DIGITAL SCARECROW

Multimedia Project Report

Sahil Gupta - 2016CSB1056

Shreyanshu Shekhar - 2016CSB1060

INTRODUCTION

These days technology is being used to solve every kind of problem. It could be related to safety, surveillance, education, sports, transportation, entertainment and agriculture. So we decided to solve a problem which is related to agriculture and we thought what is the main issue faced by the farmers. The answer was crop loss. It could happen due to various reasons like insects, the fertility of the soil, unavailability of water and it could also be destroyed by wild animals and birds. The very primitive solution for this problem which is still being used is the scarecrow. But it doesn't seem to be working quite well these days. So we thought why don't we add some technology into the scarecrow and improve it. So we started researching on the internet for some previous work that has been done and how can we improve on them.

RELATED WORK

In 2006 a team of undergrads at the University of South Florida developed a robotic scarecrow. The "Intelligent Scarecrow" used a microprocessor and surveillance camera to detect incoming poachers and was programmed to discriminate between intruders and farmers. Their assumption was that the farmers were wearing an orange vest. The team developed the system to recognize the colour. If something not coloured orange enters into a 40-foot range, water cannons are triggered, as is a series of prerecorded hawk screams and gun blasts. It also sends a text message to the farmers, alerting them of possible security breaches.

Even though it was a good implementation, it had some limitations:

- It differentiates on the basis of colour
- Water cannons need water and power
- It can not differentiate between animal an person
- The coverage range is very small

Another implementation is also called Digital Scarecrow. The device uses infrared eye which can detect animals within a range of 177,917 sq ft. The compact design and removable tripod allow it to be repositioned quickly and easily. Once the infrared eye spots a bird or critter, the solar-powered scarecrow discharges a series of ultrasonic waves meant to prevent the animal from going any further.

It has excellent coverage area but it uses only one sensor which is infrared to detect the presence of an animal. It might not be reliable every time. It also can not differentiate between animal and human. So if it detects a person and discharges a series of ultrasonic waves on him/her then, those waves might cause hearing damage to him/her.

These were some earlier implementation which had some limitations. So we aim to present our solution which doesn't have these limitations.

IDEOLOGY

The idea of this project is to scare animals and birds away. We have designed a digital scarecrow which identifies the animal or bird and plays the sound of their predator. For example, if our digital scarecrow identifies that deer is present on the farm, it will play the sound of a lion. We are using a camera and a microphone as sensors. A microprocessor to process the input data and a speaker to output the sound of the predator. To power the whole system a battery is required which can be charged by using a solar panel. This idea is cost-effective as well as sustainable. Assembling one scarecrow will require around 2 to 3 thousand rupees which is a one-time investment. It can charge on itself, therefore, it is also sustainable.

METHODOLOGY

The first question comes that why do we need two sensors, can't we use either one of them to achieve our goals. The answer is no, we need both of them to make the system more robust. Sometimes it could happen like the camera can capture the animal or bird but they aren't making any noise so the microphone can not identify them or the camera is unable to detect the animal as they could be hidden in the crops but the microphone can pick their sound and identify them. So using the camera and the microphone makes the system more complete and robust.

We are using two different models because we have two different media input. These models process the input data at every one-second interval. In every interval, one image is captured and one-second audio clip is recorded and sent to their models. They process the input data and returns the label as output. The label could be false if no animal or bird is found and it could be the name of the animal or bird that is detected by the model. Then using those two labels, finally, the sound of the predator will be played. This process will be repeated in every interval.

