Automated identification of chicken distress vocalizations using deep learning models

STUDENTS:

1.Anshul Rana(UEM218117)

2.Anurag Kanswal(UEM218118)

3.Arshjot(UEM218119)

ADVISOR:

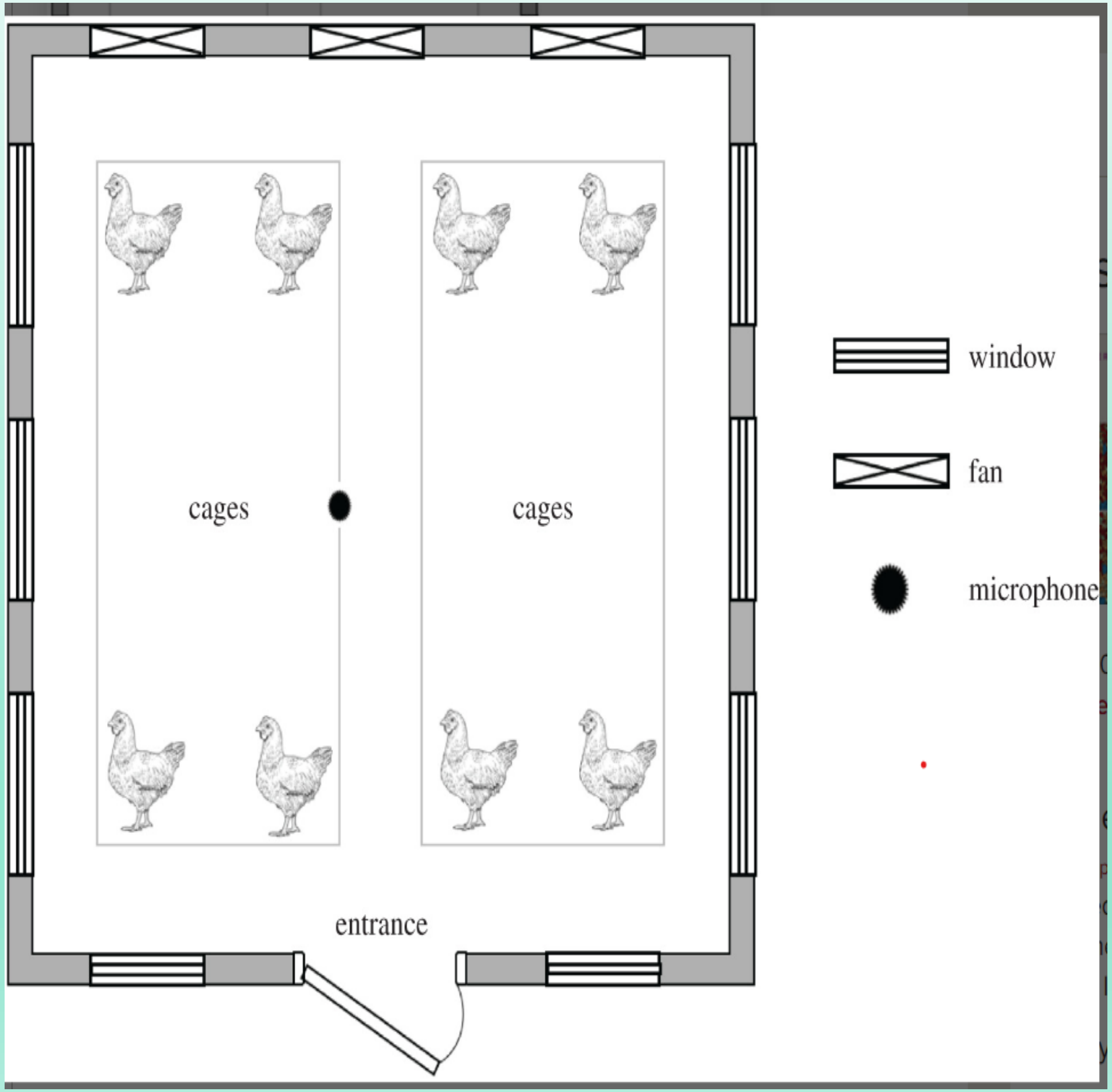
Dr. Veenu Mangat

PROBLEM WE TRAGET

Distress calling triggered by various sources of stress has been suggested as an ‘iceberg indicator’ of chicken welfare However, to date, the process of assessing the number of distress calls produced in large-scale recordings largely relies on manual annotations, which is labour-intensive, time-consuming and prone to subjective judgements of individuals. Thus, it is essential to develop new automated, objective and cost-effective methods for identifying and quantifying distress vocalizations, against a background of other vocalizations and noises that are usually contained in the audio recordings.

