
| ResNet-50 [8] | 98.03 | 5.56 | 68.75 | 92.16 |
| AlexNet [9] | 100.00 | 0.00 | 85.42 | 95.22 |
| VGG-16 [10] | 100.00 | 5.56 | 89.58 | 95.79 |
| VGG-19 [10] | 99.56 | 5.56 | 91.67 | 95.60 |

Table 5. Per-class recall (%) of 12 CNN models for Case C. (Fertilized: n=456, Unfertilized: n=47; hatching failure eggs excluded)

| Network | Fertilized Recall (%) | Unfertilized Recall (%) | Overall Accuracy (%) |
|---|---|---|---|
| SqueezeNet [1] | 99.78 | 82.98 | 98.21 |
| ShuffleNet [2] | 100.00 | 85.11 | 98.61 |
| MobileNet-v2 [3] | 99.34 | 74.47 | 97.02 |
| NASNet-Mobile [4] | 99.78 | 91.49 | 99.01 |
| EfficientNet-b0 [5] | 99.56 | 85.11 | 98.21 |
| GoogLeNet [6] | 100.00 | 93.62 | 99.40 |
| Inception-v3 [7] | 100.00 | 97.87 | 99.80 |
| ResNet-18 [8] | 100.00 | 82.98 | 98.41 |
| ResNet-50 [8] | 100.00 | 85.11 | 98.61 |
| AlexNet [9] | 100.00 | 93.62 | 99.40 |
| VGG-16 [10] | 100.00 | 95.74 | 99.60 |
| VGG-19 [10] | 100.00 | 95.74 | 99.60 |

Table 6. Per-class recall (%) of 12 CNN models for Case D. (Fertilized: n=457, Unfertilized: n=48; hatching failure eggs excluded)

| Network | Fertilized Recall (%) | Unfertilized Recall (%) | Overall Accuracy (%) |
|---|---|---|---|
| SqueezeNet [1] | 99.78 | 58.33 | 95.84 |
| ShuffleNet [2] | 99.78 | 77.08 | 97.62 |
| MobileNet-v2 [3] | 99.78 | 64.58 | 96.44 |
| NASNet-Mobile [4] | 98.03 | 85.42 | 96.83 |
| EfficientNet-b0 [5] | 99.56 | 81.25 | 97.82 |
| GoogLeNet [6] | 100.00 | 79.17 | 98.02 |
| Inception-v3 [7] | 99.78 | 87.50 | 98.61 |
| ResNet-18 [8] | 99.12 | 79.17 | 97.23 |
| ResNet-50 [8] | 100.00 | 68.75 | 97.03 |
| AlexNet [9] | 99.78 | 87.50 | 98.61 |
| VGG-16 [10] | 99.78 | 87.50 | 98.61 |
| VGG-19 [10] | 100.00 | 87.50 | 98.81 |

Table 7. Comparison of unfertilized egg recall (%) across Cases A–D for all 12 CNN models.

| **Network** | **Case A** | **Case B** | **Case C** | **Case D** |
|---|---|---|---|---|
| SqueezeNet [1] | 78.72 | 70.83 | 82.98 | 58.33 |
| ShuffleNet [2] | 70.21 | 68.75 | 85.11 | 77.08 |
| MobileNet-v2 [3] | 63.83 | 58.33 | 74.47 | 64.58 |
| NASNet-Mobile [4] | 87.23 | 62.50 | 91.49 | 85.42 |
| EfficientNet-b0 [5] | 89.36 | 87.50 | 85.11 | 81.25 |
| GoogLeNet [6] | 89.36 | 83.33 | 93.62 | 79.17 |
| Inception-v3 [7] | 93.62 | 89.58 | 97.87 | 87.50 |
| ResNet-18 [8] | 72.34 | 79.17 | 82.98 | 79.17 |
| ResNet-50 [8] | 65.96 | 68.75 | 85.11 | 68.75 |
| AlexNet [9] | 91.49 | 85.42 | 93.62 | 87.50 |
| VGG-16 [10] | 89.36 | 89.58 | 95.74 | 87.50 |
| VGG-19 [10] | 87.23 | 91.67 | 95.74 | 87.50 |