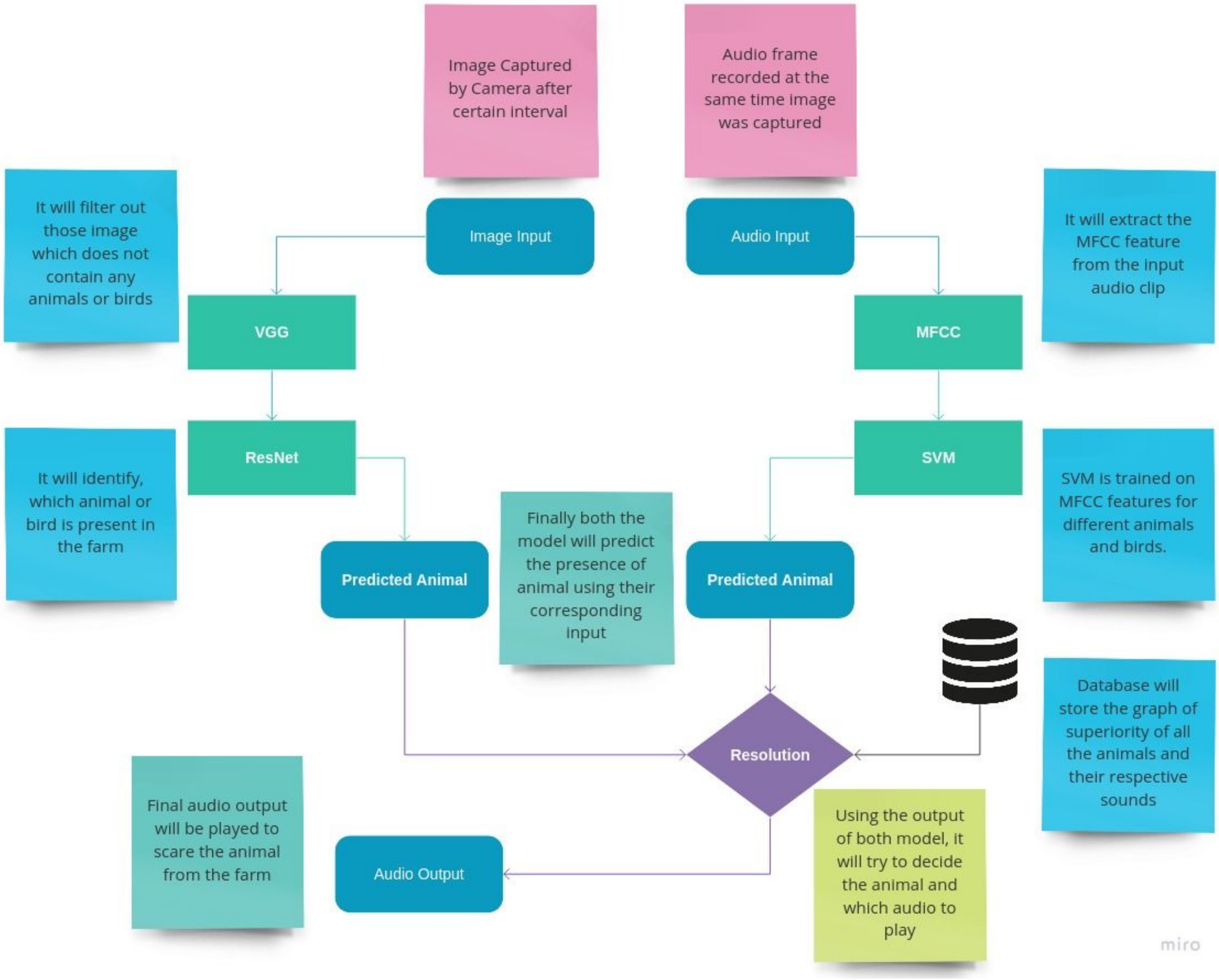


Image Captured by Camera after certain interval
Audio frame recorded at the same time image was captured
It will filter out those image which does not contain any animals or birds
Image Input
Audio Input
It will extract the MFCC feature from the input audio clip
VGG
MFCC
ResNet
SVM
It will identify, which animal or bird is present in the farm
Finally both the model will predict the presence of animal using their corresponding input
SVM is trained on MFCC features for different animals and birds.
Predicted Animal
Predicted Animal
Database will store the graph of superiority of all the animals and their respective sounds
Resolution
Final audio output will be played to scare the animal from the farm
Audio Output
Using the output of both model, it will try to decide the animal and which audio to play
miro

IMAGE MODEL

This model takes an image as input and returns the label as output. The image is coloured. The model used to detect the animal is ResNet and VGG16. These are pre-trained model trained on ImageNet dataset.

ImageNet: It is an image database which is organised according to the WordNet hierarchy. It has more than 14 million images. It has 27 high-level categories and 21,844 sub-categories. For every sub-category, there are on an average 500 images.

VGG16: This model has CNN architecture and has been trained on the ImageNet database with 1000 labels. It takes fixed size 224x224 RGB image and the final output is one of the 1000 labels.

