SOLENT UNIVERSITY
SOUTHAMPTON

FACULTY OF BUSINESS, LAW, AND DIGITAL TECHNOLOGY

MSc Applied Artificial Intelligence and Data Science

Academic Year: JANUARY 2022-2023

DETECTING EMOTIONAL CONDITIONS OF DOGS FROM ITS FACIAL EXPRESSIONS USING DEEP LEARNING AND ARDUINO ALERT SYSTEM

Student Name: Sathya Priya Mahendran
Student ID: Q15817059
Supervisor Name: Dr. Jarutas Andritsch
Submission Date: January 2023

This report is submitted in fulfilment of the requirements of Solent University for the degree of MSc Applied Artificial Intelligence and Data Science

SOLENT UNIVERSITY
SOUTHAMPTON

FACULTY OF BUSINESS, LAW, AND DIGITAL TECHNOLOGY

MSc Applied Artificial Intelligence and Data Science

Academic Year: JANUARY 2022-2023

DETECTING EMOTIONAL CONDITIONS OF DOGS FROM ITS FACIAL EXPRESSIONS USING DEEP LEARNING AND ARDUINO ALERT SYSTEM

Student Name: Sathya Priya Mahendran
Student ID: Q15817059
Supervisor Name: Dr. Jarutas Andritsch
Submission Date: January 2023

This report is submitted in fulfilment of the requirements of Solent University for the degree of MSc Applied Artificial Intelligence and Data Science

ACKNOWLEDGEMENTS

My heartfelt thanks to ultimate God for pouring his miracles for the completion of my dissertation project successfully.

I would like to express my deepest gratefulness to Solent University, Southampton for the opportunity to be part of this course MSc – Applied Artificial Intelligence and Data Science, January 2022 – 2023 batch.

I'm extremely grateful to my project supervisor Dr. Jarutas Andritsch for her continual support, encouragement during the research work, feedback, and compassionate supervision.

I would like to express my deepest thanks and sincere appreciation to my Course Leader Dr. Shakeel Ahmad, and Module Leaders Mr. Prins Butt, Dr. Femi Isiaq and Dr. Drishty Sobnath for their guidance and encouragement throughout this post-graduate course.

I could not have undertaken this journey without my husband, my wonderful son and family for their relentless motivation, understanding and enormous love during the course.

Sathya Priya Mahendran

ABSTRACT

The scope of this research is to exhibit dog's emotions can recognized from its facial expressions using deep learning and the detected angry dogs can be alerted using Arduino alert system. In our day-to-day lives, pets that too especially dogs play a major role. They act as a best companion to human beings at many times. The most initial step for every machine learning study is collection of datasets and it is done by using Flickr API. PyTorch and Torch Vision are the popular machine learning framework used for image recognition and computer vision applications which is used to load and train the dataset. Then, Convolutional Neural Network (CNN) is a deep learning algorithm is trained to make predictions on what emotion the dog is experiencing from its facial expressions. The result of this model is shown using a user-friendly Graphical User Interface (GUI) platform. Additionally, an Arduino Alert System is developed to warn angry dogs (i.e.,) whenever the emotion is detected "Angry" in GUI, this alert system has the capability to alert the dog. The main purpose of this dog's emotion detection is to help researchers, new dog owners, dog owners who likes to understand their dog's emotions easily and the Arduino alert system is to save human beings from aggressive dog's attack. The testing accuracy of each emotion like "Angry" is 79.1%, "Happy" is 67.1% and "Sad" is 75.2%. The experimental method of this research paper effectively shows 71.5%.

Keywords: Dog's emotions; Facial expressions; Deep Learning; PyTorch; Convolutional neural network (CNN); Graphical User Interface (GUI); Arduino

LIST OF ABBREVATIONS

CNN – Computational Neural Network

ML – Machine Learning

DL – Deep Learning

IDE – Integrated Development Environment

PPL – Python Programming Language

API – Application Programming Interface

PIL – Python Imaging Library

R-CNN - Region-based Convolutional Neural Network

Timm - PyTorch Image Models

SGD - Stochastic Gradient Descent

GUI – Graphical User Interface

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SECTION 1: INTRODUCTION

This section talks about the fundamental and succinct ideas about the topic, research questions and scope of the research.

SYNOPSIS

The world of artificial intelligence and deep learning are booming with new faces and challenges day after day. It has started to play a vital role in everyone's life The global market for Machine learning (ML) solutions is rapidly increasing, forcing business leaders to ramp up their efforts towards implementing ML in their businesses to not fall behind the competition (Anon. n.d.).