Table 8. Strengths and Weaknesses of Each Network

| Network | Strengths | Weaknesses |
|---|---|---|
| SqueezeNet | Low parameter memory (4.7MB) makes it suitable for resource-constrained environments, with fast training time (8 minutes). | Slightly lower accuracy compared to larger models. |
| ShuffleNet | Efficient with low parameter memory (5.5 MB), ideal for mobile applications. | Slightly lower accuracy compared to some more complex models. |
| MobileNet-v2 | Provides a balance between model complexity and computational efficiency. | Slightly lower accuracy in unfertilized scenarios. |
| NASNet-Mobile | Offers high accuracy, especially strong in unfertilized scenarios. Weaknesses: Requires more memory and longer training time. | Requires more memory and longer training time. |
| EfficientNet-b0 | High fertilized accuracy with balanced resource consumption. | Moderate unfertilized accuracy. |
| GoogLeNet | High accuracy with efficient resource usage and rapid training time. | May not meet the highest accuracy needs. |
| Inception-v3 | Provides the highest accuracy among analyzed models. | Requires very high memory usage and longer training time. |
| ResNet-18 | Quick training time and good accuracy. | Lower accuracy in unfertilized scenarios. |
| ResNet-50 | Offers high accuracy with moderate resource use. | High memory demand. |
| AlexNet | Fast training time and high accuracy. | High memory demand. |
| Vgg-16 | Provides very high accuracy. | Requires extensive memory and parameter size, leading to longer training time. |
| Vgg-19 | Similar high accuracy as VGG-16. | Requires even greater memory and parameter size. |