DATA SET DETAILS

Recordings were collected in production facilities owned by Lengfeng Poultry Ltd, in Guangxi province, People's Republic of China, in November and December 2017 and in November 2018. Chickens (mix of Chinese ‘spotted’ and ‘three-yellow’ breeds) were kept in stacked cages (three cages per stack, with 13–20 individuals per cage), with approximately 2000 to 2500 birds per house



DATA ANNOTATION

Data Annotation or Data labelling was done using Audacity software. The audio samples collected were divided into 1.5-second segments for annotation purposes.

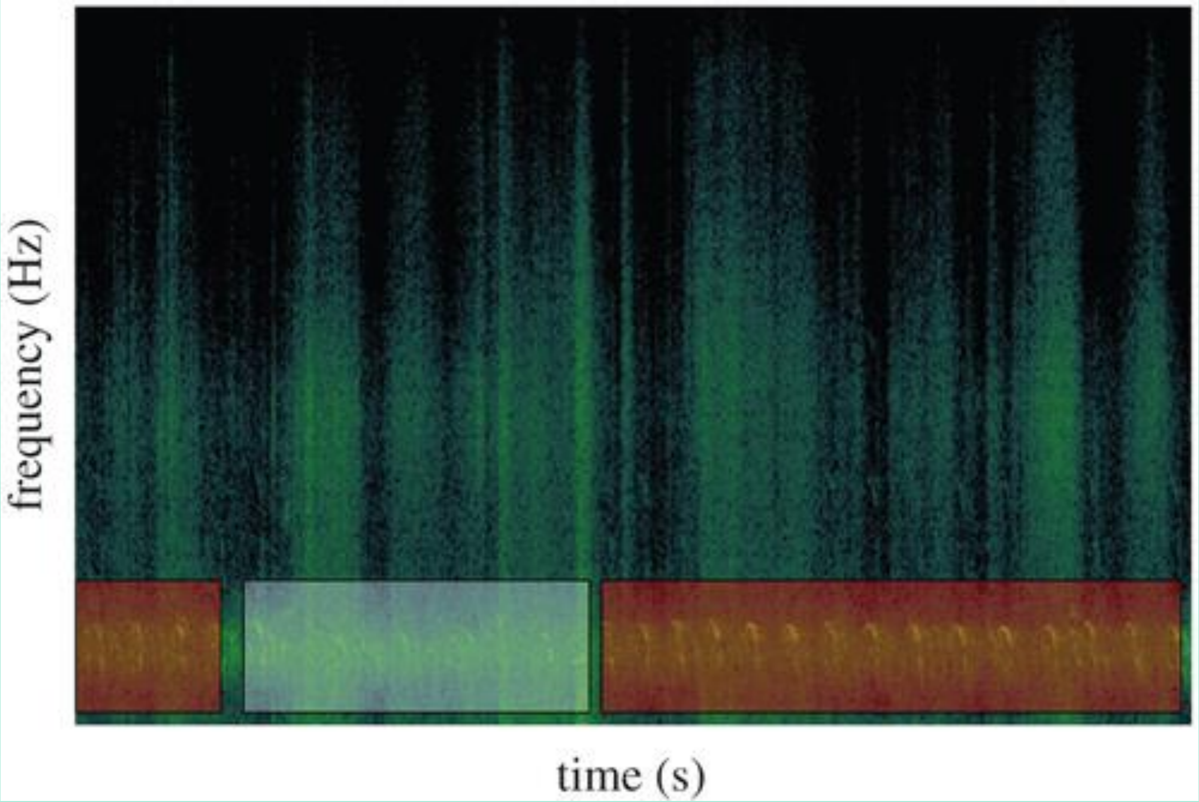


Figure . An example of how the data were annotated. The squares in red highlight the distress calls, while the square in white shows what would be termed ‘natural barn sounds’ (the absence of work sounds and distress calls).

