

# From Farm to Fork: Transforming Egg Quality and Boosting Export Potential using Artificial Intelligence on Poultry Farming

Bogathi Madhusudharsan Reddy<sup>1</sup>; Nikhil Pradip Parsawar<sup>2</sup>; Amarender Reddy Gundumalla<sup>3</sup>; Indira Kumar<sup>4</sup>; Bharani Kumar Depuru<sup>5\*</sup>

<sup>1,2</sup>Research Associate; <sup>3</sup>Mentor; <sup>4</sup>Team Leader; <sup>5</sup>CEO and Managing Director  
<sup>1,2,3,4,5</sup>AiSRPY, Hyderabad, India

Corresponding Author: Bharani Kumar Depuru<sup>5\*</sup>  
ORC ID: 0009-0003-4338-8914

**Abstract:-** Nothing can spoil fresh eggs that were laid by chickens, like dirty vent feathers, blood strains and minor cracks. Eggs play an important role in the context of human health due to its cheap cost and accessibility across the globe. But egg damage paves the way for a significant size of a shortage of nutrients, and creates demand which leads to an increase in price. The brand name of the company that supplies the eggs loses its reputation when damaged eggs reach the market along with good eggs and this can also lead to business loss. India is considered to be one of the worst farm poultry industries across the world, because of poor maintenance of farm, lack of cleanliness and non-compliance to the safety standards. Just because of this, the Indian market loses its power of negotiation on price and loses the opportunity to export the eggs to foreign countries even though India's egg production is close to the egg production margin of other countries. Not only from the perspective of commerce but also from the view of health this can lead to adverse consequences when the unclean eggs are consumed by the public or people who have less immunity.

Every year a significant number of people are suffering from diseases when the eggs, which are poorly handled in the process of quality control, are consumed. This could also lead to a fixation of mindset to the consumer that these are usual eggs in India. The consequence of this mindset could result in more severe effects in consumerism over the period of time. This might also prevent people from purchasing the eggs for consumption. The major reason for this is unhygienic rearing practice and improper quality control measures. These are open invitations for egg contamination although India is the third largest country in the world in the production of eggs. There are more than forty seven million egg producers who do not meet safety standards, most of the produce are rejected for export due to chemicals used during the rearing process.

Consumption of these eggs had led to a serious issue in India. A sizable number of people are resistant to antibiotics just because of the consumption of eggs that were produced using chemicals and eggs in which

chemicals were intact on the shell of that egg. An article that was published by a renowned newspaper states that consumption of such contaminated eggs, which had heavy metal, has led to poisoning in children, paving the way for mental development problems, sometimes mental retardation. This study has attempted to reduce the severe consequences which are a result of consuming eggs with dirt or damaged eggs. A huge effort has been made to ensure that this scenario changes over the period of time, by incorporating AI in the field of Poultry industry in order to enhance the quality of egg production. The study we conducted clearly shows that most of the poor rearing practice happens due to human error. This can be changed by replacing the human efforts by machine, we could replace the entire quality control process especially removing the spoiled eggs from the batch that is to be sent to the market.

The model that we tried to build has given marvelous results in separating the spoiled from the good eggs. The algorithm will not only identify the spoiled eggs but also stores the final output in the database where the Organisations can access it at any point of time. The AI Object detection model will detect the cracked eggs, the eggs with dirt, the eggs with blood stain and many more. In nutshell our object detection model will detect all abnormal eggs which will help us prevent the abnormal eggs from reaching the consumer.

**Keywords:-** Egg Detection, Crack Detection, Egg Classification, YOLOv9, AI in Poultry Industry, Damage Detection, Egg Strain Detection.

## I. INTRODUCTION

Egg producers pour lots of money into their business to produce good quality eggs. But all of their investments become useless expenses when the desired quality standards are not met. This study attempts to save the lives of the public by incorporation of the AI powered object detection [14] model into the production line, which will help them separate the faulty eggs from intact eggs. This will prevent the faulty eggs from reaching the market and this will secure the life of

millions of people. The incorporation of object detection and tracking models will minimise the human error that occurs due to the carelessness of humans. This will also minimise the manpower required to classify the eggs and quality

control. We leverage pre-built models such as YOLO models, RESNET, MOBILENET and many others to achieve the goal.

