

2019

Identification and Classification of Poultry Eggs: A Case Study Utilizing Computer Vision and Machine Learning

Jeremy Lubich

Southern Methodist University, jlubich@smu.edu

Kyle Thomas

Southern Methodist University, khthomas@smu.edu

Daniel W. Engels

Southern Methodist University, dwe@smu.edu

Follow this and additional works at: <https://scholar.smu.edu/datasciencereview>



Part of the [Agricultural Economics Commons](#), and the [Artificial Intelligence and Robotics Commons](#)

Recommended Citation

Lubich, Jeremy; Thomas, Kyle; and Engels, Daniel W. (2019) "Identification and Classification of Poultry Eggs: A Case Study Utilizing Computer Vision and Machine Learning," *SMU Data Science Review*. Vol. 2: No. 1, Article 20.

Available at: <https://scholar.smu.edu/datasciencereview/vol2/iss1/20>

This Article is brought to you for free and open access by SMU Scholar. It has been accepted for inclusion in SMU Data Science Review by an authorized administrator of SMU Scholar. For more information, please visit <http://digitalrepository.smu.edu>.

Identification and Classification of Poultry Eggs: A Case Study Utilizing Computer Vision and Machine Learning

Jeremy Lubich, Kyle Thomas, and Daniel Engels

Master of Science in Data Science
Southern Methodist University
Dallas, TX 75275 USA
{jlubich, khthomas, dwe}@smu.edu

Abstract. We developed a method to identify, count, and classify chickens and eggs inside nesting boxes of a chicken coop. Utilizing an Internet of Things (IoT) AWS DeepLens Camera for data capture and inferences, we trained and deployed a custom Single Shot multibox Detector (SSD) model for object detection and classification. This allows us to monitor a complex environment with multiple chickens and eggs moving and appearing simultaneously within the video frames. The models can label video frames with classifications for eight breeds of chickens and/or four colors of eggs, with 98% accuracy on chickens or eggs alone and 82.5% accuracy while detecting both types of objects. With the ability to directly infer and store classifications on the camera, this setup can work in a low/no internet bandwidth setting in addition to an internet connected environment. Having these classifications benefits farmers by providing the necessary base data required for accurately measuring the individual egg production of every chicken in the flock. Additionally, this data supports comparative analysis between individual flocks and industry benchmarks.

1 Introduction

The number of backyard egg production farmers are increasing as the demand for organic, locally produced eggs becomes more popular. Many of these farmers lack training and guidance from the commercial egg production industry to understand the health and reasons for variability of egg production in their flocks. Age, breed, weather, parasites, feed, stress, housing and other factors play a role in egg production volume and quality. Backyard farmers who treat their flocks more as pets than livestock introduce many variables and conditions which are not tightly controlled and monitored like typical commercial farms. Novice farmers may not have the knowledge and laboratory access to monitor their flock, however they do have their own set of metrics which they can collect such as age, breed and the number of eggs they are observing. The automated collection of weather and number of eggs may provide the necessary data to help make a measurement of whether a given flock is performing above or below

average compared to peer data. Finally, we aim to understand how individual chickens are producing eggs using a culmination of the aforementioned tasks.

2 Related Work

The primary obstacle to overcome is the automated process of detecting and classifying eggs in a nest, and understanding which chickens have laid which eggs. Several machine learning technologies have been implemented to identify and count objects in the past. In fact, several research projects have focused on applying machine learning to agricultural processes.

Akintayo et al used deep learning techniques to identify the presence of nematode eggs in soil samples. More specifically, this group used a Convolutional Selective Autoencoder to learn the features of nematode eggs using input image data [2]. Then, this team used a selectability function to increase discrimination between eggs and background noise (such as soil particles) [2]. This group also used automated techniques to count the identified nematode eggs [2]. The end result was an automated process that could identify and count nematode eggs in soil sample images with an accuracy of expert level human labeling [2]. This example demonstrates that applying machine learning techniques to agricultural processes is not only possible, but quite valuable.

Significant amount of research has been conducted on topics that are critically important to achieving the aforementioned goals. One such critical task is object detection. Object detection requires that the selected training data be labeled and that there is some way to identify where in the image the object resides (this can be done with a bounding box). Interestingly, Microsoft COCO (Common Object in COntext) found that object detection is highly dependent on contextual information in the image [1]. The COCO team also found that the use of traditional bounding boxes “limits the accuracy for which detection algorithms may be evaluated”. Rather, COCO found that fully segmented images had a great impact on successful object detection. However, this approach required tens of thousands of hours to complete. This shows that while highly accurate object detection models can be built they require a significant amount of training data if using the image segmentation approach. We did not possess the resources to conduct full image segmentation, and as a result we use traditional bounding boxes. Additional accuracy may be possible if full image segmentation was performed, but this is beyond the scope of this paper and we leave it to future research.

There are several common threads that are found in many machine learning approaches to image recognition. Common to most of the approaches is the use of Neural Nets. More specifically, a significant number of image recognition problems utilize Deep Learning and many approaches utilize Convolutional Neural Networks (CNN) which is a sub-category of deep learning. Additionally, most approaches utilize a significant amount of training data.

3 Data

A variety of original data sources were produced which measure and describe the flock of 22 chickens on a hobby farm in Afton, MN.

3.1 Chicken Characteristics

Each chicken in the flock has background data including age, species, identifying visual characteristics and names. Ages range from 0-3 years. Species include categories of Cherry Egger, Buff Orpington, Easter Egger (White), Easter Egger (Gray), Easter Egger (Dark), Black French Copper Maran, Black Laced Silver Wyandotte, Barred Rock, Olive Egger. Note that the Easter Egger breed is specifically designed to be multi-colored and have a wide variety of looks which we classified down to 3 distinct groupings.

3.2 Egg Characteristics

There exists a strong relation between chicken species and colors of eggs they produce, though there is overlap between multiple species to a single color type. We've classified egg colors into blue, brown, pink and olive. The Easter Egger variety produces blue eggs, the Olive Eggers produce the olive eggs, the Wyandottes produce the pink eggs while all other varieties produce the brown eggs. There are no white egg layers in the dataset.

