Duck Egg Quality Classification Based on its Shell Visual Property Through Transfer Learning Using ResNet-50

**JV Bryan DP. Caguioa, Ryhle Nodnylson E. Guinto,
Lee Reuben T. Mesias and Joel C. De Goma**
School of Information Technology
Mapua University
Makati, Metro Manila
caguioajvbryan07@gmail.com, ryhlenodnylson@gmail.com,
leemesi39@gmail.com, jcdegoma@mapua.edu.ph

Abstract

Duck egg quality inspections are done manually in the Philippines which is not reliable due to human subjectivity, visual stress, and tiredness. There is no documentation regarding the general standard on determining the quality of duck eggs and local farmers have different standards. This research aims to classify duck egg quality into 3 classes namely, Balut/Penoy, Salted Egg, and Table egg. Two angles of 600 duck eggs were captured inside an image acquisition setup. Using ResNet-50 as a base model, its last fully connected layer was replaced by a classifier block. Hyperparameter tuning with Stratified 5-fold Cross Validation was utilized. It was observed that batch size of 8, epoch of 110, and learning rate of 0.0001 has given the lowest validation loss which was used to train the final model. Performance Metrics was obtained. Overall, the result of the model yielded 90%, 95%, and 65% for Balut/Penoy, Salted Egg, and Table egg, respectively, averaging an 83.33% overall accuracy of the model. As observed, the model is not able to accurately differentiate between hairline and broken cracks. Additionally, falsely classified images also occur when the size of an egg is close to the threshold to other sizes.

Keywords
Duck egg quality, ResNet-50, K-fold CV, hyperparameter tuning

1. Introduction

1.1 Poultry Eggs

Poultry farming is very vital since eggs are common ingredients in the food we eat. From breakfast, baking, and main courses, eggs became an important ingredient in cooking. Farmers can classify egg quality by looking at its extrinsic and intrinsic features. Farmers can simply determine egg quality by checking its egg shell (Rachmawanto et al. 2020). The egg processing procedure includes collecting, washing, grading, and packaging. Washing and packaging became less of a trouble since it is commonly automated nowadays, but grading is still done manually. In this step, eggs are inspected if there are defects such as stains, blood spots, and even cracks (Beyaz et al. 2020).

Hauver and Hamann (2000) enumerated the exterior factors, namely, egg shape, texture, soundness, and cleanliness of the shell. These variables can be determined without a light source. A normal egg texture can be defined as an egg with a smooth texture, with no sound or little ridges. The soundness of the shell may be defined as the flawlessness of an egg, or whether the shell has damage or not. The cleanliness of the egg is also one of the vital features in determining egg quality. A shell that is unbroken and full of foreign material residing on 1/32 of the shell's surface (1/16 if scattered) is defined as dirty. On the other hand, a shell can be considered clean if it has only very small specks, strains, or cage marks, if such specks, stains, or cage marks are not of sufficient number or intensity to detract from the generally clean appearance of the egg. Egg shell abnormalities may occur and can be caused by diseases, environment, and nutrition.

1.2 Egg Inspection
The inspection and storing system is done manually and it is not reliable since it has several disadvantages, mainly because of human subjectivity, visual stress, and tiredness (Beyaz et al. 2020). A need for early separation for defective and cracked eggs as they may contaminate clean eggs and it degrades things it had contact with (Garcia-Alegre et al. 2000, Lunadei et al. 2011). Manually checking each egg requires great care for the farmer to avoid breaking and mishandling of eggs. But most of all, it would be very impractical for farmers to manually check a large number of eggs per day, as it would become costly in terms of labor (Quilloy et al. 2018).

Currently, there are no recorded documents nor standards when it comes to determining the quality of duck eggs produced by the Philippine Mallard Duck (Anas Luzonica). Moreover, local farms in the Philippines have different standards in classifying and segregating these eggs. As for the farmers, they only base their classification according to their feelings only, whether the eggs are considered good or rejected (Selvaraju et al. 2019).

1.3 The Problem, Gap or Opportunity
The studies are only conducted using chicken eggs as its dataset provides us the opportunity to localize it and to utilize our local duck eggs as our dataset (Rachmawanto et al. 2020, Garcia-Alegre et al. 2000, Lunadei et al. 2011, Shimizu et al. 2017). Currently there is no available official standard procedure provided in the Philippines in assessing the Philippine Mallard Duck egg quality, thus this aims to develop a model based on standard practices from the duck egg industry. The size of the duck egg is one of the criteria when determining how they will be processed for further production utilized in terms of production. The studies provide opportunities in detecting the egg quality to create a dataset in which the classes should contain striking features to differentiate from the others in order to avoid misclassification and to further improve the accuracy, find a more suitable feature extraction, segmentation process and test with another classifier (Rachmawanto et al. 2020, Garcia-Alegre et al. 2000, Lunadei et al. 2011).

1.4 Objectives
In order to broaden studies involving egg quality detection, this research develops a model for detecting egg quality through classification of visual features presented in its eggshell. This study aims to identify which characteristics of the egg would affect the classification accuracy of the model. The research will also verify the proposed model if it yields acceptable test accuracy and loss and correctly classifies the eggs. Finally, this study aims to identify the optimal hyperparameters that yield the best validation accuracy and loss.