ResNet: This model also has CNN architecture and has been trained on the ImageNet database with 1000 labels. This model deals with the vanishing gradient issue which happens when more and more layers are added to the neural network, the gradient of loss function approaches to zero, due to which training becomes hard. So this model introduces an “identity shortcut connection” or “skip connections” which skips some layers. This way they are able to add more layer than before. This model works very well in object detection and face recognition. It also takes fixed size 224x224 RGB image and the final output is one of the 1000 labels.

When it gets the input image, its first task is to detect if there is any animal or birds present. If animal or bird is detected then the next task is to identify which species of animal or bird is present. To detect the presence of bird and animal in the image it uses VGG16. To identify the species of animal and bird it uses ResNet. The output label of ResNet with maximum confidence value is the final output of this image model.

AUDIO MODEL

When given an audio sample in a computer-readable format (such as a .wav file) of a few seconds duration, we want to be able to determine what animal sound is present in the sample.

Data pre-processing: We identified the following audio properties that need preprocessing to ensure consistency across the whole dataset: Audio Channels, Sample rate and Bit-depth. Librosa is a Python package for music and audio processing that allow us to load audio in our notebook as a numpy array for analysis and manipulation. For much of the preprocessing, we will be able to use Librosa’s load() function, which by default converts the sampling rate to 22.05 kHz, normalise the data so the bit-depth values range between -1 and 1 and flatten the audio channels into mono.

Extract features: The next step is to extract the features we will need to train our model. To do this, we are going to create a visual representation of each of the audio samples which will allow us to identify features for classification, using the same techniques used to classify images with high accuracy. We will be using a similar technique known as Mel-Frequency Cepstral Coefficients (MFCC). They are a useful technique for visualising the spectrum of frequencies of a sound and how they vary during a very short period of time. The main difference is that a spectrogram uses a linear spaced frequency scale (so each frequency bin is spaced an equal number of Hertz apart), whereas an MFCC uses a quasi-logarithmic spaced frequency scale, which is more similar to how the human auditory system processes sound. For each audio file in the dataset, we will extract an MFCC (meaning we have an image representation for each audio sample) and store it in a Panda Dataframe along with its classification label. For this, we will use Librosa's mfcc() function which generates an MFCC from time series audio data.

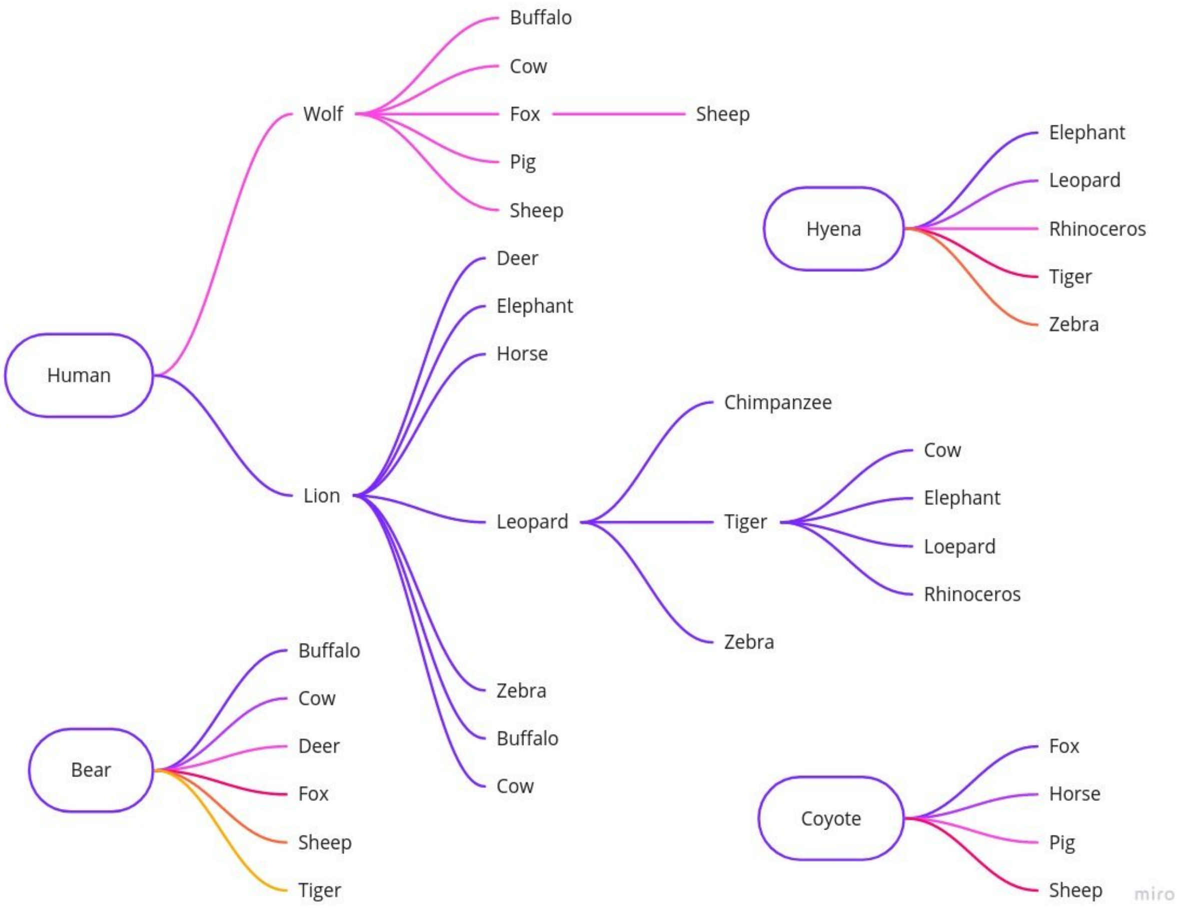
Building our model: We will use sklearn.preprocessing.LabelEncoder to encode the categorical text data into model-understandable numerical data. Then we will use sklearn.model_selection.train_test_split to split the dataset into training and testing sets. The testing set size will be 20% and we will set a random state. The next step will be to build and train a Support Vector Machine (SVM) with these data sets and make predictions.