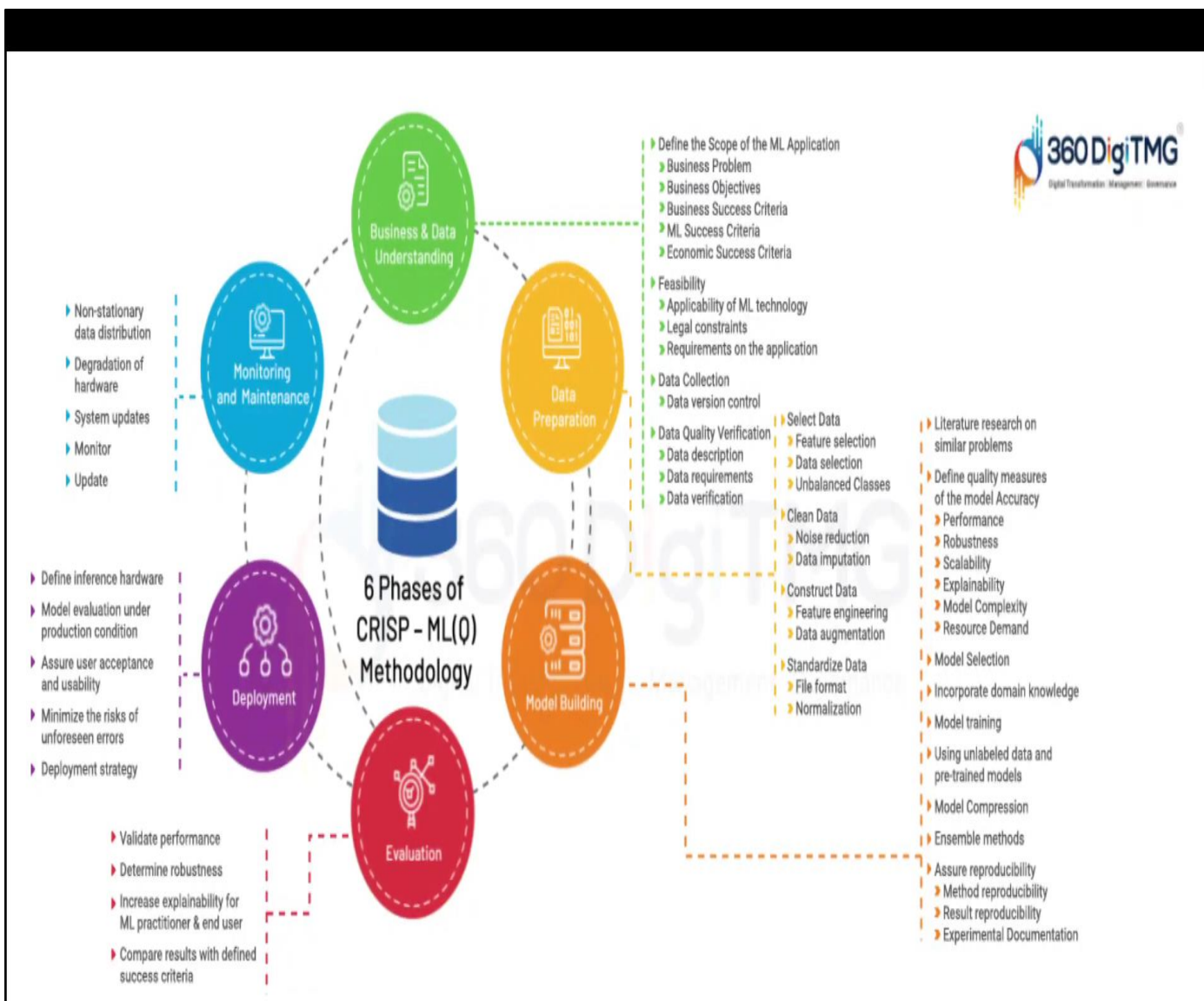


Fig 1: CRISP - ML(Q) - Project Methodology

We have adapted CRISP-ML(Q) methodology [Fig.1] that included six stages of process, namely, Business understanding, Data Understanding, Data Preparation, Model Building, Model Evaluation, Model Deployment and finally Monitoring & Maintenance. The enormous efforts that we have invested in this study has shown convincing results which have been discussed in the upcoming sections. The process begins from collection of data which were captured from a high quality camera with extraordinary frame rate. And then data preparation which involves highly customised data augmentation steps such as resizing, cropping, adding

noise, blurring, and many more to make our detection better. The prepared dataset is passed into the model for training purposes, where all required features are extracted using the convolution technique. We have used various types of evaluation techniques, for instance mean average precision, Intersection Over Union, Confusion matrix, precision and recall, to ensure that our model does the good job. Variety of deployment methods, such as flask, docker, kubernetes, AWS EBS and Streamlit, were implemented and tested to choose the best. [Fig.2] talks about the overall flow of the project.

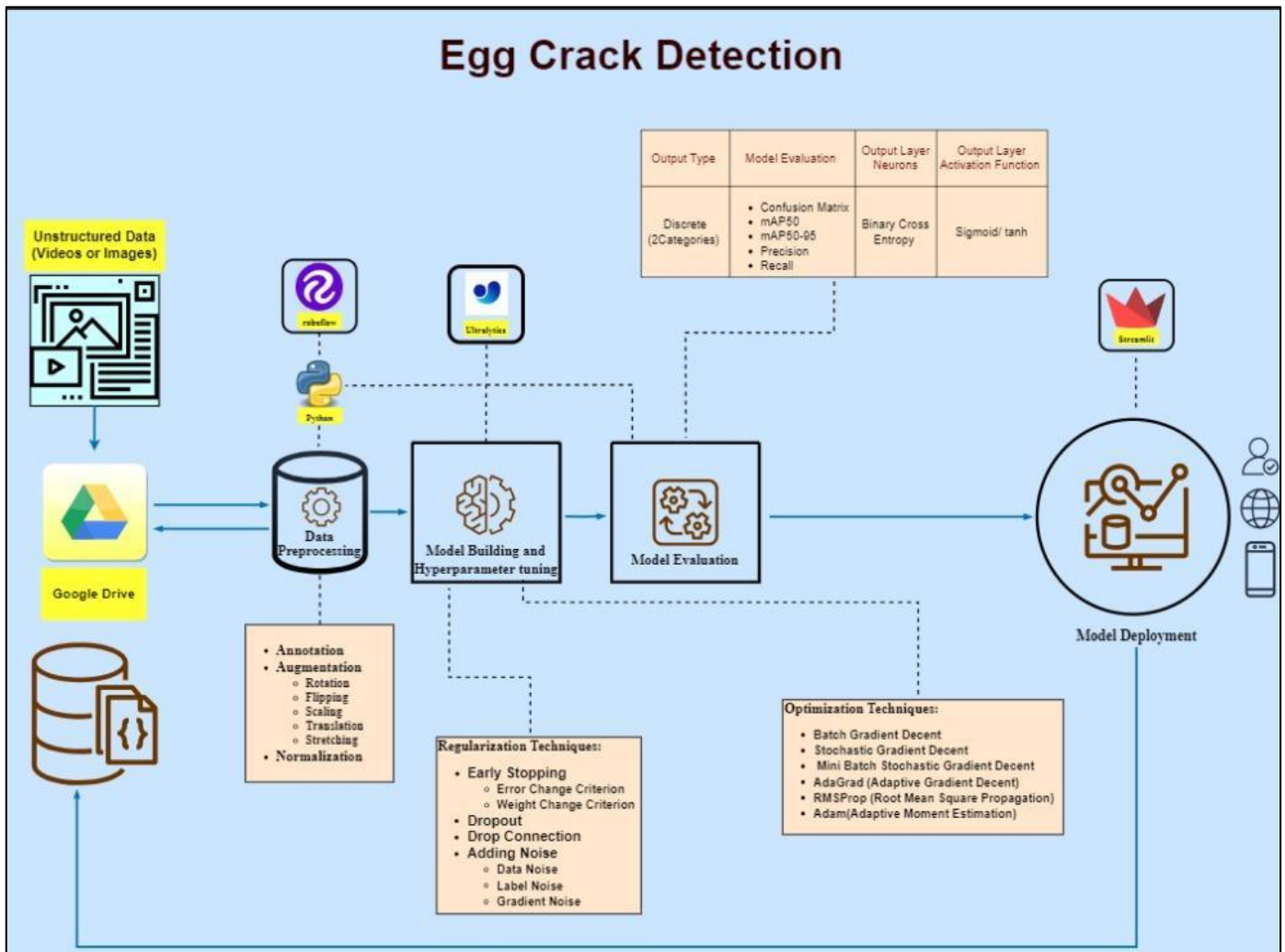


Fig 2: Project Architecture Diagram

## II. BUSINESS UNDERSTANDING

According to the National Family Health Survey (NFHS) reports for 2019–2020, 74% of Indian households consume eggs. This statistical information shows the consumption of eggs across various parts of India; it shows eggs are one of the major food items. Between the years 2022 and 2023, egg production in India increased to 8.78 billion eggs. Such a rise in production shows the growing demand for eggs, and the concerned efforts of the poultry are necessary to meet the demand.

Because of the physical construction of eggshells and the many ways they travel during industrial production and distribution, eggs frequently sustain impacts that result in eggshell cracks[3,9,13]. Another visual feature brought on by chicken poop contamination of the eggshell is dirt. Dirt on the eggshell is a common occurrence in medium-sized companies, particularly in nations where regulations do not mandate producers to adhere to specific hygiene standards. However, this is not a characteristic of completely industrialized operations. Dirt and cracks increase the likelihood that dangerous germs, such as salmonella, will enter and compromise the product's quality[9]. A strict

quality control procedure is necessary since consumers depend on food goods like eggs for their quality.