This research also aims to localize egg quality classification and standardized duck egg quality assessment by using the Philippine Mallard Duck eggs. Using the resulting model which is a non-intrusive model through image processing in detecting egg quality, it can be utilized by other researchers who would also like to develop an automated system using computer vision for real-time detection of duck egg quality. In order to encourage the application of deep learning, specifically the Convolutional Neural Network algorithm, to contribute to studies involving the fields of agriculture, and to help improve other studies pertaining to duck eggs.

2. Literature Review

2.1 Egg Quality
Quality may be defined as the inherent properties of a product that determine its degree of excellence. Those conditions and characteristics that consumers want, and are willing to pay for, are, in a broad sense, factors of quality (Hauver and Hamann 2000). In determining Egg quality, it can be classified into two general groups, namely, External and Internal properties. External properties of an egg can be observed through its appearance which is considered as the initial assessment of its quality and some examples of it are cracks and dirty spots found on the eggshell (Nasiri et al. 2020).

2.1.1 Clean Eggshell
An eggshell that is free from stains, different foreign materials, and discoloration that are visible can be considered clean. Eggshells with minimal specks, stains, and cage marks can also be considered clean if the amount does not qualify for the generally clean appearance of an egg (Hauver and Hamann 2000). According to the USDA, Freedom from stains and foreign material on the shell must be considered in assigning a quality designation to an individual egg. In the paper (Abbaspour-Gilandeh et al. 2018), eggshells without contamination and blood spots are considered clean.

2.1.2 Cracked Eggshell

According to Chukwuka et al. (2018), defects that are under eggshell integrity are gross cracks, hairline cracks, star cracks, and thin-shelled or shell-less eggs. As cracked eggs cannot be made available for retail sale. Dirty eggs are very harmful as they can bring bacteria and other diseases that can spread to other eggs and cause spoilage if it gets inside the egg (Hauver and Hamann 2000). Cracks are the easiest defect that can be detected. Arivazhagan et al. (2013) stated that eggshell cracks include, gross cracks which refer to large cracks and holes that results in a broken shell membrane, hairline cracks, i.e. very fine cracks, usually run lengthwise along with the shell and they are difficult to detect and star cracks are fine cracks radiating outwards from a central point of impact, which is often slightly indented.

2.1.3 Dirty Eggshell

Beyaz et al. (2020) developed an experimental system in order to differentiate clean eggs and dirty eggs by defining dirt stains on eggs as mainly composed of feces, can be black to brown stains, uric acid which is white stains, yolk and blood. Lunadei et al. (2001) developed an offline artificial vision system in order to classify clean and dirty or defected eggshell using multispectral images. Their model identified defected eggshells as those with organic residuals on the surface such as blood, feathers and feces while clean eggshells are only presented with natural stains.

2.1.4 Egg Size

Thipakorn et al. (2016) used a candling technique with a light source placed behind the egg in order to gather the egg's parameters. A linear regression analysis is done in order to predict the weight. Results showed a strong correlation between the predicted and real weight of the eggs. Support Vector Algorithm was used to classify an unknown egg by its size. Grid-search Algorithm was used to obtain the optimal model features. A 10-fold-cross-validation technique was utilized for training the model and testing the results.

Waranusast et al. (2017) utilized an Android phone in gathering the dataset. The pictures of the eggs were taken in a controlled environment (on a white paper) from different angles. A coin was used in order to determine how relative the size of the coin is to the egg, given the size of the coin is known. In order to detect the coin, Histogram of Oriented Gradients (HOG) object detection algorithm was used to detect the coin within the image. The GrabCut Algorithm is used to segment and calculate the pixels around the center. The coin will be the basis of the egg. The circumference of the egg will be measured and will be compared to the coin.

2.2 Deep Learning

Kamilaris et al. (2018) showed that deep learning had entered the domain of agriculture, the majority of which deals with image classification and identification that offers better performance compared to the traditional image processing techniques as it provides an advantage in terms of reduced need of feature engineering, existing models and frameworks instead of building from the start. Deep learning needs a large dataset but in return provides ability to extract and learn features automatically, to be retrained with new dataset which gives a more accurate result in vision-based tasks such as object detection, classification, and segmentation but it is sensitive to changes such as lighting, pose, background and other image subjects (Nasiri et al. 2020).

2.3 Grad-CAM

Convolutional Neural Networks have their own way of identifying and deciding the features that make their decision. In order to create a visualization, Selvaraju et al. (2019) created visual explanations for decisions that the CNN model considers when predicting a class. Their study is a combination of Grad-CAM and Guided Grad-CAM and to apply it to projects such as image classification, image captioning, visual question answering models, including ResNet architectures.

2.4 Related Works

Nasiri et al. (2020) used image acquisition setup which includes an illumination box to provide appropriate lighting conditions, roller conveyor in order to rotate eggs, Samsung CCD camera in order to capture a photo, and personal computer in order to run the model and classify which egg belongs. Through image processing which includes image rotation and zoom, height and width shift, horizontal-flip and shear intensity is applied to existing images which is then resized to 224x224x3 to create new images for the training dataset. They created a modified VGG-16 architecture in order to classify unwashed eggs images based on 3 classes: intact, bloody, broken and use transfer learning by using pre-trained weights from ImageNet in order to provide train weights to use in the dataset instead of starting with random weights.