DATA PREPROCESSING STEPS

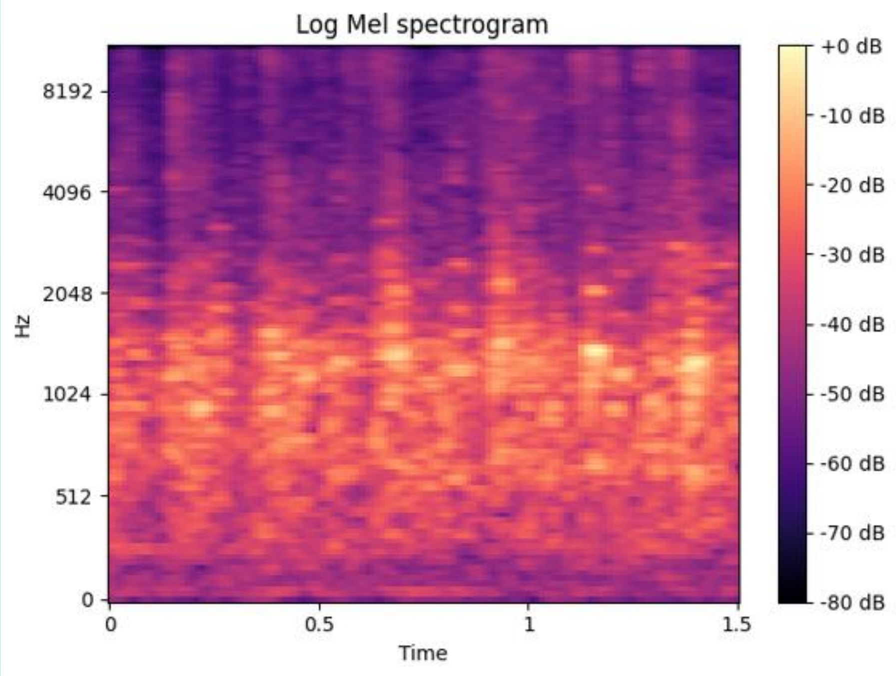
Transform the audio signals into log-Mel spectrograms, which provide a visual representation of sound as time-frequency images.

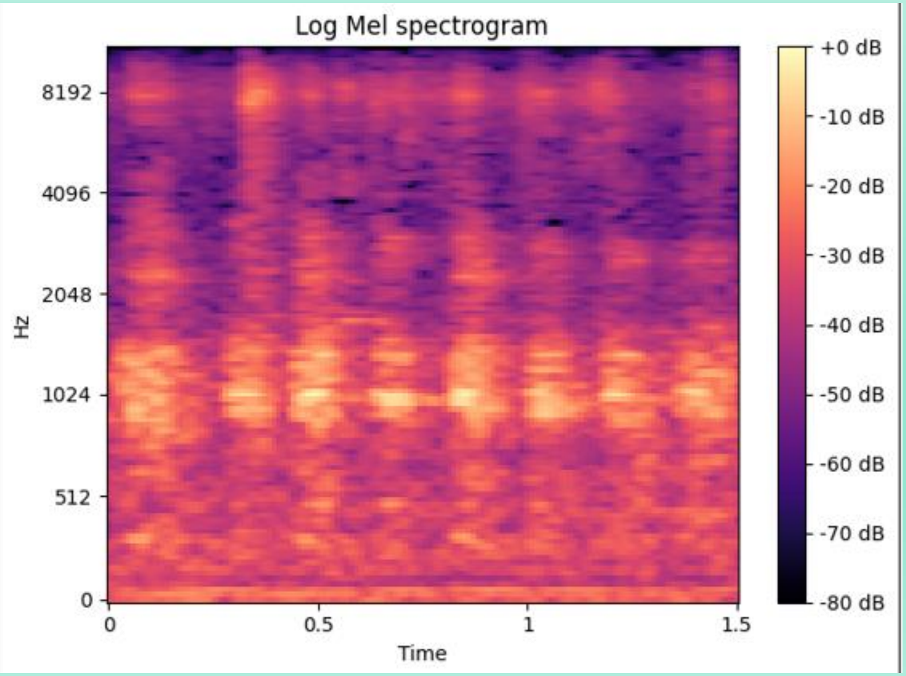
Short-Time Fourier Transform (STFT) is applied to the audio waveforms using a Hann window function

We extract the magnitude spectrum, which represents the amplitude of different frequency components over time.

Convert the magnitude spectrum into a log-Mel spectrogram, we apply a set of 128 log-Mel scale band filters

Label the samples into four distinct categories: alarm calls (722 samples), egg-laying sounds (336 samples), heat sounds (629 samples), and feeding sounds (415 samples).





Log mel Spectrogram examples

DATA AUGMENTATION TECHNIQUES

- TIME MASKING
- FREQUENCY MASKING
- GAUSSIAN NOISE INJECTION
- SPECSAME CLASS MIX METHOD

EVALUATION METRICS

The comprehensive performance of the audio classification model was indicated by the four commonly used evaluation metrics [24,48], i.e. precision, recall, F1 score and accuracy:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\%,$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\%,$$

$$\text{F1 score} = \frac{2\,\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \times 100\%$$

$$\text{and} \qquad \text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100\%,$$

MODEL ARCHITECTURE

The project adopted two types of popular CNN models, namely ResNet50 and InceptionV3, and built a custom model as well. The default input image size for ResNet50 and InceptionV3 models is (224, 224, 3) and (299, 299, 3) respectively. However, the log-Mel spectrograms used have an image size of (128, 130, 1). Also, these models have thousands of classifications of images whereas the project has four classifications. After modifying input and output shapes of these models, the total number of trainable parameters in ResNet50 was 24,053,892 and in InceptionV3 was 22,293,348.

