Detecting Chickens Using Social Media Data: A Modern Approach to Detecting and Preventing the Avian Flu in Farms

Arpan Ojha
Indiana University
Bloomington, IN
arojha@iu.edu

Brad Cooley
Indiana University
Bloomington, IN
bwcooley@iu.edu

Abstract

Animal detection is a rich, up-and-coming field with research around a black box animal detection idea. There has been a lot of work around how to leverage animal/wildlife detection without much open-sourced focus on how to achieve perfect detection. We present our model as a focus on farm versus game chickens. This research carries significant importance due to the rampant avian flu that leads to culling of thousands of livestock chickens each year. Proverbial chickens are used in physics problems as spherical point objects and they quite often resemble spherical objects with slight aberrations (head and legs). We can leverage this as most often complicated rigid shapes require a lot of complicated tensor operations, which is escaped here; game chickens have a distinctive shape that is less spherical. We propose a detailed survey of various state of the art models such as VGGnet, ViT, and ConvNeXt trained to detect chicken data. We also argue that although our methodology is niche, it can be extended to several species. Through our comprehensive study, we show the difficulties of various models for this niche field.

1. Introduction

At the start of 2022, the United States (US) and Canada began dealing with an avian influenza outbreak causing millions of deaths within chickens and other wild birds [27]. At the University of California, Davis (UC Davis) School of Veterinary Medicine, Dr. Maurice Pitesky's Cooperative Extension Poultry Lab is investigating this trend of outbreaks and researching ways to stop the spread.

1.1. Current approach

Dr. Pitesky and fellow researchers in his lab have chosen to investigate various computer vision models in an attempt to identify different breeds of chickens as they arrive at a poultry farm, specifically backyard chickens and game fowl. The purpose for this classification is because the strain originated from Asia, Africa, and Europe in 2021 [27] and there are various breeds that are predominately imported from countries outside the US, game fowl being one of the most commonly imported. The current approach utilizes an out-of-the-box ResNet-50 model in Python [21] to detect chickens versus other animals primarily using images scrapped from various social media websites. The results so far are fairly promising, but there the model is exhibiting large biases when it comes to other smaller animals.

1.2. Our role

Dr. Pitesky and the researchers at UC Davis are looking for explanations as to why their model isn't performing how they expected as well as potential ways to improve it. After an initial meeting with Myrna Cadena, a graduate student researcher in the lab, our role had also expanded to finding a way to improve the quantity and quality of data the ResNet-50 model [21] was being trained on. The final aim is to detect backyard chickens against game fowl, and eventually, to have a lightweight enough model to apply classify these chickens in a video.

2. Background

This problem itself is a fairly niche one with not may existing literature on the classification of chickens. Due to this, we chose to look at how others have approached classifying different breeds of a singular animal and applying that knowledge to our specific situation.

2.1. Understanding chickens and their shape

Proverbial chickens are used in physics problems as spherical point objects and they quite often resemble spherical objects with slight aberrations (head and legs). We can leverage this as most often complicated rigid shapes require a lot of complicated tensor operations, which is escaped here; game chickens have a distinctive shape that is less spherical. They are bred to have better reach with larger talons, less fat and more muscle mass. Understanding the key differences between a farm raised chicken and game fowl could not be possible without [14], [24].

While our initial approach focused on body shape, [14] gave us the realization that what we should be striving for in classification is differentiation in feather shape and texture. Farm and game fowl have different composition body wise, and that might be the largest difference between the two, but feather shape is the key to distinguishing even more breeds [14] beyond farm and game fowl.

Looking for animals with a generic shape similar to chickens was difficult, but in [16] they examine classification of different marine life. Knowing that general marine life like fish mostly resemble a similar generic shape such as an ellipse, allowed us to utilize some methods from the research. In [16], a new neural network, titled DeCAF, is created to deal with the classification task. Digging deeper into DeCAF, we notice that the foundation of it is a ConvNet module written in Python and pre-trained with ImageNet [17]. We plan to also utilize ConvNet as their results seemed promising for animals/marine life with generic shapes.

2.2. Dealing with social media data

With no great way of collecting chicken image data in a controlled environment to create a good training data set, Dr. Pitesky and his lab resorted to scraping images off of various social media websites like Twitter [13] [20], Backyard Chickens [5], and Craigslist [6].