3.3 Dataset 1 Hourly Nest View

A dataset of 720 jpeg images was produced to simulate 1 photo being taken per hour for 30 days of a nest. This type of dataset is known as a periodic snapshot and was created under controlled photography conditions by creating an empty nest of straw and adding one egg at a time while taking photos on a fixed camera. We took 12-15 pictures in sequence until the nest was full and then reset the nest back to being empty. During the sequence of adding eggs we would minorly adjust any existing egg positions, move some of the straw bedding around a bit, and place a feather in frame to simulate the nest being altered by a hen. We repeated the process until a total of 14 sequences was collected. We then created a simple simulation of 720 hours of flock production and pulled the corresponding image into the final dataset to match the total number of eggs in the nest for each hour per the simulation. Thus, this sequence of data does not show individual timestamps for each egg being produced but can jump from 0-5 eggs because each image represents a snapshot of an entire hour of potential egg production between images.

3.4 Image Labeling

The following datasets contain labeled images. The process for generating this data started with uploading the images to the site labelbox.com, defining the

egg and chicken categories and subcategories (as described in the characteristics sections above), manually drawing a rectangular bounding box around each object in the images and assigning a classification to the object. After images have been marked up with these labels, the site labelbox.com produced a json file containing all the X & Y locations for each labeled object and image.

3.5 Camera Setup in Coop

We deployed an Amazon DeepLens camera mounted 36 inches above a set of 4 laying nests. Each nest measured 14 inches by 15 inches and had walls 24 inches tall around the nests. The top of the nesting boxes was wide open and the camera was positioned directly over the intersecting inner walls of the 4 boxes so that it could simultaneously view all 4 nests.

Video was captured through by the 4 MP 1080p camera and was sent to Amazon Kinesis Video streams which collected a continuous data feed from December 15, 2018 through January 26, 2019. The video was converted to streaming h264 format at 1920 x 1080 resolution with a total compressed volume of 833 GBs available for training.

Internet connectivity was only required during the acquisition of training data. When the camera is in production mode, inferences are stored locally to removable storage. If there is sufficient internet bandwidth, the inferences can be automatically uploaded to a central repository. If there is low/no internet bandwidth environments, the removable storage can be collected so those inferences can be uploaded manually by a farmer. Apart from the inference data, if a farmer would like to view the real-time video streams, the camera can output those given there is sufficient bandwidth for that feature (see Figure 1).

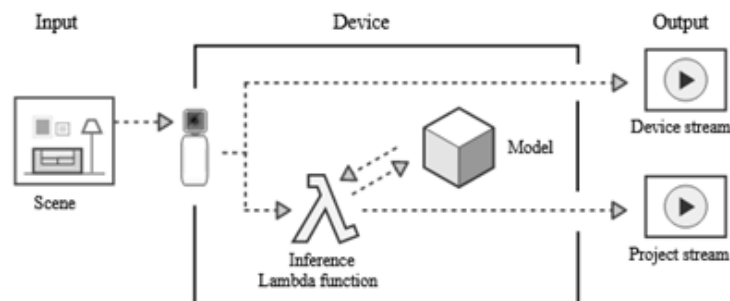


Fig. 1. This figure shows how an input image is processed by the DeepLens Camera (the “Device”) and how the output image can be stored on the device or sent back to the cloud via a project steam. Image created by AWS. [24].

3.6 Dataset 2 Eggs Only

A dataset of 100 images was prepared utilizing 50 images from dataset 1 and 50 images scraped from a web search for egg. All images were labeled with individual bounding boxes around one or more eggs in the picture. Included were many pictures of eggs that were not in nests, including egg cartons, Easter egg baskets, etc. This expanded the dataset to give it many backgrounds instead of just a typical nest.

3.7 Dataset 3 Labeled Single Nest in Coop View

A dataset of 109 images was collected and cropped from the installed coop camera. Images were cropped to a single nesting box. Resulting image size is 500 x 400 pixels and labeled with both chicken and egg objects. This dataset is designed to have only chickens or eggs within the image, however there were occasional images where two or more objects were visible within the frame.

3.8 Dataset 4 Labeled Multiple Nest in Coop View

Our largest and most complex dataset is a set of images containing the full overhead 4 nest view. These images are 1111 x 900 pixels and labeled with both chicken and egg objects. Most images contain 2 or more objects and both chicken and egg classes.

3.9 Dataset 5 Automatically (Object Detection Model) Detected Chicken and Egg Objects

The development of an object detection model running on the coop camera creates additional datasets for further classification of chicken and egg varieties and possibly chicken identification. Since these datasets are generated after a deployed object-detection model is in place we first crop all objects on their predicted bounding boxes, then perform labeling of the resulting images. We expect 1-2 thousand images to be automatically created for classification purposes.

4 Neural Networks, Object Detection, and Transfer Learning

4.1 Neural Networks and Convolutional Neural Networks

Our approach to the problem utilized convolutional neural networks. Neural networks are a systems of artificial neurons arranged in layers. The first layer is the input layer, where data is entered into the system. The last layer is the output layer which contains the output of the system. There can be an arbitrary number of layers between the input and output layers and these layers are referred to as hidden layers [4]. Once there are two or more hidden layers the neural network is referred to as a deep neural network [4].

Since such a large number of approaches utilize Deep Learning it is worth explaining this machine learning technique in more detail. Deep Learning is a “computational model[s] that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction” . Deep Learning methods are a type of representation-learning that utilize simple, but non-linear models, at various levels of abstraction [3]. By repeating this process multiple times very complex patterns and relationships can be learned [3]. In many instances, deep neural networks out perform shallow neural networks [4].

Deep Learning relies on a technique called back-propagation [3]. Back-propagation is the implementation of a cost function for neural networks [4]. It involves calculating how changes in the weights of each neuron impact the cost function and in turn give insight to the overall network [4]. Various cost functions can be used depending on the application.

Our approach to egg detection and classification relies on a specific deep learning technique called a Convolutional Neural Network (CNN). CNNs have had great success with image detection and classification. One of the major strengths of CNNs for image classification is that input data requires very little pre-processing as CNNs work on image information on a pixel by pixel basis. CNNs use several techniques in the learning process. These techniques are: convolutions, pooling, rectified linear units, and fully connected layers.

Convolutional Neural Networks have at least one convolutional layer in the network. A convolution is a mathematical operation that represents a way of combining two signals to make a third signal [8]. In a CNN this can be thought of as essentially looking for features in the input image. This is done by combining the signals from the image (signal one) and a filter matrix (signal two) to produce a third matrix (the result of the convolution). The convolution process essentially extracts features from the input image while also reducing the number of parameters in the network [9].

Convolutional Neural Networks may also utilize a technique called “pooling”. Pooling combines the output of multiple neurons from one layer into a single value in the following layer. There are various pooling methods, but the idea remains the same regardless of the pooling approach used. The goal is to reduce the image size while maintaining as much information about the image as possible [12].