In our day-to-day life, our pets play a significant role, especially dogs. Pets can diminish strain, nervousness, and despair, ease lonesomeness, embolden fitness and liveliness, and even progress your cardiovascular well-being. Compassionate for a pet can benefit kids who nurture up further protected and energetic. Added to this, it offers cherished companionship for elder adults. The research will be particularly valuable used for individuals for those people living unaccompanied, people who can't mingle with other people much, and those who are living as single. Pups are essential to the human being in countless means. The association between dogs and human beings goes back thousands of inordinate lengths of time. Pets like dogs can be able to aid people to circumnavigate their sphere. It can even support blind people and guide them. They can support people suffering from diabetes or confiscations. Many service dogs can alert their owners in times of emergencies and are ready to act. Service dogs are nothing but assistance dogs which can sort out a variety of real-world tasks for individuals as well as support their liberation and self-reliance. Even some of the dogs bring food from refrigerators to their owners. Studies by Harvard University and Kansas State show that dog interaction with humans can help combat challenges associated with some form of heart disease and autism. It exists extensively known these days that pets are intelligent to experience emotional conditions. This research is worth investigating because, in many countries, many life-threatening cases happened because of aggressive street dogs. This research can be a solution since this model detects aggressive dogs and saves human lives.

But still, people face difficulties in understanding their pet's feelings. This investigation broadsheet defines a feeling recognition structure for dogs that by design identifies dogs' feelings such as anger, fear, happiness, and relaxation. The chief possibility of the research assignment is to study if a machine learning method can identify animal reactions as of its facial expressions or body postures which help to conclude its emotions that benefits people who are without appropriate familiarity or competence to understand animals' aggressiveness otherwise kindliness can understand their emotions. And I am planning to include microcontroller hardware with this model which can identify aggressive dogs and benefit saving humans from dogs' attack.

RESEARCH QUESTION

1) Does a deep learning technique can be able to predict dog's emotions using its facial expressions?
2) Does a hardware device like Arduino Uno microcontroller can be able to detect angry dogs and produce alert signals?

AIMS AND OBJECTIVES

The fundamental goal of this study is to detect dog's emotions like happy, sad, and angry from its facial expressions using deep learning algorithm namely Convolutional Neural Network (CNN) and PyTorch which is an open-source ML framework based on the Python Programming Language and Torch library. This software will be mainly helpful for the new dog owners who want to learn their dog's emotions easily and the owners who are unable to understand their dogs' expressions. The output for this software artefact is implemented by means of Webpage where you can straightaway upload the picture of your dog to understand their emotions or there is an option with live video feed where you can directly capture your dog and detect its emotions.

Additionally, I have developed an alerting system for aggressive dogs. This system has the capability to produce alert to dogs by means of noise signals whenever the output predicts angry dogs with its facial expressions. The noise signals are produced by a piezoelectric buzzer which is set with repeller frequency 30kHz. This noise signal will be heard only by dogs and chase them away. Added to, this hardware system is designed in a way to only create alert for aggressive dogs, the main purpose of this hardware system is

for the safety of human beings from dog's attack. For the design of this hardware device, we have used an Arduino compatible Wi-Fi board based on ESP8266EX.

The main objectives of this research are:

- Collection of suitable dog's images dataset based upon their emotions from open-source platform Flickr using a python package Flickr API.
- Data pre-processing is done by cleaning, re-sizing, and removing irrelevant images which helps to check the quality of the dataset.
- Deep learning algorithm namely CNN and python packages like PyTorch and torch vision are used to analyze the images and predict three types of emotions: happy, sad, or angry.
- We will then load the pre-processed images, build a convolutional neural network, and then evaluate the performance.
- We have used HTML and CSS and shown the output in the local webpage. In the webpage, we have introduced two options either by uploading the image or can use the live video feed option to predict dog's emotions.
- Finally, we have used an Arduino compatible Wi-Fi board based on ESP8266EX to create alert in the form of noise signals when the output in the webpage detects angry dogs.