To deal with the heterogeneity of images posted on social media, it was imperative to find research on the classification of images from broad sets of data. In [23], broad sets of data with the potential of animals in them are used in various classification tasks. The authors specifically targeted low to medium resolution images with the purpose to see how well the classification tasks would perform. A brief explanation on how to deal with images where animals aren't present is also discussed in [23]. Initially, we didn't think this aspect would be important to us, but with data as unpredictable as that found on social media websites, this would be something we need to take into account when training our models; this paradigm creates a "chicken versus the world" model.

Outside of scraping images of animals from social media, it seems important to understand scraping of general image data from social media. In [15], we see the recurring theme of poor quality of data, and [20] supports this theme as well. As mentioned, the challenge of this project lies in the data we choose for model training. Good, well curated data is seemingly elusive, especially when social media is the primary source of the data. Due to time and monetary limitations of the project, we don't have the ability to curate a good data set; we must resort to readily available social media data to complete the project. In [15], the authors had some techniques on how to deal with the variability of social media image data. This understanding would prove crucial to exploring data outside of [13], [5], and [6].

2.3. Deep learning and state-of-the-art approaches

Finally, it seemed important that explore models outside of ResNet-50 [21], as it was having issues with its classification. Part of this could be due to the quality of the data being fed to the model, the number of images being fed to the model, or the model itself. Focusing on the model aspect of the problem, we wanted to explore deeper models than ResNet-50 [21] as deeper models would potentially allow for the understanding of textures and feather shapes. Exploring more techniques, we decided that VGG-19 [26] would be a good experiment for this data based on the traits of VGG-19. We also wanted to explore more state-of-the-art models, like ConvNeXt [22], a paper we also discussed in CSCI-B657. Lastly, we wanted to experiment with ViT, so we referred to [18] and the architecture proposed.

Outside of specific computer vision approaches, understanding how they are implemented in similar situations as ours was key to the development of our research. Using [19], we were able to see how deep models and deep learning can apply to a situation very similar to ours. Looking for a domain closer to ours, [16, 25] deal with marine life and their classification by means of deep learning. The marine life aspect was intriguing to us due to the similar core nature of shape classification that both papers explored. Above all else, these papers allowed us to broaden our knowledge with specific image classification technologies that you might see taught in a class like CSCI-P 556.

3. Methods

The exploration phase of this project involved us exploring multiple models and approaches to both the classification of chickens and the process of gathering data. As we used our research from Section 2, we narrowed down the models and methods we would use in approaching the problem. We have systematically listed our methodology in arriving at our final model.

3.1. Chickens versus the world

Our final model is supposed to identify different breeds of chickens (namely farm chickens from game fowl) using social media data. Breaking this problem down further, we can approach this as a chicken versus non-chicken problem as the data gathered from social media can possibly contain any number of foreign classes. There are two main problems we can envision.

Figure 1. An example of similar animal shape between two different classes. A squirrel with an relaxed tail is very similar in shape to a game fowl.

3.1.1 Unpredictability in data

The world is vast. This challenges the model to make accurate predictions, sometimes without knowing what the actual image is. For example, a model trained on horses versus chickens can only identify those two classes. An image of a bat would potentially confuse the model. While this is a very hard problem to solve fully, our method relies on a discriminative model design. We attempt to make a heavily discriminative model with a similar class and a dissimilar class.

To solve this issue, our model is trained against multiple classes of animals. Utilizing a deep learning architecture, we can account for textures of feathers and fur in addition to differences in the actual shapes; this is the product of Section 2, [16, 19]. Ultimately, this reduces our efforts in segmenting individual images, therefore enabling the model to make two tier identifications. We trained six classes where three classes were similar and three were extremely different from our base classification class, chickens. We aimed to split the deep models to work on two tier classifications. Earlier layers succinctly differentiate between the three similar class group or the other three. Final layers fine tune and identify between the similar classes. This can be made deeper and extended to species level classification.

3.1.2 Shape decomposition

The classification problem becomes challenging when identifying animals with similar shapes. We identified that quite a lot of animals have a lot of common details. For example, a pigeon and a grey pheasant can look similar in terms of shape and color. Additionally, we find sometimes two very different animals assume shapes that are virtually identical to each other, as seen in Figure 1. These issues cause an ”assumed leakage” in the models.