Shimizu et al. (2018) applied Convolutional Neural Network in small dataset in order to classify the quality of industrial products specifically eggs, they found out that the criterion in labelling egg quality varies from each human inspector specially in good egg and poor growth egg are hard to classify due to its label ambiguity which also inherited by the model. A captured an image of an egg from 4 angles to retain its feature from 3-dimensionality which shows that it improves the accuracy of the model. The images from 4 different angles are combined into 1 image and then resized by 400x400 pixels to serve as an input image. The CNN used is based on AlexNet and the last fully connected layer reduces the feature vector to six which serves as the output for the 6 qualities of the egg.

García-Alegre et al (2000) also created an artificial vision system for brown and beige eggs with the same features and addition of cracks from Lunadei et al. (2011). Results show that misclassified detection can be derived from the eggshell having a small unique dark defect or a slight uniform spot distributed all over the eggs and the difference in color provides different intensity of reflection between the defect and the background.

Rachmawanto et al. (2020) tried to classify egg quality which they grouped into three classes: good quality, rotten and defective eggs. Through their testing, they found out that defective class has the most accurate score due to it having more dominant features compared to good and rotten eggs which doesn't have any striking difference.

2.5 CNN Architecture

Subetha et al. (2020) created a comparative analysis between two deep learning algorithm's performance namely ResNet50 and VGG19, in classifying the apple leaf diseases. The dataset used contained images of apple orchids with different backgrounds, illumination, and noises which are classified into four classes: scab, healthy, multiple diseases, and rust. The researchers determined that VGG19 performed better than ResNet50 in training and validation accuracy while ResNet50 performed better than VGG19 in prediction accuracy. Both of these architectures are equal in terms of overall accuracy but the VGG19 have more computational complexity compared to ResNet50.

3. Methods

3.1 Image Preparation

Figure 1 depicts how the image is preprocessed. The images were originally 4223x4223 in dimensions. Then those images were cropped to 3600x3600. Cropping was done so that the images were removed of unnecessary pixels and the focus will be centered on the egg. The eggs will be concatenated vertically. Normalizing the pixel value of all the images to 224x224x3 was done in order to fit the dataset in the model. The data augmentation process creates a modified version of an image which would increase the amount of data for training.

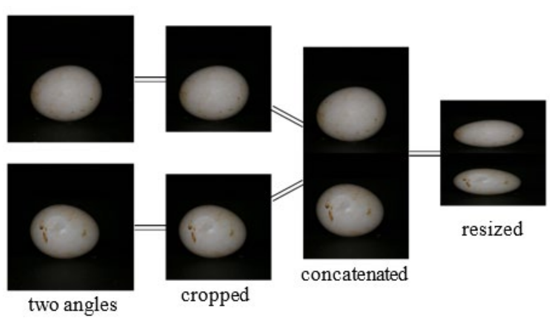


Figure 1. Preparation of images before augmentation.

The CNN model contains a high number of parameters thus it requires a large dataset to learn all weights, parameters and to also avoid overfitting (Nasiri et al. 2020). Kamilaris et. al. (2018) mentions that data augmentation is an important process in training deep learning models as it helps improve its performance and learning process. We have utilized Keras' data augmentation in order to train our model to learn from vast variations of an image. The data augmentation process involves 7 transformations of images coming from our original image. The first process would be ResNet50's preprocessing input. In Figure 2, the images were converted from RGB to BGR channels. The preprocessed images have the 50% chance to flip horizontally, same chances to flip vertically. Due to some images having different positions in the picture, we have shifted the width and height of the image in order to account for the different positions of the eggs in the image with a range of - 10px to 10px. Rotation range was used with a range of 0 to 3. It was used because some egg images were not aligned after the concatenation process. This enabled our images to have uniform orientations. Lastly, the images' brightness was altered to create images with variations of brightness which can highlight some features.

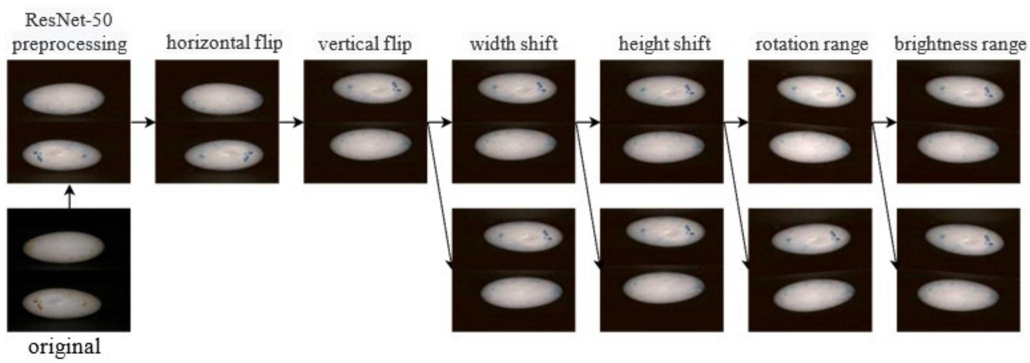


Figure 2. Data Augmentation Process

3.2 Model Architecture
Neural networks became popular as they can handle huge amounts of data. One of the most customary Artificial Neural Networks being used in Deep learning is the Convolutional Neural Network (CNN). This includes multiple layers such as convolutional layer, non-linearity layer, pooling layer and a fully connected layer. Convolutional Neural networks work well especially in dealing with images as data (Albawi et al. 2017). The convolutional layer, as the core of these networks, has the task of hierarchical extraction of nonlinear features of an image. This can also reduce the dimension of the inputs to remove unnecessary features. The pooling layer serves as a reduction tool for dimensions of the feature map in order to lessen the number of unnecessary parameters and computation time and thus to control overfitting (Nasiri et al. 2020).