However, this high volume of demand and the above problems make it challenging to maintain the high quality of egg production[4]. Manual separation of cracked and dirty eggs is time-consuming and highly susceptible to errors[1,8].

To overcome this challenge, one viable solution is to automate the process using computer vision technology[1,6,7,8]. Computer vision technology will extract the information from the images and videos for real-time projects such as classification and reorganizing[5]. When it comes to classification, a model can be trained to identify and categorize an object or an object's physical characteristics in an image[5]. When a model encounters an image that it hasn't seen before, its performance is determined by how reliable the dataset used for training is[5].

Implementing the automation process can significantly increase efficiency, accuracy, and quality assurance during the procedure. The uses of automation in the egg sorting process extend beyond operational efficiencies due to the fact that customers gain high-quality eggs that meet the standards of freshness and integrity[2].

This study follows 1) Preparing the data set, which consists of images of eggs with a black background, varying lighting, depth, angle, and side of the egg shown, will ensure the model trains in all the situations[5]. 2) As the next step, the gathered data will be uploaded to Roboflow, where it undergoes thorough preprocessing to ensure it's optimized for your machine learning tasks. 3) Build multi-class classification models based on convolutional neural network (CNN) models such as YoLov8, YoLov9, RCNN, LeNet, AlexNet, and MobileNet. These models classify crack and dirty eggs from intact eggs. 4) After the model building is complete, the models will be evaluated using various metrics, which is a crucial step in the machine learning or deep neural learning workflow to assess their performance and determine which algorithm works well for this specific task.

### III. DATA UNDERSTANDING

The data is collected by using a primary data source and a secondary data source. In the primary data source, the data is collected in an experimental setup. The eggs are collected from the local forms, market, and village hen forms as the test sample[10]. The egg picture is taken from three different smartphone rear cameras, and the images are formatted in.jpg format[1,5]. The models and pixelation of the camera are as follows: Redmi Note 11, 50MP main camera along with that 8MP ultrawide lens, 2MP macro lens, and 2MP depth lens; PoCo f5 Pro, 64MP along with that 8MP ultra-wide lens, 2MP macro camera; Oppo A78, 50MP main camera along with that 2MP in-depth camera. For every egg, five to ten images were captured, varying the background, lighting, camera distance (depth), and side of the egg visible to the camera. The dataset was prepared based on the physical attributes: eggs with free cracks are labelled as two classes because they separate the brown and white eggs, and eggs with cracks and dirty eggs are grouped as two classes with separation of brown and white. It should be mentioned that the owner deemed the harvested eggs to be clean, the dataset was captured during the day, and the weather was sunny. During the rainy season, dirty eggs are frequently discovered. Since there was still a visible difference on the egg's surface, a compromise was reached so that the eggs with filthy spots were recorded as cracked or dirty for the dataset. All the samples of egg images are shown in [Fig.4]

From secondary data sources, it will be collated from open data sources. In secondary data sources, the irrelevant data will be removed manually to meet the relevant requirements. The total number of images collected was 8321, of which white intact eggs are 2127, brown intact eggs are 2093, white cracked dirty eggs are 2069, and brown cracked dirty eggs are 2032. All the data collected from primary and secondary sources is uploaded to Roboflow for annotation, labelling, and preprocessing. The data collected from the secondary source is very limited because the data needs to represent the real-time scenario. In our real-time scenario, the background should be black, so it is not as needed as from the secondary source. All the collected images either secondary or primary data source is coloured images only.

### IV. DATA PREPARATION

The gathered dataset from primary and secondary sources is uploaded to Roboflow, a platform built to make dealing with video and images datasets easier and more convenient. Roboflow allows the user to upload, put together, and control their image and video's datasets in one place, resulting in working with large amounts of image and longer videos. Data from various sources is made simple and provides annotation tools for labeling objects of interest within images, such as bounding boxes, polygons, to enhance normal machine learning (ML) algorithms and deep neural learning model performance. Additionally, it provides integrated dataset annotation and augmentation methods, including color manipulation, flipping, scaling, and rotation, to enhance the machine Learning (ML) and Deep neural learning model's performance and generalized results by exposing it to a wider range of variables and scenarios. Users can integrate annotated datasets with popular frameworks like Tensor Flow, PyTorch, and YOLO, enabling direct training and deployment of custom computer vision models.

The uploaded image data into Roboflow underwent manual annotation by rectangular bounding boxes to identify and label specific objects or features in this study four distinct labels were assigned to classify the various eggs[1,6,7].

- White Cracked Dirty Egg: This label was given to eggs of white colour with cracks or damage and dirty on the surface.
- White Intact Egg: The label was applied to white eggs that were free of harm.
- Brown Cracked Dirty Egg: This label was given to eggs of brown colour with cracks or damage and dirty on the surface.
- Brown Intact Egg: The label was applied to brown eggs that were free of harm.

The careful annotation of the images with these four labels was essential for helping and guiding algorithms to recognize and categorize eggs according to their colour and state. With precise and detailed annotations, the algorithms could concentrate on identifying the unique visual traits linked to every label, making it easier to accurately identify and categorize eggs in the following phases of analysis and model training.

After the annotation created the versions doing the version there is options to apply the all the preprocessing steps like resizing, auto-orientation, Modify the class names, Filter Null images it ensures that all the images should contain the annotation, and augmentation to the dataset[6, 10, 11]. Increase the model's robustness and diversity, increase model generalization, and reduce overfitting.

- The first Preprocessing method used is resizing the images to make them fit into a 640x640 space. It maintains uniformity in the sizes of the images, which makes processing and standardization for model training more effective. During training, resizing can also help in

lowering demand for memory and computational complexity.

- The next preprocessing applied is auto orientation of the images to ensure constant alignment. It corrects any variations in image orientation, such as sideways or upside-down orientations.
- Filter Null is applied to the data set to extract only annotated images. This improves recall while lowering false negatives. 100% indicates that filter all null annotations, and 0% indicates to keep all null annotations.
- The next preprocessing step added to the dataset is auto adjusted contrasted to enhance line detection and normalization under different lighting conditions.
- Augmentation: The process of modifying or improving data, usually by adding things like flipping, rotating, and brightness, to expand and differ the dataset for more effective machine learning (ML) model training[3].
- Flipping: Flipped the images horizontally and vertically so it doubles the dataset. This augmentation introduces

variation in the orientation of objects within the images; it improves the model to generalize across different orientations[3].

- Rotation: Rotating the images by 90 degrees, both clockwise and anticlockwise, introduces the model to learning at different angles and positions[3].
- Brightness Adjustment: adjusted the brightness conditions by -10% to +10%; this makes the model more robust at different lighting conditions.
- Noise: 0.1% of the noise is added to the data set, representing imperfections that can pose significant challenges for machine comprehension compared to human understanding. These minor pixel alterations imperceptible to humans can profoundly impact a neural network's accuracy in making predictions.
- Blur: The 1px of gaussian blur is added to the dataset, it builds the model to be more resilient to camera focus.
- [Fig.3] talks about image labelling tool.