3.2. Data Collection

Data collection was the next major task faced. As Dr. Pitesky and his lab noted, images of solely chickens are quite hard to come by. Producing a large dataset exclusively from Google or using an existing dataset like Animals-10 [4] is not sufficent enough to train a model like they are wanting. We must utilize multiple sources of data to create a robust and strong performing model.

3.2.1 Trap Camera

Collecting animal images through a trap camera is the most efficient way to collect high quality images. Several protected enclosures and parks collect high quality images for studies. Once such example is the Serengeti snapshot project [12]. This is an excellent source to train classifiers on wildlife images. A similar approach can be taken in zoology labs with animals kept in captivity. Unfortunately for our research, due to an ongoing outbreak of the avian flu, the lab had to relocate the chickens in study and they were no longer being captured by the trap cameras. The Cooperative Extension Poultry Lab at UC Davis also had trap cameras set up in their chicken enclosure, but were shutdown due to unknown reasons.

3.2.2 Online

Online scraping of data is the next best way to collect quality images. We collected an initial data set from Animals-10 [4]. This contained roughly 2000 images per class of animal with a total of 10 animal classes. This set of chicken images however consisted of only farm raised breeds. We scrapped Twitter [13], Reddit [9, 10], and Google for other species. We accomplished this utilizing RipMe [11], an open-source web-scraping application. We collected horses, elephants, squirrels, cats, chickens, and dogs, all mainly from the Animals-10 [4] dataset. The quality of images collected from online scraping tends to be quite abysmal. There are several issues with image sizes, scales and types. Often times images labeled chickens had no chickens image in them. This leads to a lot of manual re-labeling and classification, which is a tedious task. However, in order to get the data we needed for our models, we had to do this.

3.3. Pre-processing

As mentioned earlier, we planned on training six classes of animals with progressive differences in shapes and fur/feather textures. In order of most discriminative to least discriminative model, this would be elephants, horses, dogs, cats, squirrels and chickens. We pre-processed all images in the following ways.

3.3.1 Transformations

We applied rotational transformations randomly on all images in our dataset. This transformation introduced nonlocality in trained images. It made our model independent of location of the object in question. Random rotation was applied only at 30 or 45 degrees.

3.4. Resized cropping

We applied a variable crop mechanism to account for various sizes of our class. This variable sized cropping is done at random to avoid our model overfitting on relative classes sizes. It instead focuses on the textures and relative shapes of the data being fed to it.

3.4.1 Resizing

We resized our images to 224 by 224 pixels. This size was ideal for quicker training time, while still preserving good enough accuracy for our needs. It was also sufficiently large for the model to pick up differentiation in textures and other minuscule details.

3.5. Models used

We trained several models based on their affinity to be discriminative and learn feature data. Our initial baseline model was a Convolutional Neural Network (CNN).

3.6. CNN

We trained an extremely basic CNN model to gauge the depth needed to express all sufficient features for a decent animal classification model. We shifted our pixel scales from 255 to 1. A 0.2 shear and 0.2 zoom range was then applied. Additionally, we flipped the images for training. These images were scaled down to a uniform 64 x 64 size and split into batches of size 32 each. A total of 4,660 images were used for training. The initial images were obtained from Google and Backyard Chickens [5]. Our CNN model was 3 layers deep in a sequential mode with a 3 x 3 convolution filter applied to it. We used the ReLU activation function against the input layer of a 64 x 64 x 3 sized image. The next step was pooling and adding with a stride of 2. The second layer consisted of a second convolutional kernel with a kernel size of 3 x 3. We used a filter size of 32 and a ReLU activation function for this as well. This layer fed into a max pooling with a stride of 2. Finally, the output channels were flattened and moved to a full dense connection. The final activation function was again a ReLU, but then followed by a sigmoid.

We used the adam optimizer with a binary cross entropy loss. This was compiled using accuracy metric. Processed training data was trained over 25 epochs before overfitting was observed. Training took about four minutes and testing was complete within a minute on processor 1 [1].