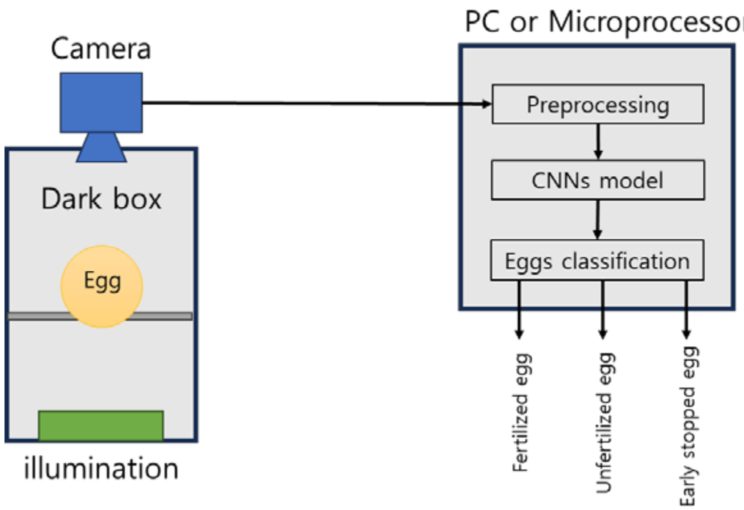



Figure 1. Block diagram of the proposed system

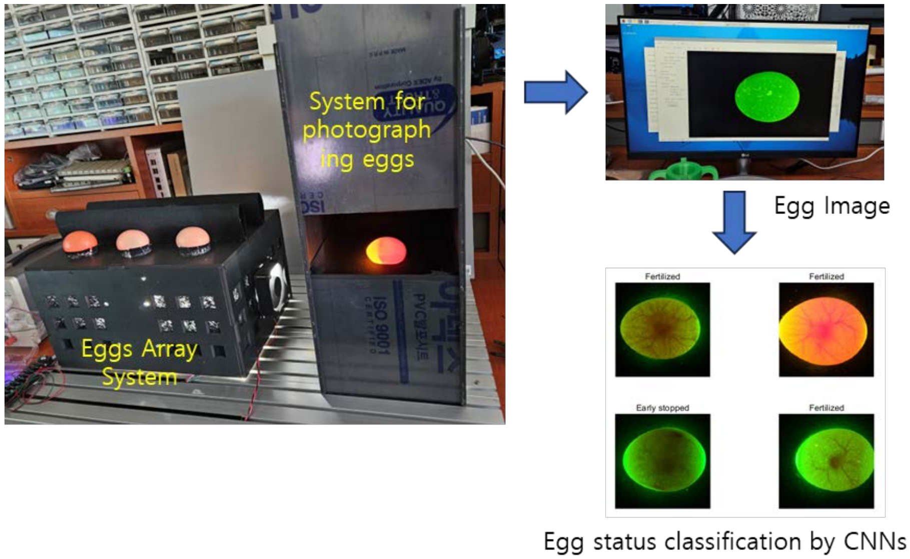



Figure 2. System for egg status classification

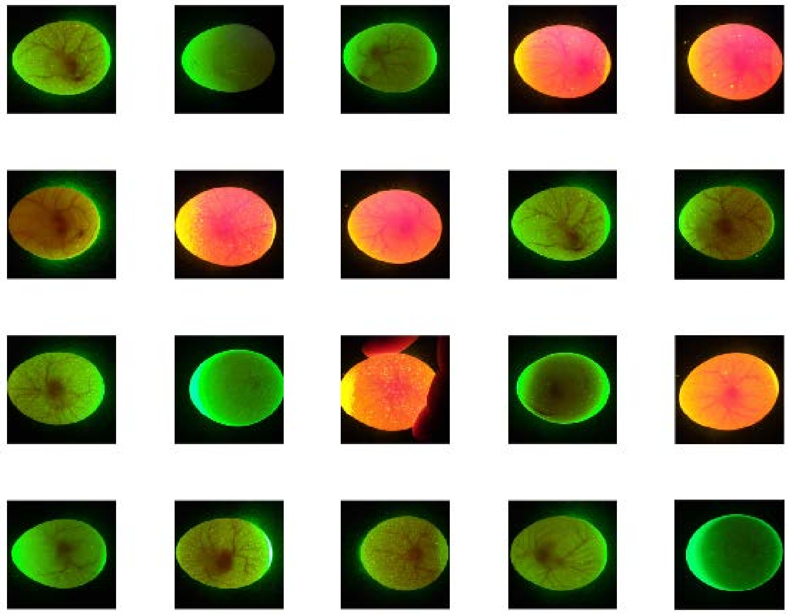


Figure 3. Sample images for learning

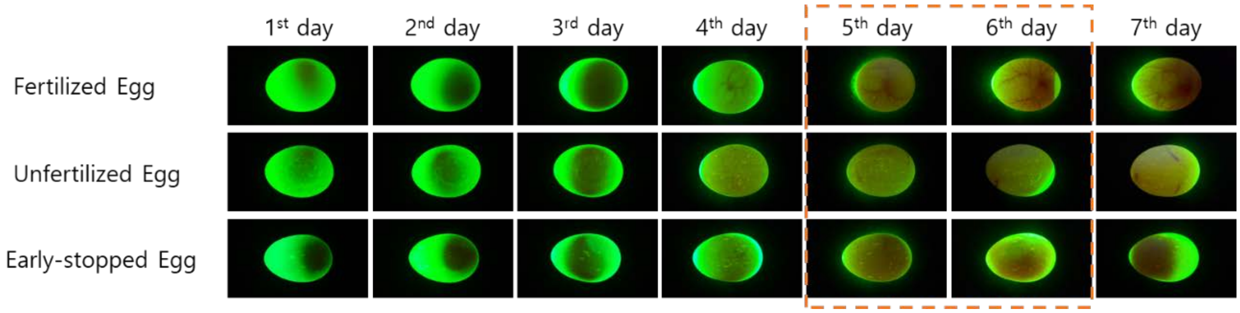
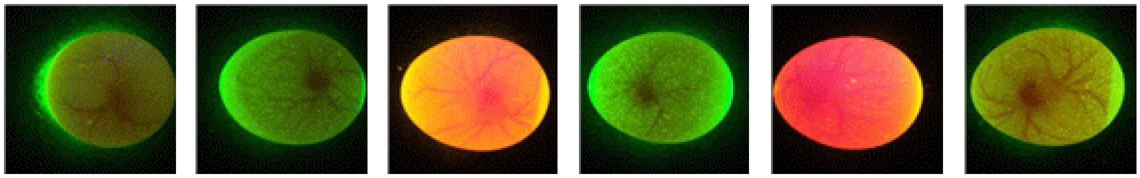
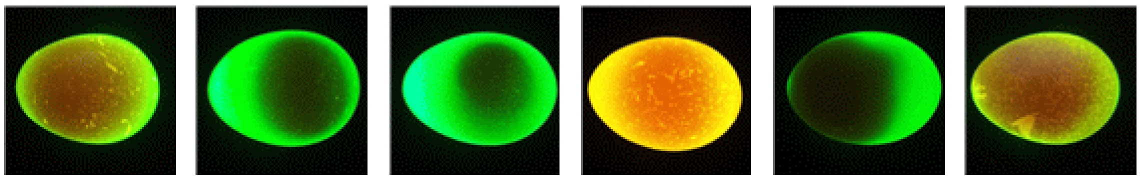