SOCIETAL IMPACT OF THE RESEARCH

In today's world, dogs have been a best companion to human beings. Most of the people prefers to be with a pet rather than to be with other human being because (1) Pets especially dogs listen to our problems and don't argue with us; (2) They show unconditional love; (3) They don't judge us; (4) They are always loyal to us; (5) Most importantly dogs are selfless. In many cases, dogs seem to be better acquaintance to human being (Anon. b). Almost 27% of people in UK have dogs as their pet with an approximate populace of 10.2 million pet dogs (Anon. n). Emotion recognition systems has already been captured much attention by the society, but emotion detection system of dogs will be somewhat new to their knowledge. Dogs being a good friend to a person, then why can't we give importance to its expressions and emotions. This project will be very useful for new dog owners, owners who are unable to understand their dog's emotions, researchers and scientists who

are into various disciplines from neuroscience to animal welfare science, and importantly areas where there are many street/homeless dogs which can anytime attack human beings.

New dog owners will not be able to understand their dog's emotions whether they are happy or sad, at that time, this research will be useful for them to detect their emotions. Dog owners sometimes feel difficult to understand few emotions of their dogs like angry or relaxed, for such cases also, this project will be helpful for them. In this project, a baseline is given to detect the angry dogs to protect human beings from attack. A full-fledged device can be modelled and fitted in the places where many street/homeless dogs are wandering around, once the angry dogs are detected, the device have the capability to threaten or chase them off by means of an electronic noise signal. This will be helpful to safeguard human lives from aggressive dogs.

It will be useful in a place like confined animal shelter where dogs can encounter a varied kind of emotions that deeply impact their emotional wellbeing in the short term and may also have long term affects. This is particularly true if they become sensitized to people, places or things that trigger their fear or frustration on an ongoing basis, or when stress and its associated negative emotional states are prolonged. Detecting their emotions are important and there are various applications that can be simply applied in our daily life based upon their emotions which can even save many dogs from getting harmed. It can be useful for predicting the pain level intensity in dogs, animal protection by animal rescue teams, this will help the communication easier between dogs and human beings. Dog's emotion detection systems can be installed in the security cameras at homes and workplaces, which can predict the change in behavior of pets. Weird behavior can indicate the owner about the mis happenings or threats happening around them or inside their houses and immediate action can be taken (Singh *et al.* 2020).

PROJECT PLAN

The image underneath depicts the Gantt chart. It is primarily a project storyboard which tells us the whole agenda of the complete duration of the project plan from start of the research, implementation of the methodologies and the submission of each task of the study. A Gantt chart is a form of bar chart that epitomizes a project plan. This plan grades the jobs to be concluded on the perpendicular axis, and time breaks on the parallel axis.

Gantt charts demonstrate the initial and end dates of the terminal features and summary features of an assignment (Wikipedia Contributors 2019b).