3.7. VGG-19

As an initial model to analyze the importance of depth, we trained a VGG-19 model. The pre-processing steps are exactly as highlighted in Section 3.3. In addition to that, we trained the model using PyTorch [8]. Hence, the processed image was transferred onto a data loader suitable for PyTorch. Data was then split into test, training and validation sets. We used transfer learning of VGG-19 on the PyTorch database to fine tune our student model. Our sequential layer consisted of two linear layers, ReLU and a softmax layer. The linear layer went from 25,088 to 6,000 followed by ReLU then another linear layer from 6,000 to 10 and finally a softmax function smashed to dimension 1. We introduced a dropout of 0.5. The VGG-19 model was trained for 20 epochs with this model having 138 million tunable parameters.

We again used the adam optimizer with a cross entropy loss for errors. This model took 10 hours to train on the Quartz Cluster at IU [2].

3.8. ResNet-50

ResNet-50 uses the image data generated after preprocessing steps obtained from the CNN model. We have a total of 5,334 images belonging to four classes. Training data is split into 32 batches. This model took about an hour to train and five minutes to test a single batch of images. We again utilized transfer learning here using ResNet-50 as our base. In our student network, we first obtain the output and perform global average pooling. This result is passed onto a ReLU activation method which is finally used to predict the classes using softmax. We train our model with an adam optimizer and categorical cross entropy. The training data is trained over 10 epochs till we reach the plateau. The results of this model are exciting as it achieves near perfect accuracy. However it has issues such as training time is large and test classification against unknown images are not too good. It was trained and tested on a MacBook M1 Max processor [2] and in a cloud computing environment [7].

3.9. ConvNeXt

We used transfer learning to train our model from the ConvNeXt-22k architecture. We pre-processd the images by resizing all images to 256 x 256. These images are converted to tensors and normalized to ImageNet default values [17]. The model is used to calculate top-1 and top-5 accuracy of chickens. This model took less than a minute to train and about half a second to run through all the images in a batch size of 32. This model was also tested on a MacBook Pro M1 Max processor [2]

3.10. Vision Transformer

Our Vision transformer follows the same pre-processing steps mentioned in Section 3.3. The ViT model was trained on image of 224 X 224 X 3 size. Initially, we trained the model with 7 patches, 2 heads, and on 5 different classes. This led to less than optimal results. We were forced to

Method	Top-1 Accuracy (%)
CNN (baseline)	75
VGG-19	87
ResNet-50	98
ConvNeXt	94
ViT	35

Table 1. Results for Top-1 Accuracy

Method	Top-5 Accuracy (%)
CNN (baseline)	95.8
VGG-19	89
ResNet-50	99
ConvNeXt	99
ViT	45

Table 2. Results for Top-5 Accuracy

Method	Time to Train / Detect (seconds)
CNN (baseline)	240 / 60
VGG-19	28,800 / 720
ResNet-50	3,600 / 300
ConvNeXt	60 / 30
ViT	600 / 5

Table 3. Results for time to train and detect (classify).

modify this model to 56 patches with 10 heads. We used an adam optimizer with cross entropy loss for errors. This model has 87 million tunable parameters. It was trained over 20 epochs before plateauing was observed. The final accuracy was not comparable to a neural network model however, there was a significant increase (15%) in accuracy as a result of tuning. ViT took about 10 minutes to train on GoogleColab Pro [3]. However, the testing time was less than five seconds. This is something important to note as we strive for a model that can process video feeds in nearly real-time.

4. Results

The results corresponding to each model are as follows:

- CNN models performed extremely well while classifying two animals that are very different. We obtained a 95.8% training accuracy. This also resulted in 85-90% test accuracy against morphologically different animals. However with similar animals or similarly shaped animals, like squirrels and chickens, we get 50-75% accuracy. Shape is expressed properly in earlier CNN layers.

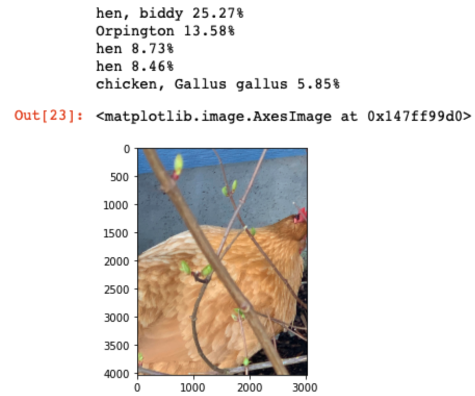

Figure 2. An example of ConvNeXt performing well, even with noisy or poor data.