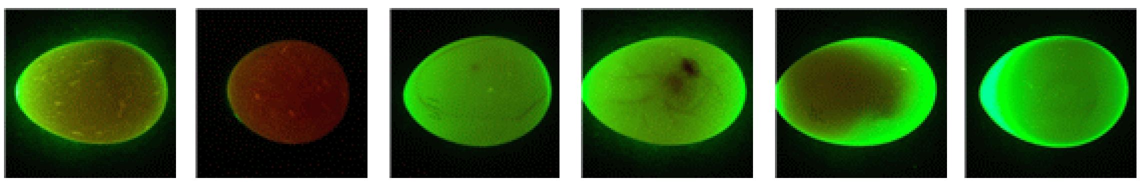

Figure 4. The fertilized, unfertilized, and hatching failure eggs by date



(a) Fertilized Eggs



(b) Unfertilized Eggs



(c) Hatching Failure Eggs

Figure 5. The fertilized, unfertilized, and hatching failure eggs taken by a camera

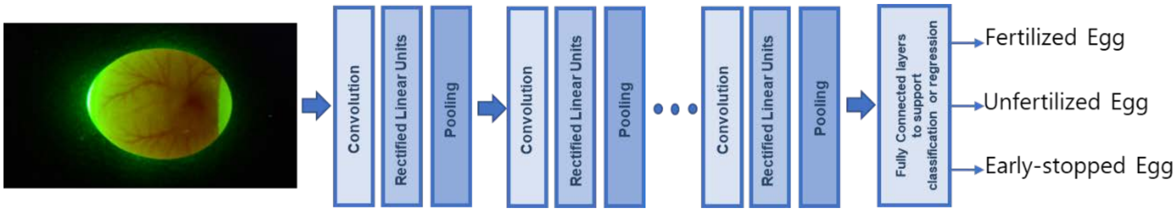
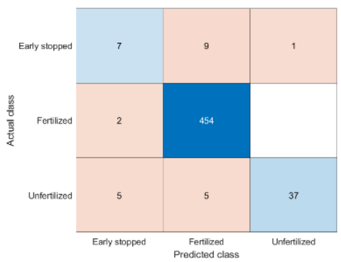
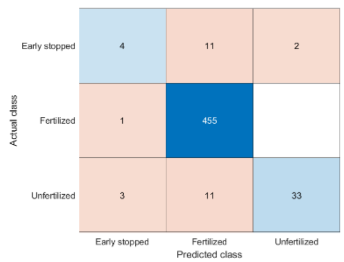