Lastly, Convolutional Neural Networks have one or more fully connected layers. This means that each output neuron from one layer is connected to each neuron in the next layer. This essentially allows the network to “vote” based on the weights of the neurons [12].

4.2 Object Detection and Mean Average Precision

Object detection models are specific applications of deep convolutional neural networks. They deal with not just recognizing an object, but also specifically identifying where in the image an object is located. Most object detection algorithms utilize a similar process. They estimate bounding boxes (where an object

is located), sample pixels within each estimated bounding box, and apply a classifier to that bounding box [10]. This approach results in high accuracy, but is typically quite computationally expensive and slow [10]. Our modeling approach utilizes a Single Shot multibox Detector (SSD). SSD which combines image localization and classification in a single step which greatly reduces computational overhead [18]. An SSD accomplishes by making multiple “default” bounding boxes, generating scores for each box, and finally adjusting the boxes to improve box placement all in one step [10]. SSD is further discussed in the Methods and Materials section.

Evaluation of object detection algorithms presents an interesting problem. There is more to consider than accuracy or recall for the classifier. Since these algorithms are also trying to specifically locate where in the image a object resides attention must be paid to the precision of bounding box placement. The method that we utilize in this paper for evaluation of bounding box placements is the average precision (AP). AP can be viewed as the summary of the precision and recall curve for the classifications in the image [13]. Furthermore, AP is defined as the “mean precision at a set of eleven equal spaced recall levels” between 0 and 1 [13]. The mean average precision (mAP) is simply the average AP across all classes.

In addition to AP the concept of intersection over union (IOU) is critical the evaluation of object detection algorithms. IOT is the amount of intersected area, or overlap, between the truth bounding boxes and predicted bounding boxes divided by the the union of the area between the two bounding boxes [13]. The IOU for any detection must be greater than 50 percent before being considered correct.

4.3 Transfer Learning

Our approach heavily utilized transfer learning. Transfer learning departs from the assumption of many machine learning models. That assumption being that the training and test data have to be from the same data set and feature space [14]. However, we found that the process of data collection, processing, and labeling greatly limited that amount of training data that we could collect for this analysis. This problem was not unique to our situation. Indeed other many real world applications have experienced similar problems. As a result, many have studied the ability of models to conduct knowledge transfer between domains to reduce the amount of training data needed to achieve good results. This approach is known as transfer learning.

Ultimately, transfer learning allows for the “domains, tasks, and distributions used in training and testing to be different” [14]. This transfer can help overcome more than a simple lack of data, some studies have shown that transfer learning can also help with different data distributions between source and target domains that can be caused by differences in image lighting, vantage point, and background information [16]. While we were initially attracted to transfer learning due to our relative poverty of training data, these other benefits were quite attractive as there are lighting imbalances between images, and the chickens are

live and therefore appear in a wide variety of angles and vantage points. The background of our training data is relatively static, so the ability to deal with imbalances in background context is not directly applicable to our application.

The pre-trained model that we utilize is a pre-trained ResNet-50 model [5]. This model consists of 50 layers, was trained on 1,280,000 images, and could detect 1,000 different classes [17]. This model expects inputs in the shape of $(N \times 3 \times H \times W)$ [15]. In this equation N is the batch size, H is the height of the image, W is the width of the image, and three represents 3-channel RGB [15]. The equation is expecting H and W to be at least 224 [15]. These assumptions are treated as hyper-parameters in the AWS model. There are many other hyper parameters than can be used to fine tune the model.

5 Hens and Egg Production

Egg production of a chicken has many variables which affect the output of a laying hen. Egg production does not require a rooster to fertilize the hens to produce an egg. A hen starts laying when she is about 6 months old. Chickens need about 25 percent of their diet to be protein. Increased protein, greater than 25 percent, in the diet may accelerate the time when the first egg is laid down to about 4 months, though there is debate on the ethics of forcing egg production this early [22]. During a hens life she lays only the number of eggs she was born with, and once she is out of eggs she will continue to live out her days but does not produce additional eggs. Typically, a hen produces only 500-600 eggs total and these will be laid over the course of 2 years and taper off.

Egg production is directly tied to the amount of daylight a chicken experiences. Typically, it takes 26 hours of daylight for a chicken to go through the process of creating an egg [23]. Artificial lighting is commonly used to keep the egg production cycle operating continuously in egg factories. Natural lighting for free range chickens varies the speed of the production process as shorter days, such as winter in north American, slow down the production of those hens.

Other factors such as weather, stress, molting and illness can reduce egg productions. Changes in weather, like cloudy grey skies and rain, can reduce the sunlight which stimulates a flock to lay and can take a couple days for the flock to get back to normal laying volumes. Twice a year chickens molt, which means they lose their feathers so that they can grow new ones [22]. During molting their bodies need additional protein to regrow feathers. Stress can be caused by changes in their environment, predators or even inter-flock fighting. Many times the roosters (adult male chickens) actively manage the pecking order within the flock and hens who are lower in the hierarchy get attacked. This can cause stress and injuries which reduce the energy and protein available to the hen to use for egg production. Sexual aggressiveness and fertilization of eggs slows down the production rate of hens.

Hens share the responsibility of incubating eggs and when a sufficient clutch of eggs has been laid a hormone triggers them to go broody. Broody hens have a strong instinct to stay on a nest of eggs until they hatch, which takes about 21

days [22]. During that time their metabolism slows, they stop laying eggs and don't leave the nest except for about once a day to quickly eat, drink and relieve themselves. A broody hen generally loses much of their body weight and after they leave the broody cycle it takes a few weeks for their body to get back into the normal egg production rhythm.

Various varieties of chickens have been selectively bred to optimize various laying qualities of chickens. Some breeds are optimized for cold or hot weather, laying vs meat production, white eggs vs colored eggs, large vs small body size [21]. Finding the right type of breed for your egg production needs can be difficult with so many choices available. The American Poultry Association is the foremost organization which defines and recognizes over 113 distinct breeds. These breeds come from many parts of the world and are organized into 12 classes. Within classes many colors, shapes and sizes are available and additionally many hybrid types of chickens exist by selectively mixing these standard breeds.