- VGG-19 has performed better than CNN. With an 89% training accuracy and 87% test accuracy, it outperformed CNN. This suggested us that deeper models learn the texture features much better.
- A consolidated effort was made in ResNet-50, which was both quicker and better than VGG-19 in terms of accuracy and training time. We got a 99% training and 98% test accuracy even with similarly textured and shaped animals. However, the training time was still on the higher side for a video feed.
- ConvNeXt model gave us state-of-the-art performance in training and testing accuracy. We obtained a 99% training accuracy and 94% worst case test accuracy. This was promising because the testing time was small enough for a video feed.
- To get an even quicker training and testing time we experimented with vision transformers. We got a 45% training and 35% testing accuracy with all the data we had for neural networks. This was a significant improvement over earlier ViT model. Although not as good as the neural networks we propose, with additional data we can train a state-of-the-art vision transformer that can outperform ConvNeXt in training time, accuracy and speed.
- ConvNeXt has an added advantage of distinguishing species data. This is the result we were looking for to make social media classification.

4.1. A focus on ConvNeXt

ConvNeXt produced very promising results, even if it's confidence in a particular label was generally low. As mentioned in the results explanation and in Table 1, ConvNeXt was on average 94% accurate when it came to correctly

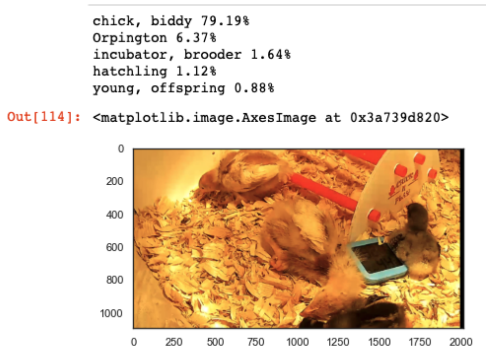

Figure 3. An example of ConvNeXt performing well on semi-good data.

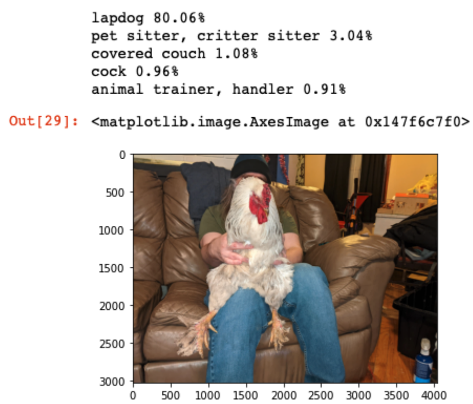

Figure 4. An example of ConvNeXt performing well on semi-good data.

classifying chickens in the image. Even with some of the noisy and particularly poor image data gathered from social media, ConvNeXt performed well, as seen in Figure 2. While the confidence of the label the model chose is relatively low, it still is the first choice it has about what is in the image; for our purposes, that is an excellent sign. There are other times where the model produces a very high confidence, even with subpar data, as seen in Figure 3.

However, there were times where ConvNeXt had it's struggles. As seen in Figure 4, this particular angle fooled the model into thinking there was a dog in the image. This is most likely due to the aforementioned angle, as well as where the contents of this image are mostly seen. A couch with a human holding an animal is most of the time not the human holding a chicken.

5. Discussion

We experimented with five progressively modern models over the same data set with various tuned parameters for maximum gain. The result was extremely supportive in favour of newer models. As we progressively move towards better models the image set of 3,000 images per class is sufficient to obtain maximum throughput from state of the art models. As suspected earlier layers learn the superficial shape features of the object in mind. This is evidenced by the performance of CNN model.

The deeper the model goes we see that further classification between similarly textured animals is possible. In our deepest layer model, ConvNeXt we find species classification possible.

Training time reduces over the same set of images as we venture into newer models. This proves that newer models are far superior in terms of training time, accuracy and speed of detection. Training chickens against a spectrum of similar and dissimilar animals proves to be effective when making a class v/s the world model. This paves a new way to study wildlife classification. We also show, depth of a model can be leveraged to make quicker and accurate predictions while looking at textures.

ViT performance throws a positive outlook of transformer models for video detection. There are a lot of observations to be made with better trained transformer models. We need to expand our study to newer transformer models such as DeiT, Swin, etc.