However, these models are complex and possess a challenge to computation efficiency due to their structure and a large number of parameters. For instance, ResNet50 has 50 deep neural network layers and InceptionV3 has 96 deep neural network layers.

Thus, a custom CNN model was created with two convolutional layers. The first convolutional layer has 32 filters of size 3x3, and the second convolutional layer has 64 filters of size 3x3. The output of each convolutional layer is then passed through a max pooling layer to reduce the size of the feature maps. The flattened output of the max pooling layers is then passed through two dense layers with 64 and 4 neurons respectively.

The final layer uses a SoftMax activation function to output the probability of each class. The optimizer used is Adam, which is a popular optimizer for training deep learning models. The loss function used is categorical cross entropy, which is the standard loss function for classification problems. This model has significantly fewer parameters, approximately 3,828,420, and faster detection speed compared to the complex ResNet50 and InceptionV3 models.

PROOF OF CONCEPT

SELF MADE MODEL

In [ ]:

```
# Define the CNN model
self_model = tf.keras.models.Sequential([
    tf.keras.layers.Conv2D(32, (3, 3), activation='relu', input_shape=(128, 130, 1)),
    tf.keras.layers.MaxPooling2D((2, 2)),
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(64, activation='relu'),
    tf.keras.layers.Dense(4, activation='softmax')
])

# Compile the model
self_model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])

# Print the model summary
# self_model.summary()
```

In [ ]:

```
self_model.fit(x_train, y_train,
          epochs=10, batch_size = 32)
```

In [ ]:

```
self_model.evaluate(x_test, y_test)
```

In [ ]:

```
# this model is saved with accuracy of 99.52%, save it again only if accuracy is more than that

# self_model.save('/content/drive/MyDrive/models/12_07_2023/trained_self_model')
```

```
Classification report of ResNet model:
              precision    recall  f1-score   support

       alarm       0.84      0.75      0.79       145
         egg       0.50      0.97      0.66        68
     feeding       0.97      0.45      0.61        83
        heat       0.99      0.96      0.98       126

    accuracy                           0.79       422
   macro avg       0.83      0.78      0.76       422
weighted avg       0.86      0.79      0.79       422

Classification report of InceptionV3 model:
              precision    recall  f1-score   support

       alarm       0.93      0.93      0.93       145
         egg       0.85      0.93      0.89        68
     feeding       0.99      0.84      0.91        83
        heat       0.95      0.99      0.97       126

    accuracy                           0.93       422
   macro avg       0.93      0.92      0.92       422
weighted avg       0.93      0.93      0.93       422

Classification report of Self-Made model:
              precision    recall  f1-score   support

       alarm       0.97      0.97      0.97       145
         egg       0.94      0.94      0.94        68
     feeding       0.97      1.00      0.98        83
        heat       1.00      0.98      0.99       126

    accuracy                           0.97       422
   macro avg       0.97      0.97      0.97       422
weighted avg       0.97      0.97      0.97       422
```

Performance metrics of pretrained and custom made model

TECHNOLOGY STACK

TensorFlow

Keras

AUDACITY

PICKLE
LIBERARY

LIBEROSA

FUTURE ADDITIONS

To enhance the project's effectiveness, an expanded data collection approach would have been preferred. Ideally, gathering audio samples from various breeds of chickens, encompassing both broiler chickens and egg-laying hens, would have been valuable. Additionally, recording sounds over extended periods, such as on a day-to-day basis for more than a week, across different seasons, could have significantly enriched our dataset. This augmentation would have allowed our model to accommodate diverse breeds, incorporate a wider range of sound classifications, and even recognize distinct seasonal variations in the sounds.

A pivotal aspect of our model lies in its potential to assist farmers by detecting alarm and distress signals, alerting them when the chickens are in jeopardy or experiencing discomfort. For example, during situations involving potential threats like animal attacks or intrusion by predators.

The application of using speech recognition to identify respiratory diseases in chickens shows promise. Integrating all these classifications into the model would enhance its effectiveness and create a potential impact on poultry farming and beyond. This solution not only improves animal health but also encourages sustainable and ethical practices in the agricultural industry, creating a positive influence on the welfare of chickens and farmers alike.

THANK YOU