Residual Network 50, commonly known as Resnet50, is a Residual Network variant which contains 48 convolutional layers, 1 max pool layer, and 1 average pool layer. The max pool layer computes the largest value that highlights the most present feature in the patch. The average pooling layer computes the average values in the patch. This architecture takes outputs of images as 256 dimensions, and a good point of this architecture is its skip connection feature and contains stacked layers, which solves the vanishing gradient descent problem. This architecture is a shallow neural network that utilizes residual blocks. These layers are trained to learn a residual function. In classical neural networks, the identity shortcut is not implemented. This makes the block learn the input directly. Although this architecture is shallower compared to other Neural Networks, it does not contain a lot of parameters since it utilizes an identity shortcut (He et al. 2015). In order to classify the given classes, the base ResNet50 model was modified by replacing the last fully connected layer with our classifier block to identify our own classes, namely Balut/Penoy, Salted Egg, and Table Egg. The classifier block is composed of a global average pooling layer which minimizes the number of parameters thus minimizing overfitting, dense layers with 512 and 256 neurons and ReLu activation function which carries out the mathematical operation, a batch normalization layers to standardize all inputs, dropout layer as a regulation layer to also reduce overfitting and the output layer that contains 3 neurons with softmax classifier to compute the normalized probability of three classes of the egg (Nasiri et al. 2020).

3.3 Model Setup
Nasiri et al. (2020) defined pre-training as giving the CNN pre-trained weights instead of random weights as it speeds up the training process and studies shown using pre-trained weights specifically on ImageNet provide much better accuracy than those trained with random parameters. In this study, the ResNet-50 model was using pre-trained weights from the ImageNet and was frozen during training except for the last layer. The last layer of the ResNet-50 model was removed and replaced with a classifier that will predict the classes (Balut/Penoy, Salted Egg, and Table Egg).

3.4 Model Training Process

To evaluate the model's performance, the dataset was split into train and test sets. The 600 concatenated images (200 for each class) were used as the dataset. It was split into a train and test set having 540 and 60 images respectively. A random state was used. The train set utilized 90% of the original dataset and used it for model training. The train set was further split into train and validation sets (80:20) as it was used to train the model with Stratified 5-fold cross validation. Stratified 5-fold cross-validation is a variation of cross-validation that evenly distributes the number of data on each class per fold. A random state was used. Data augmentation was applied for both train and validation sets in order to reduce the overfitting. The number of images generated per fold was equal to the amount of original train set multiplied by the number of epochs (540 x num of epoch). The train set was divided into 5 sets, 4 of those were used as training while 1 was served as a validation set. This process was iterated 5 times, each set being the validation set while others serve as the training set. The remaining 10% of our data was used as the test set for model evaluation. Using grid search, hyperparameter tuning was done in order to find which combination of values of batch size, epoch and learning rate will give the best performance. The optimal combination of hyperparameters was determined as the one with lowest mean validation loss and highest mean validation accuracy on cross-validation.

3.5 Stratified K-Folds Cross Validation

The best combination of hyperparameters was then used to train the final model. The train set was split into a train set and validation set with the same random state and ratio (80:20) used in the Stratified K-fold Cross Validation in the hyperparameter tuning. Prediction was done using the test set. The performance metrics were then measured. The model performance will be evaluated based on accuracy, sensitivity or recall, precision, and are under the curve (AUC). Figure 3 shows the whole process from hyperparameter tuning to training the final model.

3.6 Performance Metrics

To determine the performance of the model, the proponents will accuracy, precision, recall and Area under the curve (AUC) will be defined. Area under the Curve (AUC) defines the ability of the model to avoid misclassification (Taheri-Garavandad et al. 2015).

3.7 Gradient-weighted Class Activation Mapping (Grad-CAM)

Gradient-weighted Class Activation Mapping (Grad-CAM) was used to identify where the model focused when given an image. At the end of the last convolutional layer, it utilizes gradients of any target object which yields a coarse localization map that highlights where the model gets the important regions located in the picture for predictions.