Egg shells are created to protect and store the egg contents. The shells are mainly calcium carbonate and are naturally white colored for all breeds during the first 20 hours of the egg production process. Commercial producers mainly choose breeds which keep the eggs white colored. Naturally though some breeds create different pigments which turn the shells pink, brown, blue or green. The brown pigment is called protoporphyrin and the blue pigment is called biliverdin [19][20]. These two pigments are chemically similar with biliverdin being derived from protoporphyrin through the addition of iron to that compound. When a chicken produces biliverdin it coats the outside of the shell while the inside of the shell remains white. When a chicken produces protoporphyrin it penetrates the shell and the inside and outside have the blue color. Certain breeds can produce both pigments and produce a range of blended colors in the green spectrum, and the outside of the shells are green while the inside may be blue.

Accurate tracking of a flock's egg production can help a farmer understand if there are trends or anomalies in their egg production which should be investigated. Currently there are few methods available to accurately track egg production by hen since they are mixed together within a flock and monitoring them 24x7 is impossible.

6 Method and Materials

6.1 Approach

In order to achieve the goal of tracking individual chickens and eggs we decided to break down the problem in smaller sub-problems that would allow us to learn and push toward the final goal. Broadly speaking, the problem can be broken down into six sub-problems. These sub-problems are: egg identification, chicken identification, chicken and egg identification occurring simultaneously, breed identification of both chicken and eggs, identification of individual chicken and eggs, and tracking chickens and eggs across time. Each one of these sub-problems can be approached with an individual model or algorithm. By making

separate models we can also compare how different models preform at similar tasks. For example, one of our hypothesis was that the algorithms that were responsible for detecting chickens or eggs alone would preform better than the model that detects both simultaneously. What follows is a analysis of each of the aforementioned sub-problems. For each sub-problem two models were created. One model that utilized transfer learning, and one trained from scratch using the same base ResNet50 architecture but with the weights of the neurons randomly assigned.

The data set that we utilized consisted of 363 different images. In total there were 12 unique classes across the images. There were eight different classes that represented chicken breeds, and four different classes that represented egg-shell color. For three of the models these classes were collapsed into smaller subgroups. There was an egg sub-group which contained only images of eggs and consisted of one class. There was also a hen only sub-group which also consisted of a single class. Finally, there was a hen and egg sub-group which contained two classes rather than 12.

6.2 Egg Detection

The first model that we created was an egg detection model. To start, our team utilized an image objected detection algorithm provided by Amazon SageMaker. The included algorithm is a convolutional neural network. More specifically, it is an implementation of ResNet-50 [5]. This algorithm can be trained from scratch, or it can be trained using transfer learning [5]. If training from scratch, SageMaker creates a new ResNet instance with random weights that can then be trained from the provided training data [5]. Alternatively, the transfer learning approach uses weights and architecture from a pre-trained model as a starting point and a more specific model is trained from there. [7]. We utilized the transfer learning in this approach as we only had 100 total images for training and validation. Typically, in a deep learning approach this amount of training data would produce a nearly worthless model. However, utilizing transfer learning allows for a much more useful model with this amount of data.

Additionally, the algorithm from AWS utilizes a Single Shot multibox Detector (SSD) [7]. SSD is a method that specifically deals with creating the bounding boxes that are the output of many object detection algorithms [10]. This method works by “predicting category scores and box offsets for a fixed set of default bounding boxes using small convolution filters applied to feature maps” [10]. Additionally it should be noted that the SSD performs image localization and classification in the same step. This method is quite useful as it results in highly accurate predictions in a timely manner. The speed of this algorithm is important as it has allowed for object detection to move from a speed of seconds per frame to many frames per second [10]. This fact is critical for our application as we are attempting to conduct object detection on live streaming data.

The training data used for this algorithm consisted of 100 images of chicken eggs. Fifty of the images came from our test chickens. These images contained different number of eggs, between 1 and 16 eggs, in a nest all taken from the

same height. The other fifty images were pictures of chicken eggs gathered from the internet. These images contained eggs in different connotations. For example there were images of chicken eggs in peoples hands, or in an egg carton. Furthermore, the eggs contained images at different camera angles. The idea was that providing the algorithm labeled chicken eggs in different connotations would result in a more accurate model.

Once the training data had been created, and the base models were selected we set the hyperparameters that could be used in this object detection algorithm. There are in fact a dizzying amount of combinations that can be used when training these models as many of the training hyperparameters are continuous inputs. Traditionally, an exhaustive search of selected hyperparameters would have been conducted until an optimal solution was empirically found. However, we utilized AWS Hyperparameter Tuning feature. This approach treats the selection of hyperparameters as supervised machine learning regression problem [11]. This allows the user to input possible test ranges for hyperparameters and the tuner will at first make a few guesses to see how the model preforms [11]. From that point onward the tuning process uses regression to choose the next hyperparameters to test, and proceed in an iterative fashion to maximize the target valuation metric [11]. The validation metric that we decided to use was mean average precision (mAP).

The model was run 10 different times during the hyperparameter search process. After completing these iterations the best configuration was reported. The hyperparameters and mAP for each approach are shown below:

Table 1. Comparison of Egg Only Models

Hyperparameter	Pre-Trained	Raw Network
Epochs	50	50
Learning Rate	0.16532	0.19188
Learning Rate Schedule Factor	0.1	0.1
Learning Rate Schedule Step	10	10
Mini-batch Size	6	3
Momentum	0.9	0.9
Number of Classes	1	1
Number of Training Samples	74	74
Optimizer	adadelata	adadelata
Weight Decay	0.0005	0.0005
mAP	0.98	0.0547

The tuning jobs results in different Learning rates and mini-batch sizes. The optimizer found to be optimal for both models was adadelata. The major difference between the two models is the mAP. Using the pre-trained model allowed us to achieve a much higher mAP using the same training data.

6.3 Chicken Detection

The hen detection model followed the same strategy of the egg detection model. Just like the egg detection model, a ResNet-50 CNN was used as the base network. SSD was used as the object detection and classification algorithm. The training data used to create this model was a subset of the master training data. We search the JSON label data for images that only had hens in them (in other words, none of the training or test data contained an image of an egg). This resulted in a training set of data. Once the data set was sub-selected, and the base network was set up, we utilized a hyperparameter search to find the best model possible utilizing both a pre-trained model and a randomize network. After 50 iterations the best performing models were compared. The results are shown in the table below:

Table 2. Comparison of Chicken Only Models

Hyperparameter	Pre-Trained Raw Network	
Epoch	50	50
Learning Rate	0.2	0.13532
Learning Rate Schedule Factor	0.1	0.1
Learning Rate Schedule Step	10	10
Mini-batch Size	4	6
Momentum	0.9	0.9
Number of Classes	1	1
Number of Training Samples	142	142
Optimizer	adadelta	adadelta
Weight Decay	0.0005	0.0005
mAP	0.98	0.279

Once again the two different methods resulted in different learning rates and mini-batch size. The best optimizer remains to be adadelta. There is still a significant improvement between the pre-trained network and the raw network. Interestingly, there is also a moderate discrepancy between this single class model, and the eggs only single class model for the randomized network (eggs only mAP of 0.05 compared to hens only mAP of 0.279). This difference can likely be explained by the fact that there are more images in the chicken only data set. Alternatively, it could be due to the fact that the eggs are a much smaller object to detect compared to the chickens.