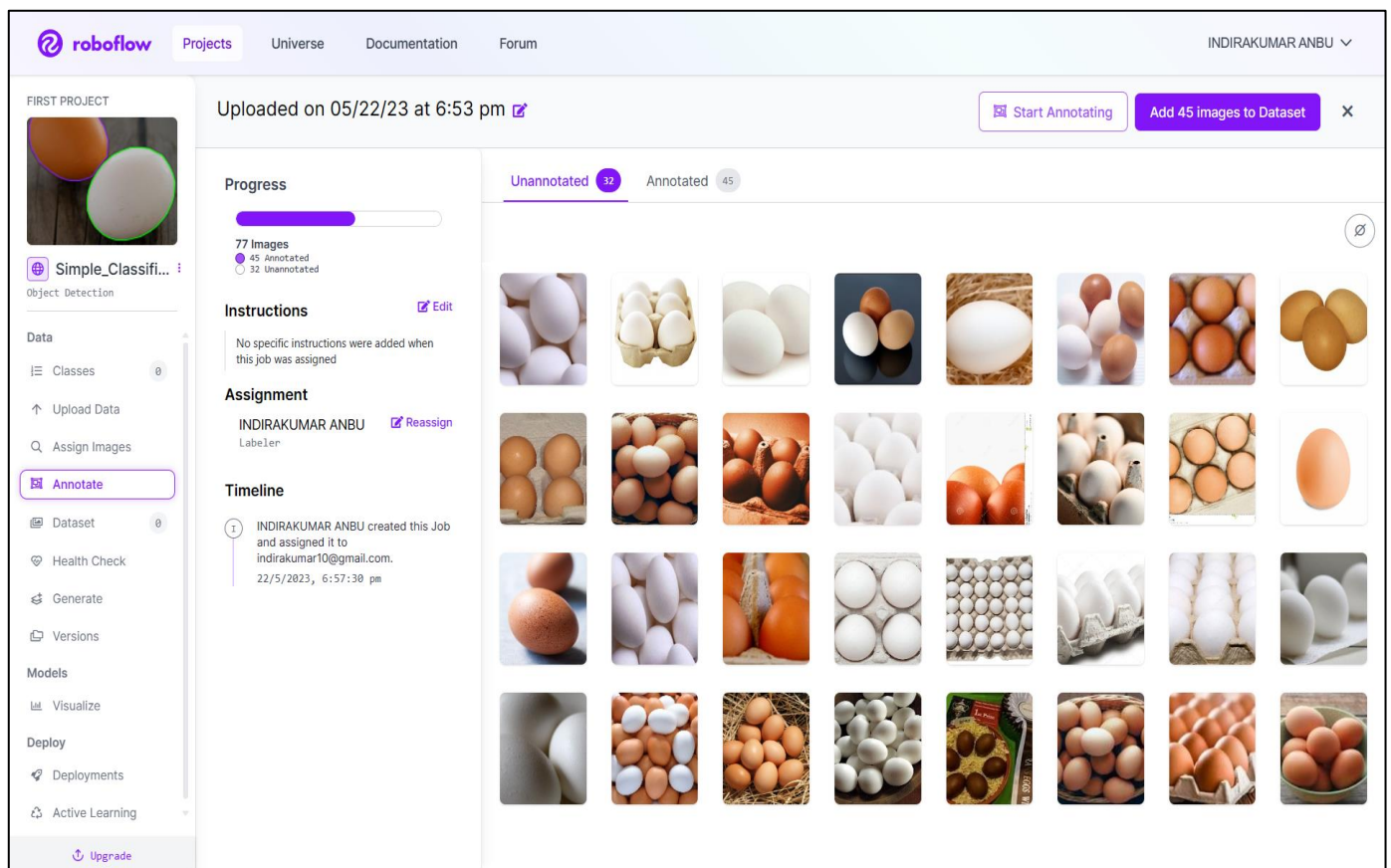


Fig. 3: Image Labelling Tool

The dataset underwent substantial growth upon augmentation, going from 8321 photos at launch to 33235 images in total. The dataset's diversity and richness were significantly enhanced through this augmentation method, offering a more extensive and diversified set of examples for machine learning (ML) and Deep neural learning model training.

In Roboflow, the datasets can be exported by various formats to suit different machine learning (ML) and Deep neural learning frameworks and projects. Some of the common export formats in Roboflow is.

- JSON: This exporting format enables compatibility with frameworks such as TensorFlow Object Detection API and Detectron.
- TXT: The txt files format is suitable for all the Yolo models; it is a state-of-the-art model for object detection.

- CSV(Comma-Separated-Values): It is a simple and widely used text-based format for tabular data that allows easy integration with various machine learning (ML) models.
- XML: In this only one format is available, that is Pascal VOC makes it easier to integrate with OpenCV and Darknet/YOLO frameworks.

Following the Roboflow export format selection. The dataset will be transformed by the Roboflow into a format that is compatible with the chosen export format. This could entail transforming the metadata, labels, and annotations on images into the necessary structures and schema that the selected format specifies.[Fig.4] talks about the Damaged egg and intact egg.