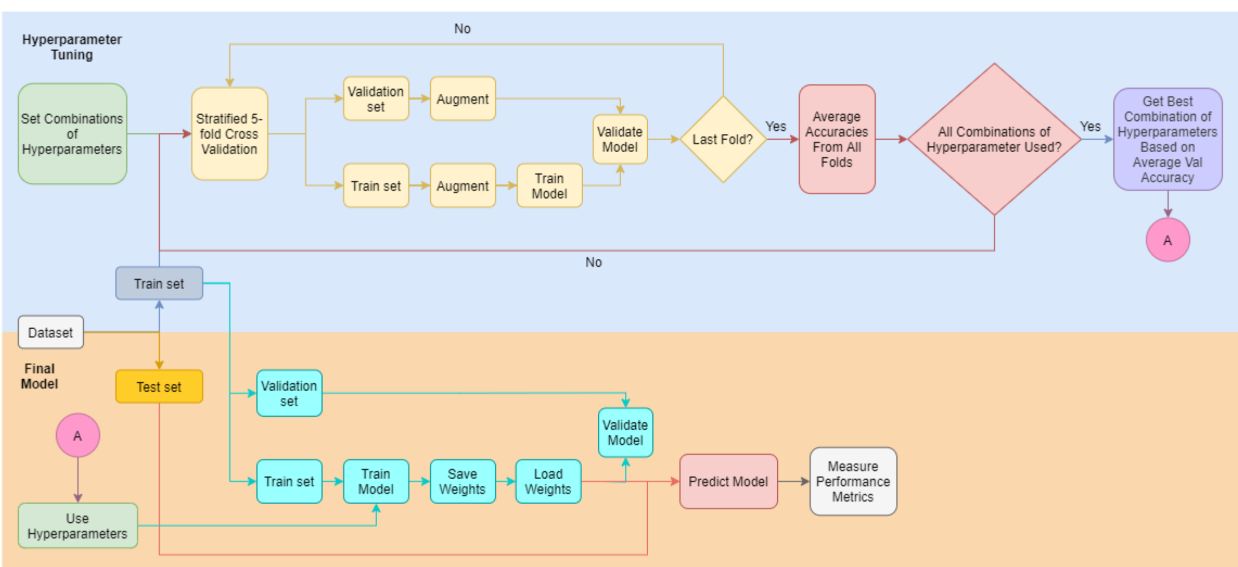


Figure 3. Process Flow Chart of Hyperparameter Tuning to Training the Final Model

4. Data Collection

4.1 Data Acquisition Setup

The proponents will set up an appropriate environment for capturing images. In figure 4, We have created a black box in which the egg will be captured. The egg is placed in the center of the blackbox in order to account for the size. The blackbox is made of illustration boards with a dimension of 22.86cm x 22.86cm x 7.68cm and the black side of it was used as the background of the environment in order to differentiate the object from the picture better. Inside, there is an average sized hole to keep the egg in place. On the outside, the box had one cut hole on each side to have room for the phone's rear-facing camera. A cover is placed on top of the box to prevent external light sources. All angles of the eggs were captured using a Huawei P30 Lite's 48MP rear-facing camera and its built-in flash to provide sufficient lighting. The camera settings were set to ISO 50 and shutter speed to 1/250 to provide the most detailed image and highlight the desired features.

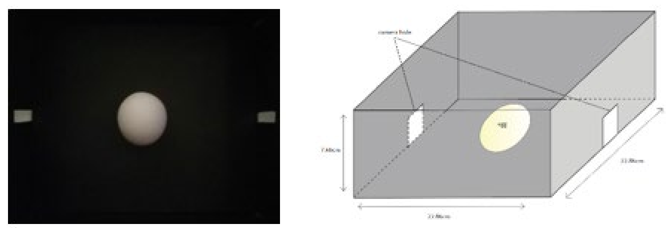


Figure 4. Duck Egg Image Acquisition Setup. Left (top view of setup), Right (initial draft).

The eggs are gathered from a local poultry farm. They are selected based on the manual annotations by the expert handling the eggs. Three classes are considered in the samples, namely, Balut/Penoy, Salted Egg, and Table Egg. A total of 600 eggs (200 for each class) are captured from both angles. Figure 5 shows the characteristics portrayed in each class. Eggs under Balut/Penoy are classified as eggs with little to no dirt. Eggs that are also large in size are considered under this category. Clean eggs contain scatter stains that are less than 1/16 or localized that are less than 1/32 of its shell. Eggs under Salted eggs are classified as eggs that are small-medium in size and contain localized dirt more than 1/32 of its shell. They may also contain hairline cracks, which are line-like cracks but the membrane is still intact. Eggs under Table eggs are classified as eggs that are peewee in size. They may also contain breakage which can be toe-punched or a completely damaged eggshell and membrane.

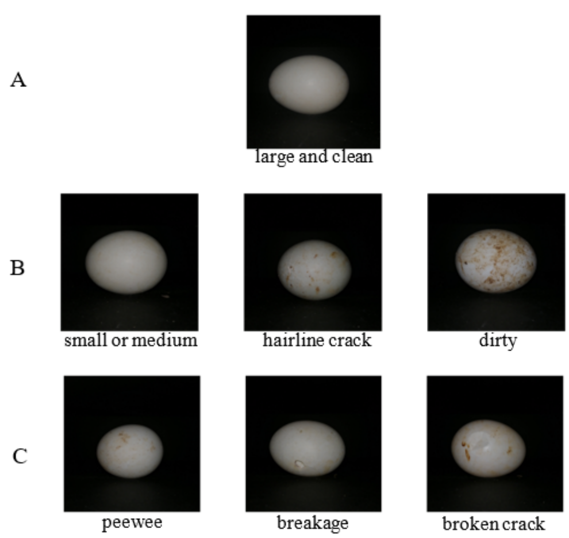


Figure 5. A) Balut Penoy, B) Salted Egg, C) Table Egg

5. Results and Discussion

5.1 Hyperparameter Tuning

To attain the best combination of hyperparameters for the model, hyperparameter tuning using grid search was done. Hyperparameters that have differing values include the batch size, number of epochs, and learning rate. A model checkpoint that monitors the validation loss and will save the weights of the model that has the lowest validation loss. Initial testing provides a baseline in which values will not result in excessive overfitting and underfitting. For batch size values 8, 16, 32 were used. For epoch values, 50, 70, 90, and 110 were used. And lastly, for learning rate values, 0.0001 and 0.00001 were used. The results of hyperparameter optimization include the means of training, validation accuracy and loss from stratified 5-fold cross validation which are sorted in descending order based on the validation accuracy as shown in Table 1. The hyperparameter combination of batch size, number of epochs and learning rate having the values 8, 110, and 0.0001, respectively, yielded the highest mean validation accuracy of 85% and lowest validation loss of 0.358 thus this is the optimal hyperparameter for the final model.