6.4 Chicken and Eggs

So far, we have shown that we can build a very accurate model that detects both chickens and eggs separately. However, we now want to build on top of this

model and see if there is a decrease in accuracy when detecting both eggs and chickens simultaneously. The training data for this model consisted of all the images of chickens and eggs from four nesting boxes. In total, there were 290 images used for training and 72 images used as validation. The images contained labeled training data of eggs alone, chickens alone, and chicken and eggs in the same image.

The training process for this model is the same as the previous two models. A base ResNet-50 model utilizing transfer learning and a hyperparameter search is used to search for a model with the lowest mAP score. Likewise, a raw network is trained using the same data for comparison. The final models had the following hyperparameters:

Table 3. Comparison of Two Class Chicken and Egg Models

Hyperparameter	Pre-Trained Raw Network	
Epochs	50	50
Learning Rate	0.07367	0.17767
Learning Rate Schedule Factor	0.1	0.1
Learning Rate Schedule Step	10	10
Mini-batch Size	5	4
Momentum	0.9	0.9
Number of Classes	2	2
Number of Training Samples	290	290
Optimizer	adadelta	adadelta
Weight Decay	0.0005	0.0005
mAP	0.8974	0.2615

This model follows the same pattern of the previous two sub-problems. Learning rates and mini-batch size differ between the two models, yet the optimizer remains the same. There is not as much difference between the raw network model, and the raw network model of the hens only classifier. In fact the performance decreased between the raw network models despite there being more training data in the two class problem. This could be due to the random nature of the hyper-parameter tuning, or it could also be the fact that the raw network simply does not have enough training data to reliably find eggs in the images. This is an interesting hypothesis because the chickens are much more complex in their shape, color, and relative size when compared to the eggs making them easier for the network to identify. Therefore, the inclusion of the eggs in this model could explain the decrease in the mAP score as there is simply not enough data available for the network to reliably detect eggs.

While the mAP of the pre-trained network is still impressive considering the poultry training data set, it is significantly less accurate than both the egg only

and chicken only models. This is likely due to the fact that this model had to manage multiple classes.

6.5 Breed Detection

We have seen that the object detection algorithm can reliably distinguish between chickens and eggs. The last thing that we wanted to test for the object detection model was its ability to detect individual breeds of chickens and eggs. To accomplish this task, we had to alter the JSON meta-data. We kept the same image locations, but we had to update the class information to reflect individual chicken breeds and eggs. Chickens are much easier to label. In total there were eight different breeds of chicken utilizing the chicken coop. However, labeling eggs proved to be more difficult. We could not readily distinguish between the eggs by breed, so these were grouped by color. In total, there were four different egg colors identified in the images.

Once the training data had been updated to reflect these new class labels, we essentially re-ran the chicken and eggs model in another hyperparameter search. We used the same base network and utilized transfer learning. Our hypothesis was that the mAP of this model would decrease significantly as there were now 12 different classes for the algorithm to learn. The best model learned after the hyperparameter search had a mAP of 0.8255 which represents only a small decrease in accuracy over the chicken and egg detection model. The learned model had the following hyperparameters:

Table 4. Comparison of Twelve Class Chicken and Egg Models

Hyperparameter	Pre-Trained Raw Network	
Epochs	50	50
Learning Rate	0.023547	0.01211
Learning Rate Schedule Factor	0.1	0.1
Learning Rate Schedule Step	10	10
Mini-batch Size	6	4
Momentum	0.9	0.9
Number of Classes	12	12
Number of Training Samples	290	290
Optimizer	adadelata	adadelata
Weight Decay	0.0005	0.0005
mAP	0.8255	0.2646

Once again the both of the models find different optimal learning rates and mini-batch sizes, and settle on the same optimizer. There is still a large difference between the pre-trained model and the raw model. Interestingly, this model with

12 classes in it did not perform much worse than the two class model. There was a 7.19 percent decrease between the 2 class chicken and egg model and the 12 class pre-trained models. There was a slight increase in accuracy between two raw networks (0.3 percent increase). We found these results to be very interesting. Based on the large drop in mAP when moving from one class to two classes we expected significantly lower mAPs for the 12 class models.

An example output of the 12 class model is shown below. Note that it classifies hen breed and egg color, and that the bounding boxes have accurately localized the objects.

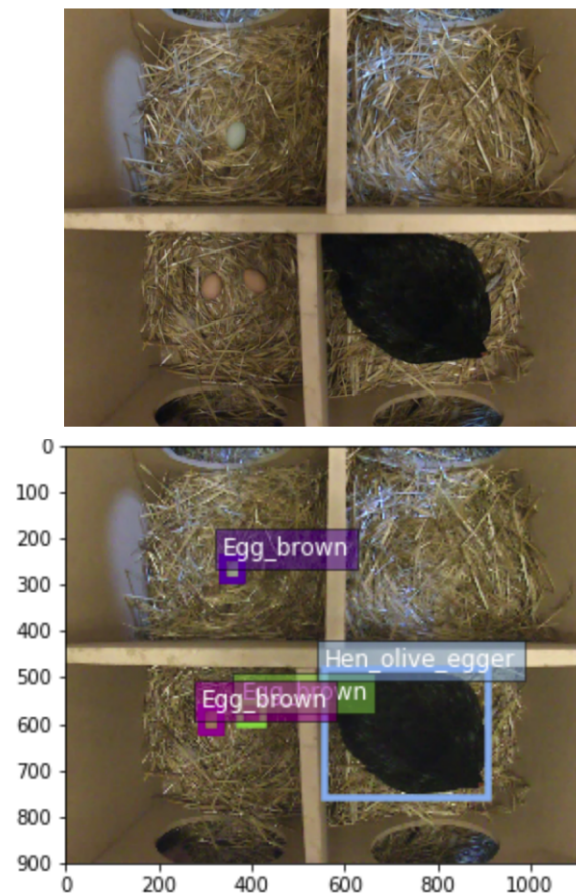


Fig. 2. This image shows the output of the modeling process. A before and after image is shown for ease of comparison.