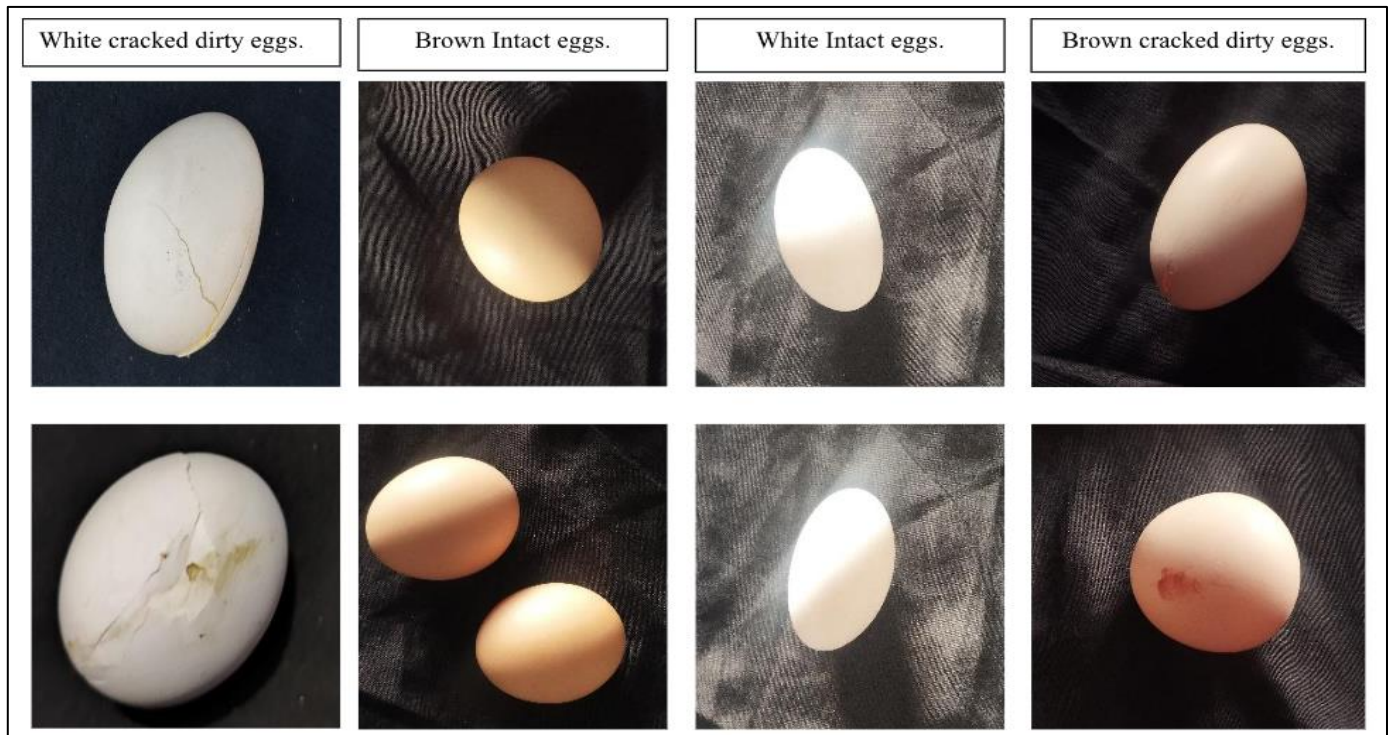


Fig 4: Damaged Eggs and Intact Eggs

## V. MODEL BUILDING

Model building is one of the important stage of CRISP ML(Q) we tried various types of model design to perform operations in computer vision our task is object detection there are multiple models available for object detection and we implemented on some of the model architectures like yolov5, yolov8 ,resnet ,mobilenet yolov9 and faster RCNN.

### A. YOLOV5

It is from the Yolo family's 5th version of Yolo. It has features like it can do real-time object detection and it is well known for its accuracy and speed[9].

### B. YOLOV8

It is the 8th version of yolo architecture. It is much faster than yolov5 in terms of accuracy and speed is better than the previous versions. It has a backbone of cspdarknet53 also it has a self-attention mechanism on top of the head.

### C. Mobile Net

Mobilenet has introduced a global parameters width multiplier and resolution multiplier this architecture works in a way first depthwise convolution layer then batch normalization next layer is the activation function relay again

one convolution layer followed by the activation function relay [3,4].

### D. RESNET:

RESNET is a cnn type of architecture for deep learning the name RESNET is a residual network which is the residual block of the model architecture it is used to train a deep neural network that has more than 100 layers[3,4,11].

### E. Faster -RCNN:

Faster RCNN is introduced when CNN is not able to perform well on multi-objects.Faster RCNN works like it first extracts the region proposals from the input image, then computes the CNN features and gives output. the selective search algorithm used to reduce the region proposals from RCNN[6,11].

### F. YOLO V9

It is the latest state-of-the-art (SOTA) model of the Yolo family. YOLOv9 is a combination of Programmable Gradient Information (PGI) and the Generalized Efficient Layer Aggregation Network (GELAN). it has 10% fewer parameters and 5-15% fewer calculations than previous models but still, it achieved an improvement of 0.5 to 0.6 % improvement in average precision.[Fig.5] talks about the PGI Architecture and [Fig.6] talks about the Glean Architecture.