Table 1: Top-5 Results of hyperparameter Optimization based on Validation Loss

Batch Size	Epochs	Learning Rate	Mean Training		Mean Validation	
			Accuracy	Loss	Accuracy	Loss
8	110	0.0001	80.73	0.468	85	0.358
16	110	0.0001	83.13	0.411	84.81	0.373
8	90	0.0001	79.7	0.491	83.33	0.383
16	90	0.0001	81.93	0.439	84.63	0.389
16	50	0.0001	78.72	0.506	84.81	0.404
32	90	0.0001	82.42	0.426	83.89	0.405

Using the optimal hyperparameter, the proposed model was trained with a batch size of 8, 110 epochs with a learning rate of 0.0001. Using data augmentation, our model was trained to 59,400 images each fold. The accuracy and loss of training, validation and testing of each fold are shown in Table 2. The results of each fold were averaged and yielded a training, validation, and test accuracy of 80.73, 85% and 87.67% respectively. The loss for training, validation and test is 0.468, 0.358 and 0.292, respectively. These hyperparameters were used due to the fold training accuracies are almost similar. The validation accuracy yielded during training differs due to the way how the classes were distributed. Nevertheless, the validation accuracy is above the training accuracy, which means that the model was able to generalize. Among all folds, Fold 3 exhibited the highest training and validation accuracy and lowest validation loss, 81.00%, 88.89%, and 0.294, respectively.

Table 2: Results of the optimal hyperparameter per fold

Fold	Training		Validation	
	Accuracy	Loss	Accuracy	Loss
1	80.98	0.465	83.33	0.35
2	80.88	0.465	81.48	0.39
3	81.00	0.466	88.89	0.29
4	80.61	0.469	84.26	0.38
5	80.15	0.477	87.04	0.36
Average	80.73	0.468	85.00	0.35

5.2 Model Testing

We used the best hyperparameters in training the final model. The train set was split into a train set and validation set with the same random state used during the hyperparameter tuning. Data augmentation was also done for both the train set and validation set. It is clear that the model training showed little to no overfitting. The training and validation accuracy and loss are close to each other. This means that the final model was fitted perfectly and the data was generalized well. The resulting validation accuracy and loss were 84.26% and 0.50, respectively.

5.3 Model Evaluation

In Figure 6, it shows the result of the model's prediction on the three classes using the test set. There are 20 images per class, 60 images total in the test set. For Balut/Penoy, 18 out 20 Balut/Penoy eggs were classified correctly and misclassified 1 for each other class. For Salted Eggs, 19 out 20 Salted eggs were classified correctly and misclassified 1 on balut/penoy class. For Table Eggs, 13 out 20 Salted eggs were classified correctly and misclassified 7 on Salted Egg class.

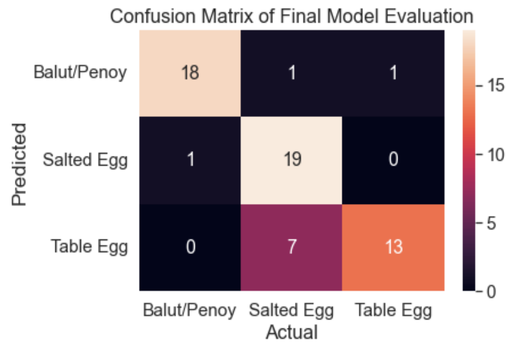


Figure 6: Final model test prediction on test set

Table 3 presents the statistical parameter of each class and its average to evaluate the whole proposed model by predicting the test set using the final model. The Salted Egg class attained the highest accuracy with 95% while the Balut/Penoy and Table Egg class achieved 90% and 65%, respectively. The mean score for accuracy, precision, and recall are 83.33%, 0.859881, and 0.83, respectively.

Table 3: Average statistical parameter of the proposed CNN model

Class	Accuracy	Precision	Recall
Balut/Penoy	90	94.74	90
Salted Egg	95	70.37	95
Table Egg	65	92.86	65
Average	83.33	85.99	83.33

5.4 Result Visualization using Grad-CAM

Figure 7.1 shows the correctly predicted eggs based on size. From these images, the model focuses on the eggs shell to determine the size, especially with Balut/Penoy eggs. The model also predicts the size of some eggs by focusing the corners of the image and calculating the pixels covered by the gradient in the background. Another factor for why the egg focuses on the middle of the egg is because the model also learned that Balut/Penoy eggs have clean surfaces.

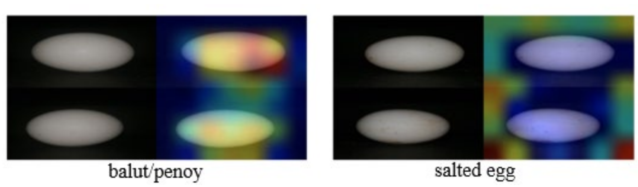
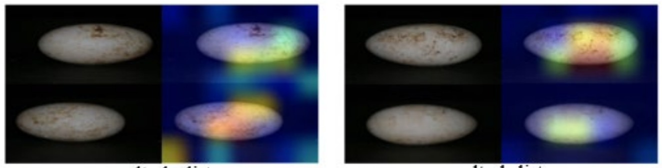