6.6 Egg Quantification and Tracking

In order to effectively understand the egg production of hens the number of eggs, by egg type, must be calculated and tracked between frames. Fundamentally this can be broken into two sub-problems. Counting eggs, and tracking them between the frames. Our approach to solving these problems involves a more traditional approach using custom made algorithms.

Counting the eggs is relatively straight forward. Each processed image from the 12 class object detection model comes with a confidence score of each detection, a predicted class label (integer representation), and a bounding box location. We can easily utilize this data to count the eggs by class membership. However, the output of object detection model is quite noisy, in that there are many predictions that are of low confidence that contain miss-classifications. We created a simple algorithm that filters out predictions based on a confidence level threshold, and returns a sub-set of predictions. We can then use this filtered sub-set to count the number of classes that remain. The challenge remains in finding a cutoff threshold that reliably filters out misclassifications without also discarding valid predictions. Interestingly, even through the mAP values for the 12 class object detection model is relatively high, there are individual predictions of eggs in some of the test data that have extremely low confidence values associated with them. This means that in order to properly detect all of the eggs the threshold calculation must be finely controlled. An example of how small differences in the cutoff threshold can change results is shown in Figure 3. Currently, we arrive at a cutoff value through empirical testing, although a more rigorous approach would be warranted if this were to be deployed in a production environment. In general, we find that low confidence (high teens to low twenties) tend to find all of the eggs without missing eggs in the nest. This process was repeated for chickens.

Low threshold values increases the odds of false positives. We are of the opinion that having false negatives (not detecting an egg) would be preferable as the mistake would be easier to catch in subsequent input data. For example, if an egg was not detected and a new hen laid an egg there is a chance that the missed egg will be moved. This movement could produce a more favorable image and result in detection. Then, the counting is only off slightly by a hen, rather than having the potential for misidentified eggs to go missing in the next image. When deployed onto a video stream there may be many inferences taken on the same/similar images due to the streaming nature of the camera. This allows for multiple comparisons of the same or similar input images to correct for any missed eggs.

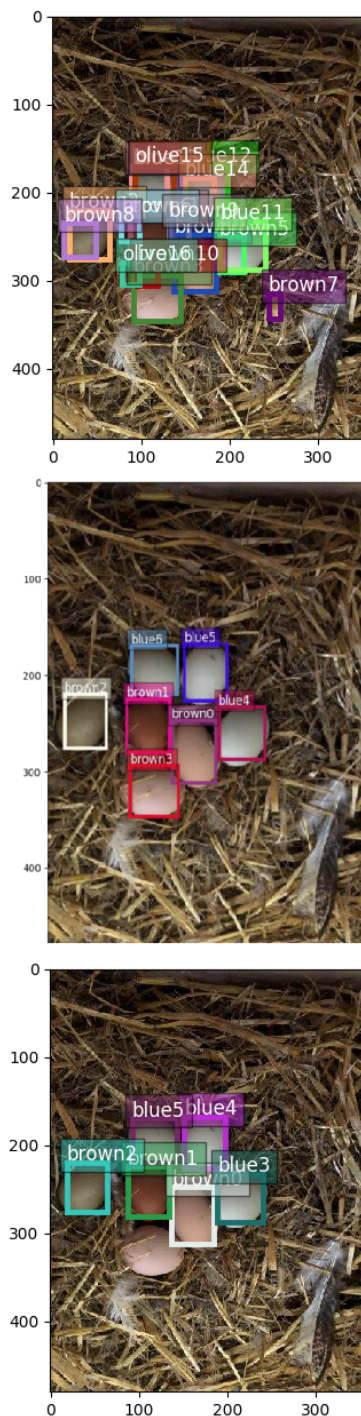


Fig. 3. This image shows how the cutoff threshold of prediction confidence impacts bounding boxes and the detection of eggs. On the top, the cutoff threshold was too low (0.05). The image in the middle correctly identifies all of the eggs and their classes (threshold of 0.17). The bottom image shows that an egg has been missed if the threshold level changes only slightly (threshold of 0.18). This demonstrates the need to develop a metric to determine optimal threshold so that predictions can be appropriately used to count and track individual eggs.

Once we could reliably detect and identify chicken eggs and breeds we wanted to track eggs between images. Ideally, this would allow us to determine when a new egg was laid, and which breed of hen laid it. Multiple attempts were tried, but most ended in failure. At first, we wanted to test to see if we could track eggs in the image based on the average color of the egg shell. This was done by tacking identified egg labels (using class information), and sampling pixels within their bounding boxes. Then, we could take the sampled pixels and conduct simple t-tests to determine if the average RGB color between two different images was the same. In theory, this could allow us to track individual eggs between images as there should be minor differences in egg shell color that would impact the overall color values. Our process involved sampling 1000 pixels with replacement from the two different images. We tested first, between eggs that we knew were the same. We ran into two main problems with this simple approach. First, there were differences in lighting between images which impacted the sampled colors. In addition, the number of samples resulted in a very powerful test and a very minor difference in egg shell colors due to the sampling process was treated as a false negative. We believe that this problem could be solved if differences in lighting were better controlled.

Another method that we examined was using the class labels and information contained in the bounding box to track egg location using euclidean distance. The idea here is that we could use class membership to partition the eggs into groups. Then within each group we can track the location between images by finding the closet egg of the same color based on their last known location. The current version of the algorithm iterates through all of the detected eggs at or above the threshold value and maps it to the closet match in the next image. However, due to the data structure the same egg can be used more than once. This algorithm needs to be improved so that eggs can only be used in 1 mapping and account for ties in distance.

6.7 Model Deployment

Once the object detection and classification models have been trained they are deployed to the field using an Amazon Deep Lens Camera. Once deployed, our team utilizes Kinesis Video Streams which is another technology provided by Amazon. Amazon Kinesis Video Streams allows the team to use the previously trained models to detect the presence of chickens. This serves as a triggering event to record a predetermined amount of video (typically between 30 and 60 seconds). This video capture is triggered each time a chicken is detected. Therefore, if a chicken enters the nest to lay an egg then there is video of the chicken the entire time until a chicken is no longer detected (empty nest or eggs only in the image). We can then use the last image in the video to run more analysis on egg production with the knowledge that a chicken has been in the nest recently. We can then count the eggs in the nest and compare that information to the state before the chicken entered the nest.