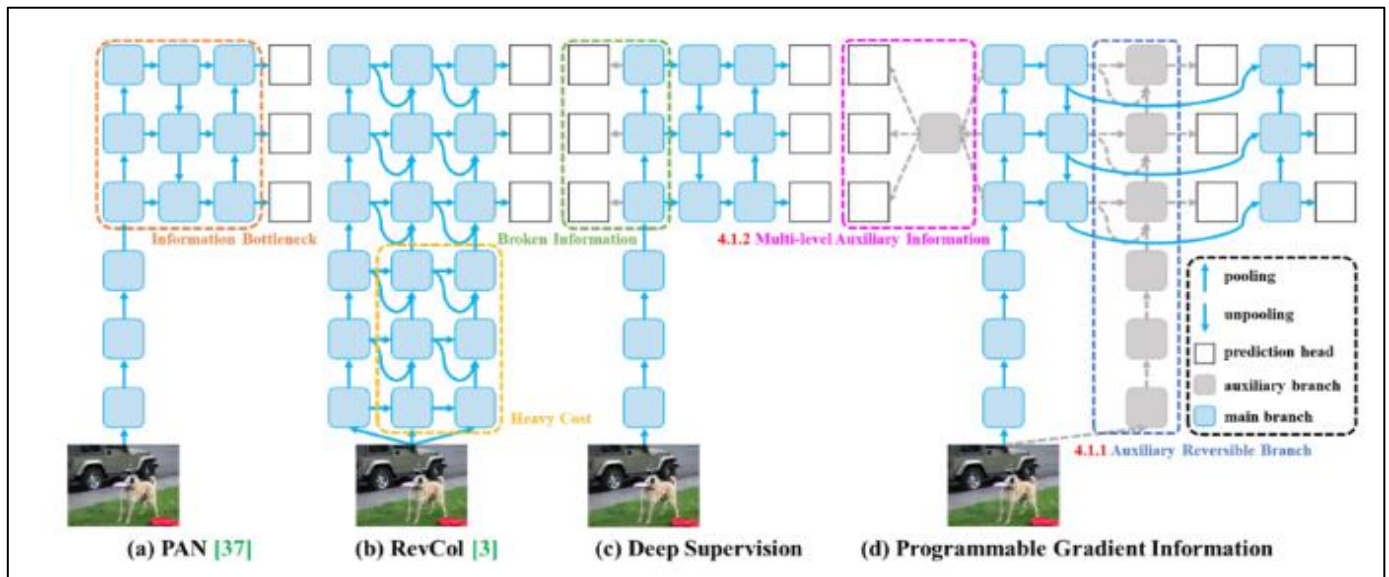


Fig 5: PGI Architecture

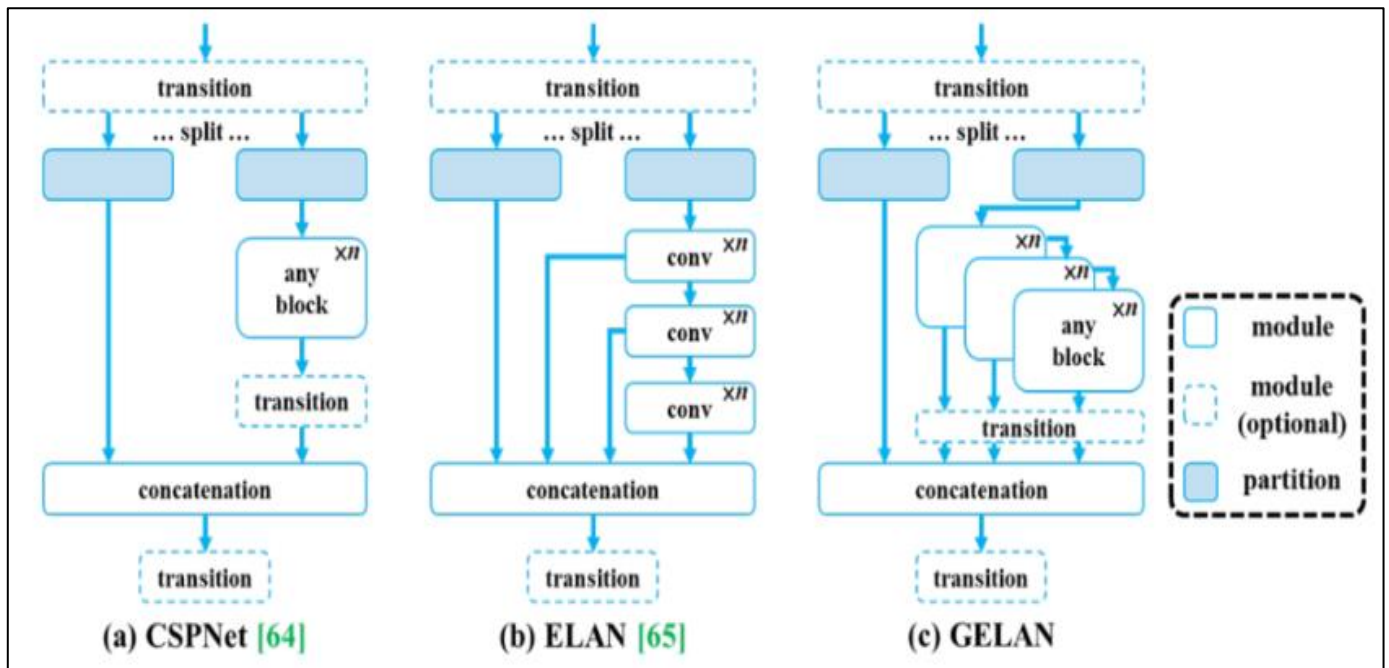


Fig 6: Glean Architecture

After evolving all the model's performance we decided to go with the Yolo v9 architecture because of its features, its MAP score is higher than every other model and its speed and accuracy, performance, easy of use, and versatility are major factors to choose Yolo v9.

**G. What is Convolution?**

Convolution layer is like a tool used by computer to understand pictures it's significant because it helps the computer pick out important details and shapes from the pictures imagine you have a special magnifying glass called a filter that you use to look closely at different parts of the picture you then multiply what you see in that area by numbers on the filter and add up all those results this gives you one number that tells you something significant about what region of the picture that's basically what convolution

layer does. It helps computers recognize pictures by looking at them closely and picking out important features.[1,10]

**H. What is Feature Extraction?**

Feature extraction is a process of gathering relative information(data) from the images. There are different image features of images such as color, shape, size, texture, location, orientation, depth etc.it learns the pattern understanding by its features. These features are used for further image processing.

- How features of an image are extracted in the convolution layer?
- Before jumping into how the extraction happens, the basic terminologies should be clear.

**I. Kernel:**

Kernel is nothing but a filter that is used to extract the features from the images. the kernel is a matrix that moves over the input data, performs dot product with the subregion of input data and gets the output as a summation of that. The kernel moves on the input data by stride value[10].

**J. Stride:**

Stride is the filter that moves across the image from left to right, top to bottom with one pixel column change on horizontal movements. The one-pixel row change on vertical movements is almost always symmetrical in height and width dimensions default dimension (1,1)[10].

**K. Padding:**

Padding is the best approach, where the number of pixels needed for the convolution kernels to process the edge pixel are added onto the outside, copying the pixel from the edge of the image we use padding. padding used to fix border effect problems. We can preserve our information using padding[10].

**L. Pooling:**

Pooling is required to downsample the detection of features in feature map. Pooling layers provides an approach to downsampling feature map by summarising the presence of features in patches of the feature map. two common methods one is average pooling and max pooling that summarize the average presence of the feature and most activated presence of the feature respectively.

**M. Flatten layer:**

A flattening layer is a once the pooled feature map obtained the next step is to flatten it. It involves transforming the entire pooled feature map matrix into a single column which is then fed to the neural network for processing.

First, we took the input image on which we applied kernel filter on that we get the feature map and we applied next pooling on that and repeated the process that we define how much iteration we want, then after getting pooled feature map finally last step is flattening the layer in one dimension that input is our extracted features. That is how the we extract the image features in the convolution layer[1].

**➤ Activation Function used in YOLOv9**

Activation function is an internal state of neuron used to convert input signal on node of neural network to an output signal.

**➤ ReLU:**

ReLU widely used activation function. It is used in CNN for task like image detection, recognition due to its computational efficiency it calculates only active neurons that's why it is faster than other activation functions. It is used in YOLO v9.

$$f(z) = \max(0, x)$$

In ReLU if the input is less than 0 that means output is zero and the input is greater than 0 it is the same value as an output. We used ReLU in backpropagation algorithms.

**➤ Leaky ReLU**

It is an upgraded variation of ReLU activation function. This Leaky ReLU helps in preventing dead neurons in ReLU it is useful in deep learning models where avoiding the dying ReLU problem is critical. It has a small positive slope in negative area.

$$f(z) = \max(0.1x, x)$$

Leaky ReLU is the same as ReLU it does back propagation, even for negative input values.

**➤ Difficulties in Training the Object Detection Models**

Google Colab is a platform where we implemented our model. It is hosted by Jupyter Notebook. It provides free access to computing resources. Including GPU and TPU. For model building, we followed the transfer learning approach. We used a pre-trained YOLOv9.yaml file to train our model. It contains a few parameters, number of classes, activation function, depth multiplier, and width multiplier. The backbone and head architecture is in that YOLOv9.yaml file.

We fed the YOLOv9.yaml file and trained models over 100 epochs with learning rate, momentum, image size, and weight decay parameters. Our loss function is the binary cross entropy.

While model building we faced some difficulties specifically the overfitting issue, for addressing that we performed hyperparameter tuning with checking multiple values on parameters. Opted regularization techniques like early stopping, dropout, and data augmentation that's how we overcame that issue. This approach we followed to build a model and we got our best.pt file. Saving the trained model and reloading the trained model we saved our best.pt and last.pt file. In the local directory, reloaded the model to predict the results. Our prediction accuracy is 98% using the YOLOv9 model.