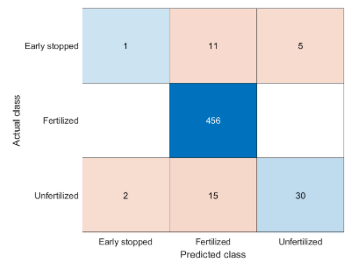

Figure 6. The structure of CNN(Convolutionary Neural Network)



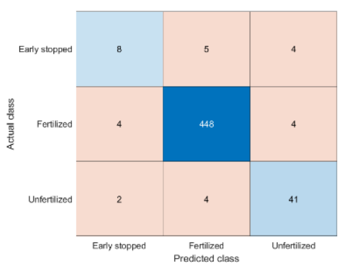

a) SqueezeNet:0.95769



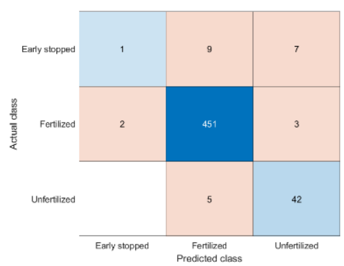

(b) ShuffleNet:0.94615



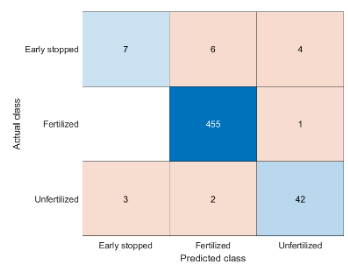

(c) mobileNetv2:0.93654



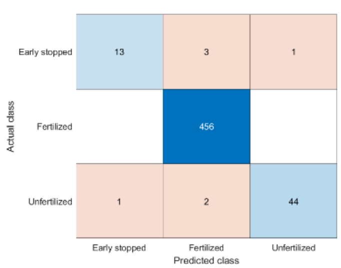

(d) NASNetmobile:0.95577



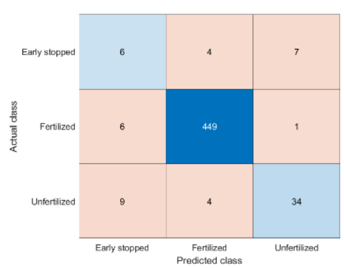

(e) EfficientNetb0:0.95



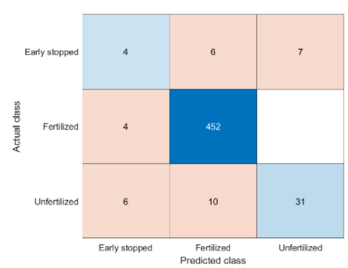