Pre-processing stage can be better streamlined to suit each model. We need to study the expression of each chicken species in order make a better last stage classification.

6. Conclusion

This paper proposes a state of the art model for classifying chickens in social media data. We made a systematic study of progressively deeper models and proved, deeper models are needed to recognize important animal features. Using these deeper layers, models can learn discriminative features expressively. By comparing the training time we show the newer architectures leverage the best of both accuracy and quicker training. Image pre-processing is essential in prevention of over-fitting . Transformer models need a lot of image data to make classifications as good as a neural network model. However, testing time is much better for transformer models. A better model can be designed by combining segmentation with deep learning models for animal image classification.

Our proposed model can be deployed in a real world environment and help in animal husbandry. This can reduce culling in large scale and help prevent live razing of chickens. This model can also be extended to other animals such as under sea creatures or wildlife. It has the potential to help understand aquatic fauna better.

7. Code and Materials

All code and models as well as the data used to train them can be found at https://github.iu.edu/bwcooley/CV-Final.

References

[1] 16gb core intel i5. 4

[2] 16gb quartz cluster at indiana university. 4

[3] 32gb googlecolab with gpu enabled. 5

[4] Animals10 dataset. 3

[5] Backyard chickens - raising chickens 101. 2, 4

[6] Craigslist. classified advertisements. 2

[7] Macbook m1 with gpu enabled. 4

[8] Pytorch. an open source machine learning framework that accelerates the path from research prototyping to production deployment. 4

[9] Reddit r/backyardchickens. interesting stuff for the small flock community and the humane big flock community. 3

[10] Reddit r/chickens. a place to post your photos, videos, and questions about chickens! 3

[11] Ripme, website media scraping software. 3

[12] Serengeti snapshot project. 3

[13] Twitter. it's what's happening. 2, 3

[14] A. Al-Nasser, H. Al-Khalaifa, A. Al-Saffar, F. Khalil, M. Albahouh, G. Ragheb, A. Al-Haddad, and M. Mashaly. Overview of chicken taxonomy and domestication. *World's Poultry Science Journal*, 63(2):285–300, 2007. 2

[15] Giuseppe Amato, Paolo Bolettieri, Vinicius Monteiro de Lira, Cristina Ioana Muntean, Raffaele Perego, and Chiara Renso. Social media image recognition for food trend analysis. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, page 1333–1336, New York, NY, USA, 2017. Association for Computing Machinery. 2

[16] Zheng Cao, Jose C. Principe, Bing Ouyang, Fraser Dalgleish, and Anni Vuorenkoski. Marine animal classification using combined cnn and hand-designed image features. In *OCEANS 2015 - MTS/IEEE Washington*, pages 1–6, 2015. 2, 3

[17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 2, 4

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. 2

[19] Mads Dyrmann, Henrik Karstoft, and Henrik Skov Midtiby. Plant species classification using deep convolutional neural network. *Biosystems Engineering*, 151:72–80, 2016. 2, 3

[20] Srividhya Ganesan, Rachana P, Dijan Maharjan, Sagar Panth, and Alok Prasad Kurmi. Event detection using geotagged twitter information. In *2022 International Conference for Advancement in Technology (ICONAT)*, pages 1–5, 2022. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. 1, 2

[22] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *CoRR*, abs/2201.03545, 2022. 2

[23] Hung Nguyen, Sarah J. Maclagan, Tu Dinh Nguyen, Thin Nguyen, Paul Flemons, Kylie Andrews, Euan G. Ritchie, and Dinh Phung. Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. In *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 40–49, 2017. 2

[24] Cedric Okinda, Innocent Nyalala, Tchalla Korohou, Celestine Okinda, Jintao Wang, Tracy Achieng, Patrick Wamalwa, Tai Mang, and Mingxia Shen. A review on computer vision systems in monitoring of poultry: A welfare perspective. *Artificial Intelligence in Agriculture*, 4:184–208, 2020. 2

[25] Dhruv Rathi, Sushant Jain, and S. Indu. Underwater fish species classification using convolutional neural network and deep learning. In *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)*, pages 1–6, 2017. 2

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014. 2

[27] Michelle Wille and Ian G. Barr. Resurgence of avian influenza virus. *Science*, 376(6592):459–460, 2022. 1