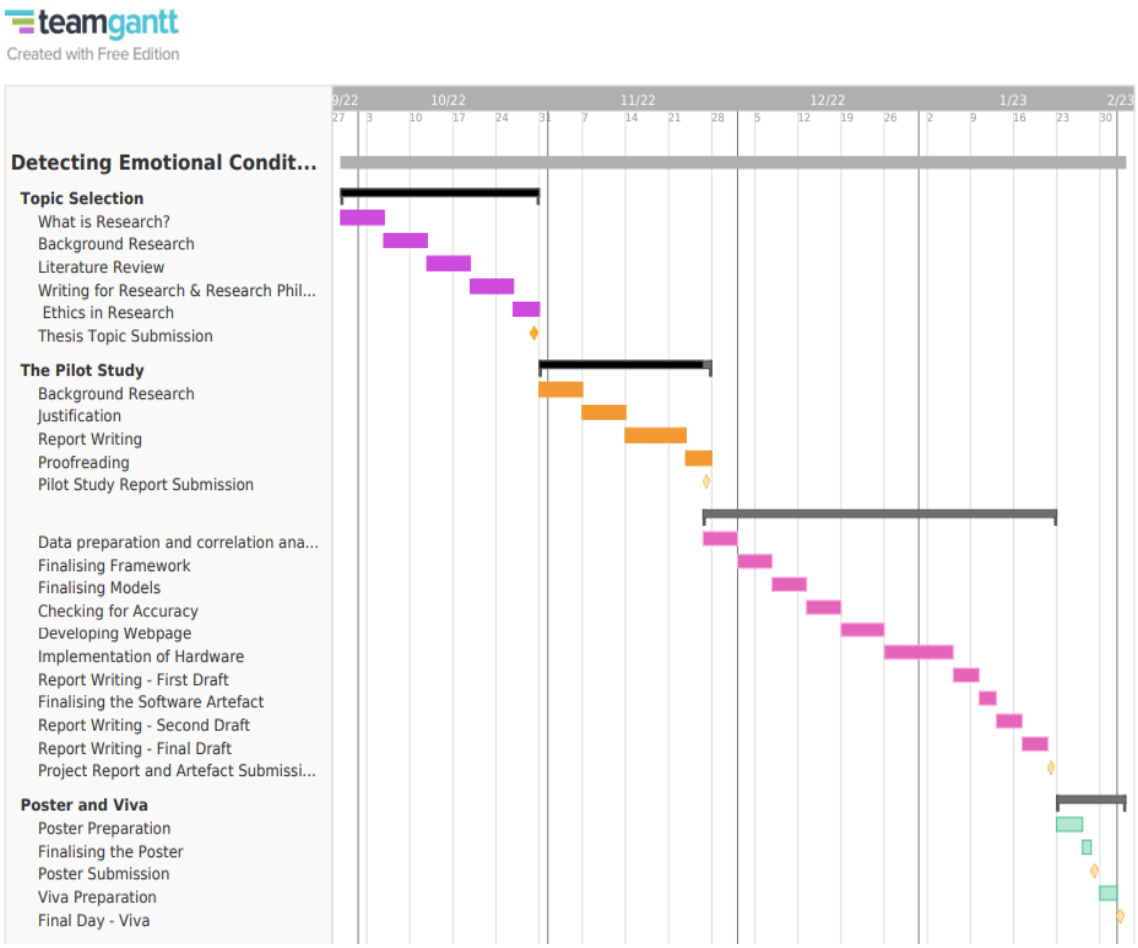


Figure 1. Project Plan - Gantt Chart

Succeeding this introductory section, comes next the main sections that emphases on reaching the goal of the project by covering the objectives of the research.

• Section 2: Literature Review, talks about the preceding works, latest and different types of techniques and results.

• Section 3: Methods and Implementation, explains about the implementation method and adopted approaches.

• Section 4: Results, exemplifies the effects achieved for the execution.

• Section 5: Discussion, validates the outcomes for developing conclusions.

• Section 6: Conclusion, summing up the scope of the project.

• Section 7: Challenges and Limitations, clarifies about the challenges faced in this existing research and possibility for upcoming extension are stated.

SECTION 2: LITERATURE REVIEW

The study is segmented into following categories to attain critical insights into the focus areas identified through this study which are relevant to answer the research question.

1. Emotion classes used in the emotion recognition papers
2. Data sources and data pre-processing activities used in the literature
3. Data analysis tools and visualization techniques used
4. Machine Learning models used and comparison of their results
5. Limitations of studies and recommendations for future

The papers used for the Literature study are taken from various databases like Google Scholar, Multidisciplinary Digital Publishing Institute (MDPI), Wiley.com, Researchgate.net, Science Direct, and Semantic Scholar.

Table 1. PRISMA flowchart from systematic literature search (Edanz-Learning-Team 2022)

Identification	Articles from the database search = 20	Additional articles from other resources = 5
	Articles after removing duplicates = 15	
Screening	Articles screened by title and abstract = 3	Articles excluded with reasons = 3
Eligibility	Articles accessed by full text = 6	Full text articles excluded = 3 These papers have dog's emotions research but not based on deep learning techniques.
Inclusion	Articles included in qualitative analysis = 6	

1) This research paper was on detecting dog's emotions by applying latest deep learning methods to categorize (positive) anticipation and (negative) frustration of dogs where the dataset is collected in a controlled experimental setting. They investigate the correctness of numerous strengths like ResNet, ViT under various observations to their project, and found the elements of a self-supervised ViT (DINOViT) are excellent compared to the other techniques. This work was the first project to focus on the automated classification of canine emotions on data

acquired in a controlled experiment. They achieved accuracy of 87% for training data and 85% for validation data using DINO-ViT technique (Boneh-Shitrit *et al.* 2022).