PREDATOR

At this point, we will get the labels both from audio and video model. Now we want to figure out the predator. As we are getting labels from two different models, there could be few cases. For our implementation, we want a high recall, as we don't want to miss any case where the animal is present on the farm and our system is not able to detect it. So if any of the labels detect any animal, we will consider it.

1. If both the models return false, i.e. both the models are unable to detect anything on the farm so this system won't output any sound.
2. If only one of the model returns the animal or both the model return the same animal then using the predator graph we built, the system will output the sound of that predator.
3. If both the model returns different animal then using the predator graph the system will output the sound of predator which is the common ancestor of both the animal.

Predator Graph: We build this graph which has animals at the nodes and edges denotes the predator relationship. We took all 15 animals found their list of predators using google and built this graph. We use it to find the predators.



DATASET

There are thousands of animal and bird species. We could not use every animal as most of them are not relevant for our use case. So for animals, we decided to use the most relevant 15 animals which we might encounter on the farm. The animals are buffalo, chimpanzee, cow, deer, elephant, fox, horse, leopard, lion, pig, rhinoceros, sheep, tiger, wolf and zebra. For birds we thought, there is no need to add different species of birds. We can treat all the birds the same and use the sound of a gunshot to scare them away. So in our dataset, we have a different kind of birds but we are treating all of them as birds.

IMAGE DATASET

Animals with attributes 2 dataset: It consists of 37322 images of 50 animals classes with pre-extracted feature representations for each image. The classes are aligned with Osherson's classical class/attribute matrix, thereby providing 85 numeric attribute values for each class.

AUDIO DATASET

Due to unavailability of any Animal Sounds dataset on the internet, we had to create a dataset of our own for training and testing. For this, we browsed many websites and collected around 320 sound samples of 18 animals with 20 - 30 audio files for each animal. These files are in various formats available like mp3, wav, etc. Each audio clip varies from 2 - 10 seconds and contains the sound of only a single animal.

RESULTS

	ACCURACY	PRECISION	RECALL
IMAGE	90.14	98.95	90.86
AUDIO	75.00	-	-

Due to less number of audio samples, the trained model is not as accurate as expected to be. Overall, with the combination of image and audio inputs, our model is able to perform fairly well on test data.

REFERENCES

1. https://medium.com/coinmonks/automated-animal-identification-using-deep-learning-techniques-41039f2a994d
2. Image Dataset: Animals with attributes 2: https://cvml.ist.ac.at/AwA2/
3. Audio Dataset is collected from https://www.findsounds.com/animals.html
4. https://medium.com/@mikesmales/sound-classification-using-deep-learning-8bc2aa1990b7