**VI. MODEL EVALUATION**

Evaluating how well our egg crack detection system performs is really important. To do this properly, we use different measures called evaluation metrics. These help us understand how good our system is at finding cracks in eggs and ensuring its proper operation. We use several metrics, like precision, recall, F1 score, mean average precision (mAP), false positives, and false negatives.[8]

Precision tells us how often our system correctly identifies cracked eggs without mistakenly saying an egg is cracked when it's not. Recall, or sensitivity, shows how good our system is at finding all the cracked eggs, without missing any. The F1 score is like a balance between precision and recall, giving us an overall idea of how accurate our system is.[9]



For egg crack detection, mean average precision (mAP) gives us a general idea of how well our system works across different types of eggs. False positives occur when our system says an egg is cracked when it's not, while false negatives occur when our system misses a cracked egg. We want to minimize both of these errors to ensure our system works accurately and doesn't cause unnecessary problems.[8]

By monitoring the number of false positives and false negatives in our system, we can see whether it's improving. If we see fewer of these errors over time, it means our system is improving and becoming more effective at finding egg cracks, which is important for reducing costs and ensuring the safety of our eggs.

### VII. MODEL ACCURACY TABLE

Table 1: Comparison of Model Performance

MODEL NAME	mAP(%)
YOLOv5s	94%
YOLOv8n	97%
YOLOv9c	98%
Faster R-CNN	92%
ResNet	90%

The best accuracy we got from the YOLO v9 model. [Table.1] talks about the model accuracy.

- **Metrics:** Past YOLO versions (like YOLOv8) were evaluated using metrics like mean Average Precision (mAP), which measures the model's ability to correctly identify and localize objects across different classes in the dataset. Other metrics might include precision, recall, and F1 score. YOLOv9 showcases significant advancements in object detection, offering a compelling balance between accuracy, speed, and resource efficiency.[8] Its comprehensive evaluation through various metrics and comparison with established models highlights its strengths and potential for deployment in real-world applications.[8,12] Further research can delve deeper into specific aspects of YOLOv9, such as hyperparameter optimization techniques and their performance on diverse datasets and tasks.
- **Hyperparameters:** Common hyperparameters for YOLO models include:
  - *Input Image Resolution*
  - Number of anchor boxes (used for object detection)
  - Learning rate (controls how much the model updates its weights during training)
  - Batch size (number of images processed at once during training)

- Number of training epochs (full cycles of passing through the training data)

### VIII. MODEL DEPLOYMENT

Deploying a model is like getting it ready for real-world use, so people can actually use it. We're going to talk about how we deploy our egg crack detection model using Streamlit, a really cool tool for building interactive websites.

First, we need to make sure we have the right files saved. These files hold all the important information about our model, like how it was trained and what it learned. We'll use these files to set up our model when it's running on the web.

Now, let's talk about Streamlit. It's great because it's easy to use and lets us build web apps with Python. We chose it because it's simple and lets us show our model's predictions in real-time. With Streamlit, we can quickly build and test our ideas, which is awesome for development. Plus, deploying with Streamlit is super easy - just one command, and it's done. We can host our app on platforms like Streamlit Share, Heroku, or AWS so that everyone can access it easily.

### IX. MONITORING AND MAINTENANCE

#### A. Monitoring

- *Ongoing Model Evaluation Techniques*
  - Keep a close eye on the YOLOv8 model's effectiveness in pinpointing eggshell fractures by employing the following measures:
    - **Performance Indicators:** Monitor crucial indicators such as accuracy, precision, sensitivity, and the harmonic mean of precision and sensitivity (F1 score). Employ confusion matrices as a diagnostic tool to pinpoint potential areas for enhancement.
    - **Inaccurate Predictions:** Vigilantly track instances of both false alarms and missed detections to pinpoint and rectify any inaccuracies in the model's forecasts.
    - **Instantaneous Performance Insights:** Implement systems for instant anomaly detection within the model's operation. This can be achieved through ongoing logging and the establishment of alert mechanisms.[12]
- *Data Integrity Supervision*
  - Ensure the dataset's integrity is preserved to maintain the model's effectiveness over time by:
    - **Dataset Surveillance:** Regularly scrutinize the dataset for any shifts in data patterns that could impact the model's accuracy. Refresh the dataset accordingly.
    - **Annotation Accuracy:** Guarantee the precision and consistency of image annotations to prevent the introduction of prejudice or errors into the model.
- *Streamlit Interface Supervision*
  - Ensure the Streamlit interface provides an uninterrupted and efficient user experience through

- **Operational Stability:** Continuously monitor the interface's operational stability to confirm its availability to users.
- **Troubleshooting Logs:** Set up systems for logging errors to quickly identify and solve any user-facing issues during application interaction.

#### B. Maintenance

##### ➤ *Frequent Model Refreshes*

To ensure the model remains reflective of current trends and enhances its accuracy, initiate a routine for consistent model updates:

- **Data Refreshment:** Routinely incorporate fresh images into the dataset to capture environmental shifts.
- **Update Cadence:** Set the retraining intervals by evaluating the pace of data collection and the accuracy objectives.

##### ➤ *Streamlit Application Enhancements*

Elevate the Streamlit application's user interface and functionality by:

- **Feature Evolution:** Roll out enhancements and new functionalities guided by user insights and changing needs.
- **Compatibility Assurance:** Monitor updates to Streamlit and related dependencies to guarantee smooth operation with new versions.

##### ➤ *Documentation Refinement*

Ensure the documentation is current to support understanding and efficient problem-solving:

- **Model Adjustments:** Chronicle any modifications to the model's structure, tuning parameters, or training methodologies.
- **App Interface Updates:** Revise the Streamlit guide to mirror updates in the application's design or capabilities.

#### C. Hyperparameters:

- **Convolutional Layer 1** has 6 filters with a kernel size of 5x5 and uses the Tanh activation function, processing input images of 28x28 pixels.
- **Average Pooling Layer 1** uses a pool size of 2x2 with strides of 2.
- **Convolutional Layer 2** increases the number of filters to 16, keeping the same kernel size and activation function.
- **Average Pooling Layer 2** repeats the configuration of the first pooling layer.

#### D. Mobile Net

MobileNet architectures, including versions 1, 2, and 3. MobileNetV1 introduces a width multiplier as a hyperparameter, allowing for the adjustment of the number of channels in each layer, directly influencing the computational cost and model size. This multiplier, also known as "alpha," can decrease (if <1.0) or increase (if >1.0) the number of

filters proportionally, affecting the computational cost quadratically.

MobileNetV2 and V3 further refine this approach with architectural improvements like inverted residuals, linear bottlenecks, and an emphasis on efficient layer structures.