Figure 7.1: Correctly predicted base on egg size

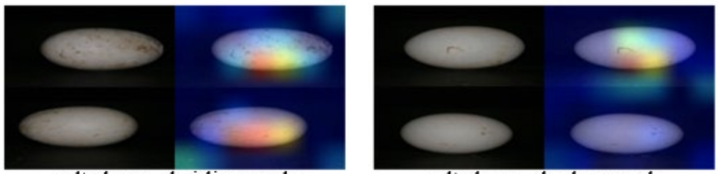
Figure 7.2 shows how the model sees eggs that are dirty. Eggs under this category are usually in Salted Eggs. The model analyzes the dirt easily by focusing on a specific part of the eggshell with impurities.



salted -dirty   salted-dirty

Figure 7.2: Correctly detected dirt on salted egg

In Figure 7.3, the eggs with hairline (salted eggs) and broken cracks (table eggs) were identified by the model by highlighting specific surfaces that have holes or visible lines.



salted egg – hairline crack   salted egg – broken crack

Figure 7.3: Correctly classified cracks on salted and table egg

Figure 7.4 shows the correctly classified eggs, but the model cannot see the correct features for the intended class. Specifically, for table eggs. The model was not able to focus on the egg's features. From the observations, images with low feature detection tend to be classified in table eggs. This might be the reason why there is low count of prediction in table egg class as seen on Figure 6.

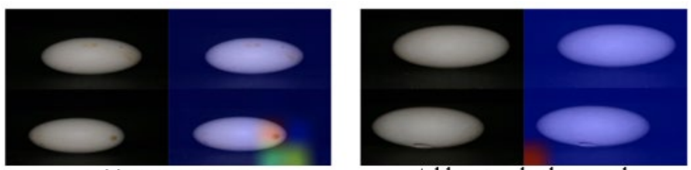
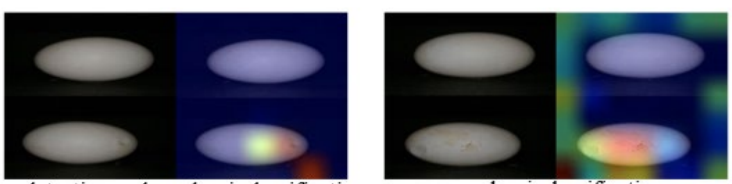


table egg - peewee   table egg – broken crack

Figure 7.4: Correctly classified table egg but low feature detection

Figure 7.5 shows the misclassified table eggs to salted eggs due to the model inability to accurately differentiate between the two kinds of cracks. Another factor is that low feature detection tends to be classified as table egg.



low detection and crack misclassification   crack misclassification

Figure 7.5: Misclassified table egg to salted egg due to low feature detection and crack misclassification

Figure 7.6 shows misclassification of eggs due to size. The model classification for size is having trouble classifying when the threshold for medium egg is close to large eggs, which confuses our model to determine whether it is large or medium. Same goes with misclassified table eggs, wherein table eggs that are close to the threshold of small are considered to be in that class.

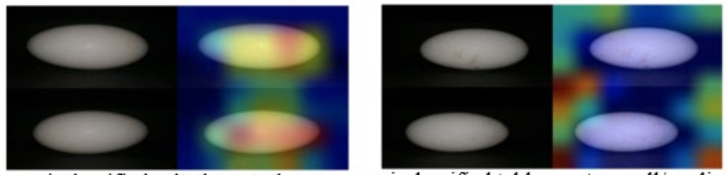


Figure 7.6: Misclassified classes due to size

Figure 7.7 shows images that are table eggs but classified as salted eggs due to some cracks not being detected due to the location of the crack. From this image, the breakage is located at the tip of the egg. The model cannot see that broken crack as it is, making the focus of the model scattered to different regions of the image. This problem might be resolved by training more data with cracks at the tip of the egg or finding ways to capture the entire surface of the egg.

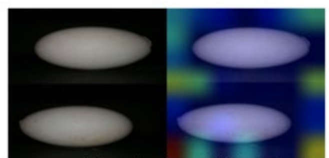


Figure 7.7: Misclassified table egg to salted egg due to crack on top of the egg not being detected

6. Conclusion

This research aims to classify the duck (Philippine Mallard) egg quality (Balut/Penoy, Salted Egg, Table Egg) using deep learning. The researchers have gathered 600 eggs (200 for each class) and captured its images to serve as the dataset. Using ResNet-50 as a base model, its last fully connected layer was replaced by a classifier block that will enable the model to predict the 3 classes. The block consists of a Global Average Pooling Layer, two Dense Layers, Batch Normalization Layer, and a Dropout Layer. During training, stratified 5-fold validation was used to serve as a validation. Through grid search, hyperparameter tuning was performed and concluded that a batch size, epoch, and learning rate of 8, 110, 0.0001, respectively, provided our model an optimal performance. Overall, the result of the model yielded 90%, 95%, and 65% for Balut/Penoy, Salted Egg, and Table egg, respectively, averaging an 83.33% overall accuracy of the model. As observed, the model is not able to accurately differentiate between hairline and broken cracks. Additionally, falsely classified images also occur when the size of the egg is close to the threshold of the succeeding and preceding size.

There are notable gaps in our research that can be improved. For future researchers, we would recommend classifying the duck egg with its size first before other features. It is preferred to do a comparison of the proposed model with the traditional computer vision models and feature selection techniques in order to improve classification accuracy. To find a solution on how to capture the entire surface of the egg. Increase the dataset to have more generalized data and determine if it will yield better results.