(f) GoogLeNet:0.96923



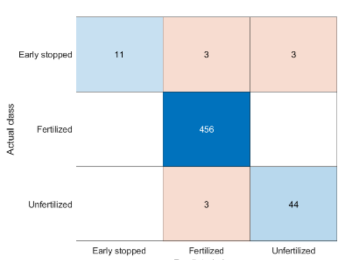

(g) Inceptionv3:0.98654



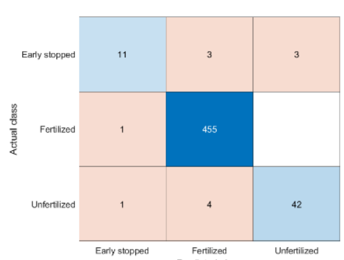

(i) ResNet18:0.94038



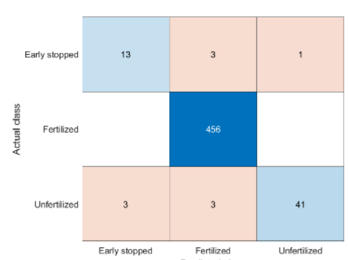

(h) ResNet50:0.93654




(j) AlexNet:0.98269

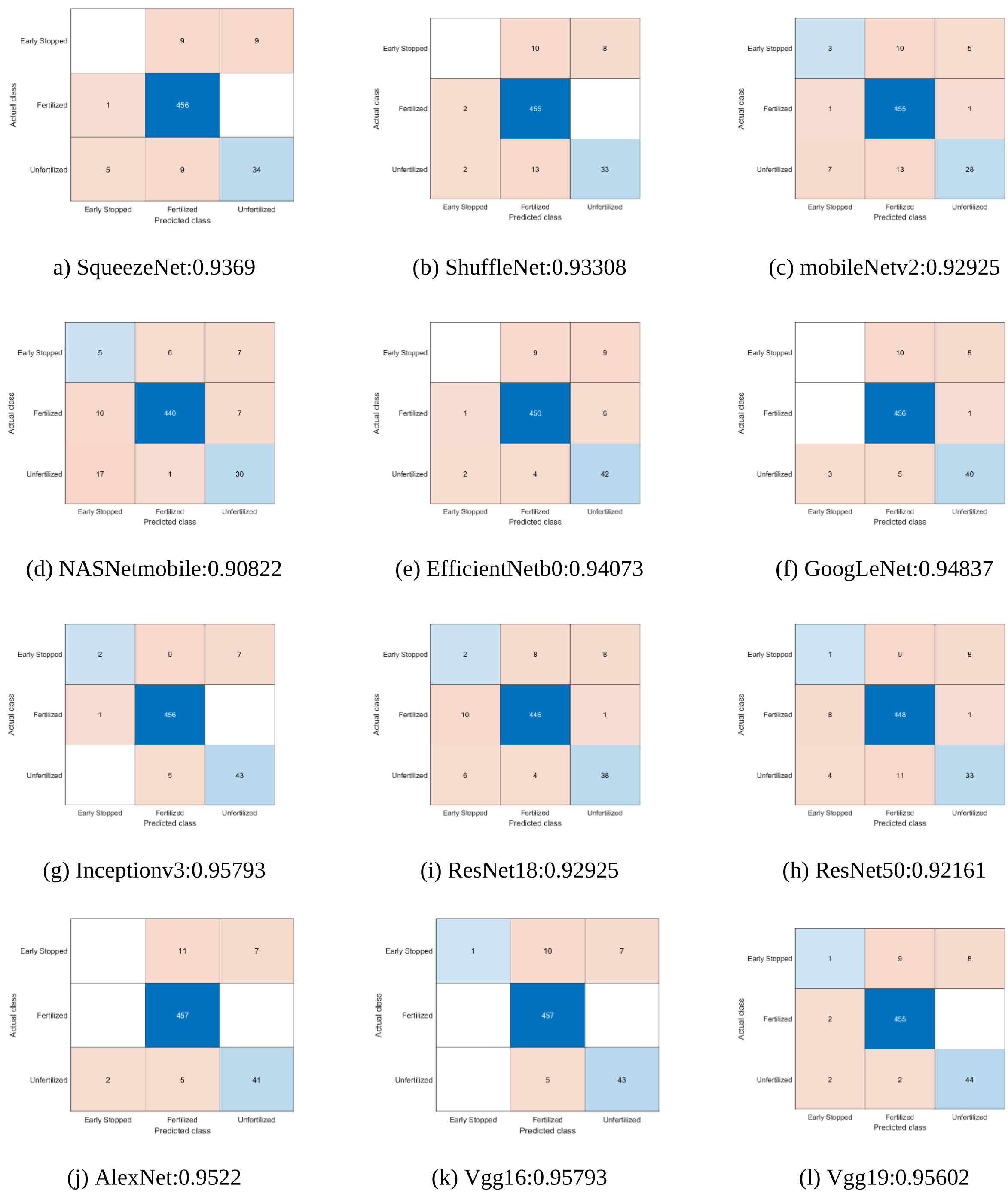



(k) Vgg16:0.97692




(l) Vgg19:0.98077

Figure 7. The Accuracy of CNNs for Case A



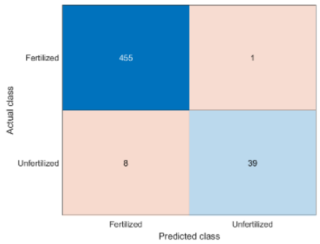
a) SqueezeNet:0.9369 (b) ShuffleNet:0.93308 (c) mobileNetv2:0.92925