#### E. YOLOv9

- **Metrics:** Past YOLO versions (like YOLOv8) were evaluated using metrics like mean Average Precision (mAP), which measures the model's ability to correctly identify and localize objects across different classes in the dataset. Other metrics might include precision, recall, and F1 score. YOLOv9 showcases significant advancements in object detection, offering a compelling balance between accuracy, speed, and resource efficiency.[8,12] Its comprehensive evaluation through various metrics and comparison with established models highlights its strengths and potential for deployment in real-world applications. Further research can delve deeper into specific aspects of YOLOv9, such as hyperparameter optimization techniques and their performance on diverse datasets and tasks.[12]

#### F. Hyper Parameters:

- Common hyperparameters for YOLO models include:
- Input image resolution
- Number of anchor boxes (used for object detection)
- Learning rate (controls how much the model updates its weights during training)
- Batch size (number of images processed at once during training)
- Number of training epochs (full cycles of passing through the training data)

#### G. R-CNN

- **mAP:** Main metric (precision & recall).
- **AP:** Per-class average precision.
- **Localization Error (IoU):** Accuracy of bounding boxes.
- **Speed:** Training and inference time.
- **Visualisation:** Qualitative assessment of detections.
- **Hyperparameters:**
- **RPN Anchors:** Size and aspect ratio of proposal boxes.
- **Learning Rate:** Controls weight updates during training.
- **Batch Size:** Number of images processed together.
- **Epochs:** Number of times to iterate through the dataset.
- **Optimizer:** Algorithm for updating model weights.
- **Weight Decay:** Regularization technique to prevent overfitting.

## X. RESULTS AND DISCUSSION

Though we have done wide range experiments to streamline the intact eggs, The latest model, YOLOV9 has shown extraordinary results in detecting the intact eggs. We began the study with the idea of minimizing the damaged eggs from the production line, but the study has strongly

shown that it is possible to prevent the damaged eggs from reaching the consumer if the YOLOv9 is incorporated into the production.[12] But, a single instance would not be sufficient to achieve the goal, but multiple cameras have to be set up above the conveyor belt to detect the damaged egg. Only then the damaged egg can be detected from another instance even if it escapes from one instance. Implementing this into the production will minimize the expenditure seamlessly and will reduce the manpower by 10%. Though we have implemented AI powered systems to minimize human error, in few matters, contribution from human efforts is inevitable.

## XI. CONCLUSION

After successful implementation of the AI powered YOLOv9 model, the naked truth is that the poor rearing practice, improper quality control and other human errors can be completely removed by these models with low cost and seamless performance. The study started with the assumption that the model might give a considerable amount of accuracy, but what happened in reality was a good surprise for us, because we ended up with 98% accuracy. From this, it is evident that AI can make anything possible with low cost and with commendable accuracy of detection. The data required to train these models is also abundant in today's world. This can not only be implemented for quality control but also to detect the abnormality of chickens, to check the womb quality in eggs, and many more tasks can quickly be achieved with almost little cost.

## REFERENCES

- [1]. Caguioa, J., Guinto, R., Mesias, L., & DE GOMA, J. (2022, March). Duck Egg Quality Classification Based on its Shell Visual Property Through Transfer Learning Using ResNet-50. In *12th Annual International Conference on Industrial Engineering and Operations Management*, <https://doi.org/10.46254/AN12.20220240>.
- [2]. Yang, Xiao & Bist, Ramesh & Subedi, Sachin & Chai, Lilong. (2023). A Computer Vision-Based Automatic System for Egg Grading and Defect Detection. *Animals*. 13. 2354. [10.3390/ani13142354](https://doi.org/10.3390/ani13142354).
- [3]. Tang, Wenquan & Hu, Jianchao & Wang, Qiaohua. (2022). High-Throughput Online Visual Detection Method of Cracked Preserved Eggs Based on Deep Learning. *Applied Sciences*. 12. 952. [10.3390/app12030952](https://doi.org/10.3390/app12030952).
- [4]. Ab. Nasir, Ahmad Fakhri & Sabarudin, Siti & P P Abdul Majeed, Anwar & Abdul ghani, Ahmad shahrizan. (2018). Automated egg grading system using computer vision: Investigation on weight measure versus shape parameters. *IOP Conference Series: Materials Science and Engineering*. 342. 012003. [10.1088/1757-899X/342/1/012003](https://doi.org/10.1088/1757-899X/342/1/012003).
- [5]. C. B. Sy, S. Chan, M. F. Sim, L. M. Te and M. Alipio, "Evaluation of CNN-Based Models for Classifying Poultry Eggs Images in Variable Angle, Background, and Lighting Conditions," 2023 IEEE World AI IoT Congress (AllIoT), Seattle, WA, USA, 2023, pp. 0324-0330, doi: 10.1109/AllIoT58121.2023.10174259.
- [6]. Datta, Ashis & Botta, Bhavya & Gattam, Sai. (2019). Damage detection on chicken eggshells using Faster R-CNN. *10.13031/aim.201901244*.
- [7]. Thipakorn, Jeerapa & Waranusast, Rattapoom & Riyamongkol, Panomkhawn. (2017). Egg weight prediction and egg size classification using image processing and machine learning. 477-480. [10.1109/ECTICon.2017.8096278](https://doi.org/10.1109/ECTICon.2017.8096278).
- [8]. Huang, Y.; Luo, Y.; Cao, Y.; Lin, X.; Wei, H.; Wu, M.; Yang, X.; Zhao, Z. Damage Detection of Unwashed Eggs through Video and Deep Learning. *Foods* 2023, 12, 2179. <https://doi.org/10.3390/foods12112179>
- [9]. Dehrouyeh, M & Omid, Mahmoud & Ahmadi, Hojat & Mohtasebi, Seyed & Jamzad, Mansour. (2010). Grading and Quality Inspection of Defected Eggs Using Machine Vision. *International Journal of Advanced Science & Technology*. 17. 23-30.
- [10]. Amin Nasiri, Mahmoud Omid, Amin Taheri-Garavand, An automatic sorting system for unwashed eggs using deep learning, *Journal of Food Engineering*, Volume 283, 2020, 110036, ISSN 0260-8774, <https://doi.org/10.1016/j.jfoodeng.2020.110036>.
- [11]. J. M. P. Sánchez, L. A. O. Moreno, J. L. R. Rodríguez and I. D. M. Corro, "Poultry Egg Classification System Using Deep Learning," 2023 5th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Istanbul, Turkiye, 2023, pp. 01-06, doi: 10.1109/HORA58378.2023.10156776
- [12]. Z. Zhao, Z. Zhang, Y. Luo, Y. Huang, X. Lin and S. Luo, "Research on Real-time Identification and Tracking Method of Defective Eggs Based on Deep Learning," 2022 International Conference on High Performance Big Data and Intelligent Systems (HDIS), Tianjin, China, 2022, pp. 173-179, doi: 10.1109/HDIS56859.2022.9991594.
- [13]. Bhavya Botta, Sai Swaroop Reddy Gattam, Ashis Kumar Datta,
- [14]. Eggshell crack detection using deep convolutional neural networks,
- [15]. *Journal of Food Engineering*, Volume 315, 2022, 110798, ISSN 0260-8774, <https://doi.org/10.1016>.
- [16]. Depuru BK, Putsala S, Mishra P. Automating poultry farm management with artificial intelligence: Real-time detection and tracking of broiler chickens for enhanced and efficient health monitoring. *Trop Anim Health Prod*. 2024 Feb 12;56(2):75. doi: 10.1007/s11250-024-03922-2. PMID: 38347390.