7 Results and Discussion

Utilization transfer learning has yielded great results for this project. With only 363 images we were able to create object detection models with mAP values between 0.8255 and 0.99. Without utilizing transfer learning, while using the same training data, the object detection models had mAP values between 0.0547 and 0.279. Clearly, using transfer learning allowed our analysis to proceed significantly faster than if we had built a model from scratch. We likely would have had to have hundreds, or even thousands of more training images to achieve similar results.

While all of the models that were based on transfer learning had good mAP values, we found that the prediction output had a wide variety of confidence values for individual egg detections. Some of the predictions that correctly identified an egg class contained very low confidence values. This presented us with an interesting problem of picking a cutoff confidence threshold to partition the predictions into practical working data set that we could feed to tracking algorithms. We have no definitive answer on what cutoff threshold should be used and we recommend that this problem be further studied so that a valid metric can be created and used to reliably select a cutoff threshold.

Once the final models were created, we investigated methods to track individual eggs between images. We utilize the class membership and bounding box information from the models to track egg location. We investigated two primary methods. First, we broke apart the eggs in each picture based on its predicted class membership. We then iterated through each egg by group membership and found the closest egg based on label and euclidean distance based on the X and Y coordinates of the bounding box. Another method that we examined was tracking eggs based on egg shell color. For each egg we sampled the pixel data within the bounding box to get RGB values for each egg. Then we conducted simple t-tests between each egg to determine if the average color of the egg was the same between the two images. Unfortunately, this method did not differentiate between eggs. We believe that this is due to color differences between the images based on lighting. We believe that this problem could be controlled for by fixing lighting levels in the coop to minimize color distortions. Thus far we do not have a reliable method to track individual eggs between images. The ability to track individual objects, in our example these objects happen to be eggs, between two images across time represents an interesting research problem, but one that proved to be beyond the scope of this work.

Once tracking of individual eggs is possible, and can be mapped back to a single chicken, analysis could be conducted on that chickens productivity automatically. These analytics could then be compared to statistical base lines to evaluate chickens as a whole, and the productivity of the flock as a whole. If there is insufficient data the productivity of differing breeds could be tracked automatically once egg tracking is complete. While this is not as a fine analysis as individual chickens, it still can provide farmers with detailed insights into their flocks.

8 Ethics

Acquiring, processing and storing the video and labels of poultry flock production carries ethical considerations which must be addressed from several different perspectives. If released as a product, the farmers who provide video access to their flocks provide a certain amount of data currency which can be tapped into and must be protected and handled in an ethical manner. As researchers we hold a responsibility to ensure there is transparency and security in how we handle the data as leaks or mishandling of this data could have damaging consequences to farmers.

There is considerable debate about the ethics of animal food production in general, so farmers need to protect the images of their farm to maintain an image of wholesomeness. One of the worst looking places in a farm may be the chicken coop; conditions where a video camera is monitoring a flock may not be representative of the overall conditions that the chickens experience. This biased sample of living conditions may be used against a farmer; thus, our responsibility is to protect the images acquired from being made available to the public. Consent to collect and keep acquired video data private is paramount.

The primary goal for most poultry farmers is to maximize produce eggs. Data on free range poultry flock production is typically only possible to collect on an aggregate basis, that is, we may know that 25-70 percent of a flock is producing, but we don't typically know which birds are producing and which are not. As farmers have better technology to measure the production rates of their flock as well as individual birds they may have better data to perhaps change conditions but may also be led to cull birds more selectively/aggressively if data shows certain birds are not producing as much as others.

What happens if our data suggest a chicken is an under-performer? If a chicken is truly under-performing, a farmer may decide to cull that chicken. What if however, it was simply a temporary period of under-performance that would naturally resolve, such as molting or illness? If we led the farmer to cull that bird early, he may lose out on the benefit of having that bird produce at a later in the future. We as researchers have an ethical responsibility of ensuring our data is accurate and is presented in a manner which does not incorrectly label certain birds as under-performing unless sufficient data warrants that label.

We must ask ourselves what our obligation is if we observe a poorly cared for flock. What if we see signs of cruelty, abuse, neglect and/or disease? Are we obliged to report on these conditions even if it may breach the confidentiality agreements we have with the farmer? The American Veterinary Medical Association's (AVMA) website lists a state by state list of requirements for reporting animal abuse. Some states have a mandatory policy; however these laws typically apply only to veterinarians, and not researchers such as ourselves. Nevertheless, if this system is further developed we may have to determine if and when we report animal abuse even if it is not a legal requirement.

Chicken breeding and poultry production businesses must guard their trade secrets and production rates to stay competitive. We as researchers bear a significant responsibility to keep any acquired data protected and safe. Farmers

owner the data about their flocks and we must get express consent on any use and anonymization of their data.

As we collect data from multiple flocks we have the ability to do cross flock analysis and spot differences between how flock conditions materially correspond to production rates. For example, if we spot certain types of feed correlate to better production outcomes we may be able to bring these product recommendations forward to the poultry community. Consider the scenario where we get an industry sponsor which produces a certain brand of feed. Would we be ethically obligated to disclose that sponsor? Would we be able to ethically recommend their feed, especially if we saw it was not the best option? These are among a myriad of ethical considerations that arise when we start looking at inter-flock analysis and sponsors.

9 Future Work

Two major sections of future work involves long term time-series analysis and inter-flock analysis.

Our research was conducted only on a 6-week sample of data during the winter of 2018-19 on a flock of 22 chickens which were of mixed ages. In the future, we'd like to start collecting data on individual egg production starting from their first eggs until their last. Over time we can build a lifetime profile of a chicken, so we can better understand how seasonality and other factors tie into the production rates of individual birds. During a bird's lifetime they may go through molting, illness or other factors which cause temporary periods of decreased production. It is unknown if there are factors or even periods of increased production. A longer period of data collection may take up to 3-5 years as many chickens keep producing up until that age (they may live for more than 10 years, but egg production ceases before then).

Another future work would look at the correlation between weather and production. It is known that sunlight is directly tied to production rates, however less is known about how temperature, humidity and stress factors into production rates of free-range flocks. Can farmers mitigate against some of these conditions and how effective are would these mitigation strategies?

One type of future work involves inter-flock analysis. For example, by comparing peer flocks we can determine if a flock is under or out performing other flocks. Can we show that different feeds can impact the production of similar flocks, holding all other factors constant? The same type of analysis could be performed for species, coop-type, amount of free-range time, amount of medical treatment, type of medications and/or immunizations, and a myriad of other conditions.