(d) NASNetmobile:0.90822 (e) EfficientNetb0:0.94073 (f) GoogLeNet:0.94837

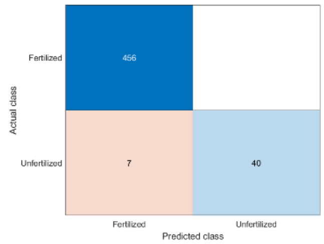
(g) Inceptionv3:0.95793 (i) ResNet18:0.92925 (h) ResNet50:0.92161

(j) AlexNet:0.9522 (k) Vgg16:0.95793 (l) Vgg19:0.95602

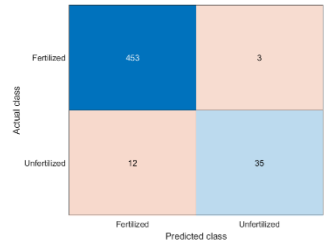
Figure 8. The Accuracy of CNNs for Case B



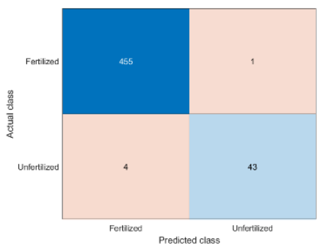

a) SqueezeNet:0.98211



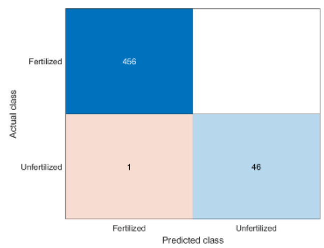

(b) ShuffleNet:0.98608



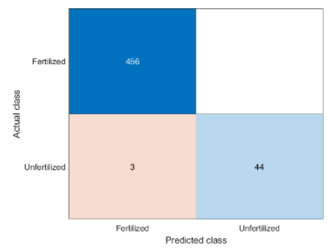

(c) mobileNetv2:0.97018



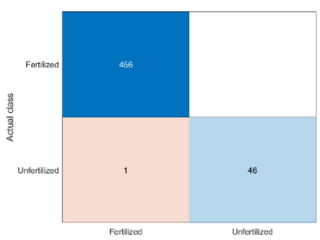

(d) NASNetmobile:0.99006



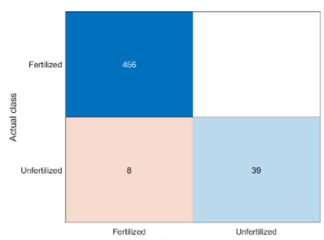

(e) EfficientNetb0:0.99801



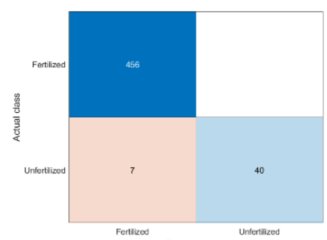

(f) GoogLeNet:0.99404



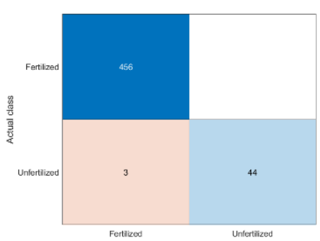

(g) Inceptionv3:0.99801



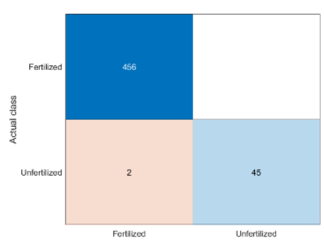

(i) ResNet18:0.9841



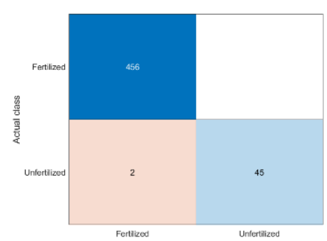

(h) ResNet50:0.98608