References

Abbaspour-Gilandeh, Y. And Azizi, A., Yumurta Kabuğundaki Çatlakların Bilgisayar Görüntüsü ve Hough Dönüşümü Kullanılarak Tanımlanması. *Yüzüncü Yıl Üniversitesi Tarım Bilimleri Dergisi*, pp.375-383, 2018.

Albawi, S., Mohammed, T. and Al-Zawi, S., Understanding of a convolutional neural network, *2017 International Conference on Engineering and Technology (ICET)*, 2017.

Arivazhagan, S., Shebiah, R., Sudharsan, H., Kannan, R. and Ramesh, R., External and Internal Defect Detection of Egg using Machine Vision, *Journal of Emerging Trends in Computing and Information Sciences*, 2013.

Beyaz, A., Özlü, S. and Gerdan, D., Experimental Recognition System for Dirty Eggshell by Using Image Analysis Technique. *Turkish Journal of Agriculture - Food Science and Technology*, 8(5), pp.1122-1126, 2020.

Chukwuka, O., Okoli, I., Okeudo, N., Udedibie, A., Ogbuewu, I., Aladi, N., Iheshiulor, O. and Omede, A., Egg Quality Defects in Poultry Management and Food Safety, *Asian Journal of Agricultural Research*, 5, pp.1-16, 2010.

Garcia-Alegre, M., Ribeiro, A., Guinea, D. and Cristobal, G., Eggshell defects detection based on color processing, *Machine Vision Applications in Industrial Inspection VIII*, 2000.

Hauver, W. and Hamann, J., *Egg grading manual*, Washington, D.C.: U.S. Dept. of Agriculture, Agricultural Marketing Service, 2000.

He, K., Zhang, X., Ren, S. and Sun, J., Deep Residual Network for Image Recognition, 2015.

Kamilaris, A. and Prenafeta-Boldú, F., Deep learning in agriculture: A survey, *Computers and Electronics in Agriculture*, 147, pp.70-90, 2018.

Lunadei, L., Ruiz-Garcia, L., Bodria, L. and Guidetti, R., Automatic Identification of Defects on Eggshell Through a Multispectral Vision System, *Food and Bioprocess Technology*, 5(8), pp.3042-3050, 2011.

Nasiri, A., Omid, M. and Taheri-Garavand, A., An automatic sorting system for unwashed eggs using deep learning, *Journal of Food Engineering*, 283, p.110036, 2020.

Quilloy, E., Delfin, S. and Pepito, B., Single-line automated sorter using mechatronics and machine vision system for Philippine table eggs, *African Journal of Agricultural Research*, 13(17), pp.918-926, 2018.

Rachmawanto, E., Atika Sari, C., Villadelfiya, R., Ignatius Moses Setiadi, D., Rijati, N., Kartikadarma, E., Doheir, M. and Astuti, S., Eggs Classification based on Egg Shell Image using K-Nearest Neighbors Classifier, *2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, 2020.

Selvaraju, R., Cogswell, M., Das, A., Vedantam, R., Parikh, D. and Batra, D., Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization, *International Journal of Computer Vision*, 128(2), pp.336-359, 2019.

Shimizu, R., Yanagawa, S., Shimizu, T., Hamada, M. and Kuroda, T., Convolutional neural network for industrial egg classification, *2017 International SoC Design Conference (ISOCC)*, 2017.

Subetha, T., Khilar, R. and Subaja Christo, M., A comparative analysis on plant pathology classification using deep learning architecture – Resnet and VGG19, *Materials Today: Proceedings*, 2021.

Taheri-Garavand, A., Ahmadi, H., Omid, M., Mohtasebi, S., Mollazade, K., Russell Smith, A. and Carlomagno, G., An intelligent approach for cooling radiator fault diagnosis based on infrared thermal image processing technique, *Applied Thermal Engineering*, 87, pp.434-443, 2015.

Thipakorn, J., Waranusast, R. and Riyamongkol, P., Egg weight prediction and egg size classification using image processing and machine learning, *2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, 2017.

Waranusast, R., Intayod, P. and Makhod, D., Egg size classification on Android mobile devices using image processing and machine learning, *2016 Fifth ICT International Student Project Conference (ICT-ISPC)*, 2016.

Biographies

JV Bryan DP. Caguioa is an undergraduate student at Mapua University which is currently at his final year waiting for graduation on Bachelor of Science in Computer Science. He is a receiver of president's list award during his first semester and fourth semester on school year 2019 - 2020. He has done his internship at Chimes Consulting in which he was trained as a Software and Website Tester and planned to pursue software engineering as a career. His interest involves automation on menial tasks to large projects through application of artificial intelligence.

Ryhle Nodnyhlson E. Guinto studies at Mapua University, taking a Bachelor of Science in Computer Science. He specializes in the field of Artificial Intelligence. He is interested working with Data, specifically Data Analytics. It is currently his final year and waiting for his graduation. He also plans to pursue a career in Business Analytics. He has experience in Data visualization and other BI tools.

Lee Reuben T. Mesias is a Bachelor of Science Computer Science student of Mapua University at his final year waiting for graduation. He became an engineering student intern in Analog Devices, Inc. He takes interest on solving problems and wants to create applications that tracks the flow of data in the internet. He pursues to work outside the country in order to work in a stable job.

Joel C. De Goma is a PhD in Computer Student of Mapua University. He specializes in Artificial Intelligence and Data Science