There are some technical challenges that we would like to work through to aid in this analysis. We have observed that there are color imbalances in the images caused by differences in the time of day, climatic conditions, and the number of chickens in the coop. Additional accuracy may be possible if the lighting conditions in the coop were controlled, or through additional pre-processing of

the images. The mAP values of the models would very likely be improved with additional training images. Building upon the foundation of an accurate system for the measurement of poultry production by bird really unlocks potential for future work that is simply impossible to do through conventional method and we are excited to see where the future of this work goes.

10 Conclusions

We present a practical solution solving measuring egg production of chickens in a free-range flock environment utilizing an IOT camera system. Our custom SSD object detection and classification model classified when chickens and eggs were detected by the video camera. Our models can label video frames with classifications for 8 breeds of chickens and 4 colors of eggs, with 98% accuracy on chickens or eggs alone and 82.5% accuracy while detecting both types of objects.

Tuned accuracy is needed for proper thresholding of object detection. Corner cases of object detection near the threshold are not appropriately captured using the models due to the sensitivity of the threshold. Tuning inaccuracy can result in either too few or too many detections of both chickens and eggs.

Future research and value-added analysis of the inference data from our models can work towards the future goals of understanding the egg production of individual chickens while utilizing this method of automated data collection and enrichment. This data may allow farmers and researchers to better understand flock health and variability when multiple flocks are monitored with this system.

References

1. Tsung-Yi Lin et al. *Microsoft COCO: Common Objects in Context* <https://arxiv.org/pdf/1405.0312.pdf>
2. Akintayo, Adedotun et al. *A deep learning framework to discern and count microscopic nematode eggs*
3. Lecun, Y., Bengio, Y., and Hinton, G. (n.d.). Deep learning. *Nature*, 521(7553), 436444. <https://doi.org/10.1038/nature14539>
4. Michael A. Nielsen, "Neural Networks and Deep Learning". Determination Press. 2015. <http://neuralnetworksanddeeplearning.com/index.html>
5. Amazon SageMaker. Image Classification Algorithm. <https://docs.aws.amazon.com/sagemaker/latest/dg/image-classification.html>
6. Amazon SageMaker. How Image Classification Works. <https://docs.aws.amazon.com/sagemaker/latest/dg/IC-HowItWorks.html>
7. Amazon SageMaker. How Object Detection Works. <https://docs.aws.amazon.com/sagemaker/latest/dg/algo-object-detection-tech-notes.html>
8. Smith, Stephen W. 1997. *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Publishing. <http://www.dspguide.com/>. Retrieved 11/3/2018.

9. Aghdam, Hamed H, and Elnaz J. Heravi. Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification. 2017. Internet resource. Retrived 11/3/2018.
<https://www.worldcat.org/title/guide-to-convolutional-neural-networks-a-practical-application-to-traffic-sign-detection-and-classification/oclc/987790957>
10. Liu W. et al. (2016) SSD: Single Shot MultiBox Detector. In: Leibe B., Matas J., Sebe N., Welling M. (eds) Computer Vision ECCV 2016. ECCV 2016. Lecture Notes in Computer Science, vol 9905. Springer, Cham
11. Amazon SageMaker. How Hyperparameter Tuning Works. <https://docs.aws.amazon.com/sagemaker/latest/dg/automatic-model-tuning-how-it-works.html>
12. Roher, Brandon. How do Convolutional Neural Networks work?. https://brohrer.github.io/how_convolutional_neural_networks_work.html
13. Everingham, M., Van Gool, L., Williams, C.K.I. et al. Int J Comput Vis (2010) 88: 303. <https://doi.org/10.1007/s11263-009-0275-4>
14. S. J. Pan and Q. Yang, "A Survey on Transfer Learning," in IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359, Oct. 2010.
15. MXNet: Model Zoo. Gluon Model Zoo. Overview. https://mxnet.incubator.apache.org/versions/master/api/python/gluon/model_zoo.html
16. Oquab, Maxime and Bottou, Leon and Laptev, Ivan and Sivic, Josef. Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). June, 2014.
17. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In IEEE International Conference on Computer Vision and Pattern Recognition.
18. Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, Kevin Murphy; The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 7310-7311
19. R. Zhao, G.-Y. Xu, Z.-Z. Liu, J.-Y. Li, N. Yang, A study on eggshell pigmentation: biliverdin in blue-shelled chickens, Poultry Science, Volume 85, Issue 3, March 2006, Pages 546549, <https://doi.org/10.1093/ps/85.3.546>
20. S. Samiullah, J. R. Roberts, K. Chousalkar, Eggshell color in brown-egg laying hens a review, Poultry Science, Volume 94, Issue 10, October 2015, Pages 25662575, <https://doi.org/10.3382/ps/pev202>
21. Imsland, Freyja & Feng, Chungang & Boije, Henrik & Bed'Hom, Bertrand & Fillon, Valrie & Dorshorst, Ben & Rubin, Carl-Johan & Liu, Ranran & Gao, Yu & Gu, Xiaorong & Wang, Yanqiang & Gourichon, David & C Zody, Michael & Zecchin, William & Vieaud, Agathe & Tixier-Boichard, Michle & hu, Xiaoxiang & Hallbk, Finn & Li, Ning & Andersson, Leif. (2012). The Rose-comb Mutation in Chickens Constitutes a Structural Rearrangement Causing Both Altered Comb Morphology and Defective Sperm Motility. PLoS genetics. 8. e1002775. [10.1371/journal.pgen.1002775](https://doi.org/10.1371/journal.pgen.1002775).
22. Jacqueline P. Jacob, Henry R. Wilson, Richard D. Miles, Gary D. Butcher, and F. Ben Mather. Factors Affecting Egg Production in Backyard Chicken Flocks. <http://edis.ifas.ufl.edu/pdffiles/ps/ps02900.PDF>
23. Chad Zadina, Extension Poultry Assistant Sheila E. Scheideler. Proper Light Management for Your Home Laying Flock. University of Ne-

- braskaLincoln Extension POULTRY C-3, Management Issued October 2004
https://hort.purdue.edu/tristate_organic/poultry_2007/Light%20Management.pdf
24. Amazon Web Services. AWS DeepLens. Get Hands On Experience with Deep Learning with Our New Video Camera.
<https://aws.amazon.com/blogs/aws/deeplens>