(j) AlexNet:0.99404

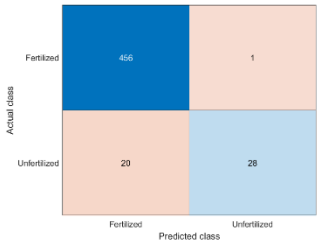



(k) Vgg16:0.99602

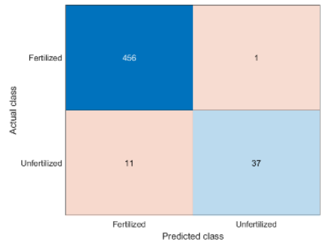



(l) Vgg19:0.99602

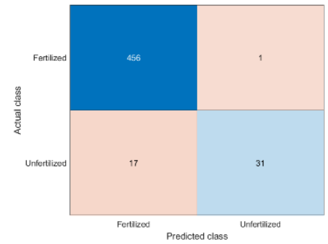
Figure 9. The Accuracy of CNNs for Case C



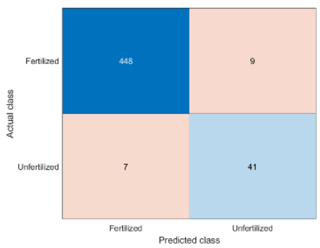

a) SqueezeNet:0.95842



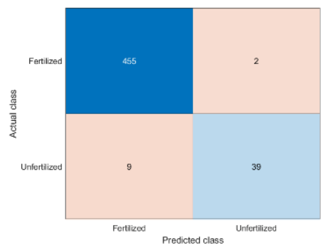

(b) ShuffleNet:0.97624



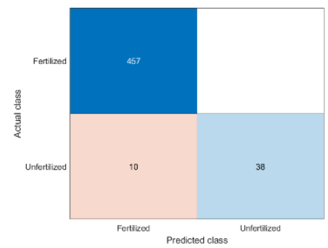

(c) mobileNetv2:0.96436



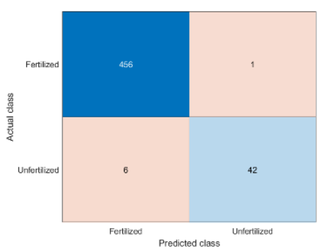

(d) NASNetmobile:0.96832



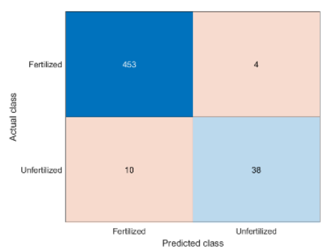

(e) EfficientNetb0:0.97822



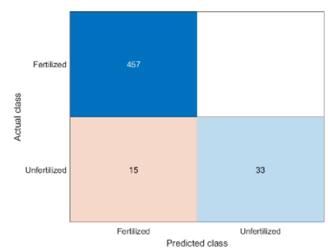

(f) GoogLeNet:0.9802



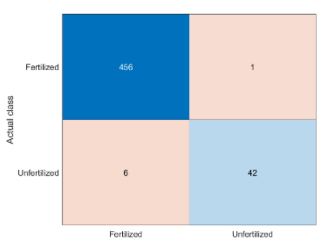

(g) Inceptionv3:0.98614



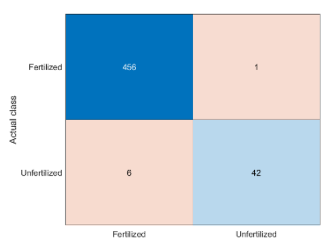

(i) ResNet18:0.97228



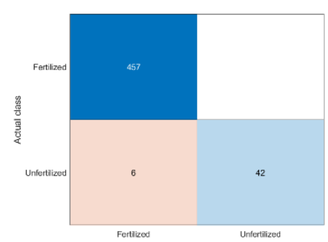

(h) ResNet50:0.9703




(j) AlexNet:0.98614




(k) Vgg16:0.98614




(l) Vgg19:0.98812

Figure 10. The Accuracy of CNNs for Case D