2) This research paper has used Mask R-CNN deep learning technique to detect dog's images and generate contour mask maps. They have used posture analysis algorithms to obtain position data and noise waves are transformed into spectrograms as features. They have mel-frequency cepstral coefficients for feature extraction, coupled with a Gaussian mixture model-hidden Markov model for voice recognition. The experimental method of this research paper effectively improves the accuracy for posture analysis is 90.7%, state recognition is 85.71% and sentiment analysis is 33.33% (Tsai and Huang 2021).
3) This research paper has used CNN deep learning method and Alexnet for image recognition, segmentation, detection, and labelling. The experimental method of this research paper effectively shows in an average of 86.96% strongly correct decisions, and only 5.43% weakly correct (Franzoni *et al.* Oct 14, 2019).
4) This paper describes an emotion recognition system for dogs automatically identifying the emotions anger, fear, happiness, and relaxation. A new dog key point detection model was built using the framework DeepLabCut for animal key point detector training. The newly trained detector learned from a total of 13,809 annotated dog images and possesses the capability to estimate the coordinates of 24 different dog body part key points. They have used neural network, decision trees, to predict dog's emotions and posture analysis. Multiple models were developed to determine a dog's emotional state based on input images with an accuracy between 62.5% and 67.5% (Ferres, Schloesser and Gloor 2022)
5) This research paper describes about the classification of the species type and emotion of pets. They have explored several deep learning models including YOLO and Faster-RCNN. They have included exploration of models such as Xception, VGG16, ResNet-152, Inception-v3, InceptionResNetV2 and MobileNetV2 together with a detailed exploration of training parameters such as the batch size, loss function, kernel size, dropout, and pooling layers and different fully connected layer options. Eventually the finalized model was based on YOLO due to the accuracy and speed of classification. They delivered the result as a standalone mobile application.

The overwhelming feedback on both applications has been positive with ratings of 4.7/5.0 on the Apple app store (R. O. Sinnott *et al.* 2021).

The conclusion behind this literature review is predicting dog emotions research work will be done by many people since it can be a significant focus in fields varying from neuroscience, psychopharmacology, pain research and animal welfare science (Mendl, Burman and Paul 2010). It will be useful in a place like confined animal shelter where dogs can encounter a varied kind of emotions that deeply impact their emotional wellbeing in the short term and may also have long term affects. This is particularly true if they become sensitized to people, places or things that trigger their fear or frustration on an ongoing basis, or when stress and its associated negative emotional states are prolonged. Detecting their emotions are important and there are various applications that can be simply applied in our daily life based upon their emotions which can even save many dogs from getting harmed. It can be useful for predicting the pain level intensity in dogs, animal protection by animal rescue teams, this will help the communication easier between dogs and human beings. Dog's emotion detection systems can be installed in the security cameras at homes and workplaces, which can predict the change in behavior of pets. Weird behavior can indicate the owner about the mis happenings or threats happening around them or inside their houses and immediate action can be taken (Singh *et al.* 2020).

Table 2. Related Papers related to Dog's Emotions Recognition

Project Title	**Authors**	**Algorithms**
Categorization of emotions in dog behavior based on the deep neural network.	Zdzisław Kowalczuk, Michał Czubenko and Weronikȧ Zmuda-Trzebiatowska, 2022.	Deep neural network called mVGG16 is used.
Predicting Dog Emotions Based on Posture Analysis Using DeepLabCut.	Kim Ferres, Timo Schloesser and Peter A. Gloor, 2022.	Neural Network and Decision Tree with few calculated pose metrics are used.

Sentiment Analysis of Pets Using Deep Learning Technologies in Artificial Intelligence of Things System.	Ming-Fong Tsai and Jhao-Yang Huang, 2021.	Mask R-CNN and Faster R-CNN are used.
Deep Learning Models for Automated Classification of Dog Emotional States from Facial Expressions.	Tali Boneh-Shitrit, Shir Amir, Annika Bremhorst, Daniel S. Mills, Stefanie Riemer, Dror Fried and Anna Zamansky, 2022.	Mask R-CNN, Supervised and DINO ResNet, ViT are used.
Assessment of the Emotional State in Domestic Dogs Using a Bi-dimensional Model of Emotions and a Machine Learning Approach for the Analysis of its Vocalizations.	Humberto Pérez-Espinosa, Verónica Reyes-Meza, María de Lourdes Arteaga-Castaneda, Ismael Espinosa-Curiel, Amando Bautista and Juan Martínez-Miranda, 2017.	Support Vector Machines is used.
Run or Pat: Using Deep Learning to Classify the Species Type and Emotion of Pets.	Richard O. Sinnott, Uwe Aickelin, Yunjie Jia, Elizabeth R.J. Sinnott, Pei-Yun Sun and Rio Susanto, 2022	Faster R-CNN, Mask R-CNN, You Only Look Once (YOLO) and Single Snapshot Detection (SSD) are used.

SECTION 3: METHODS AND IMPLEMENTATION

SYSTEM ARCHITECTURE

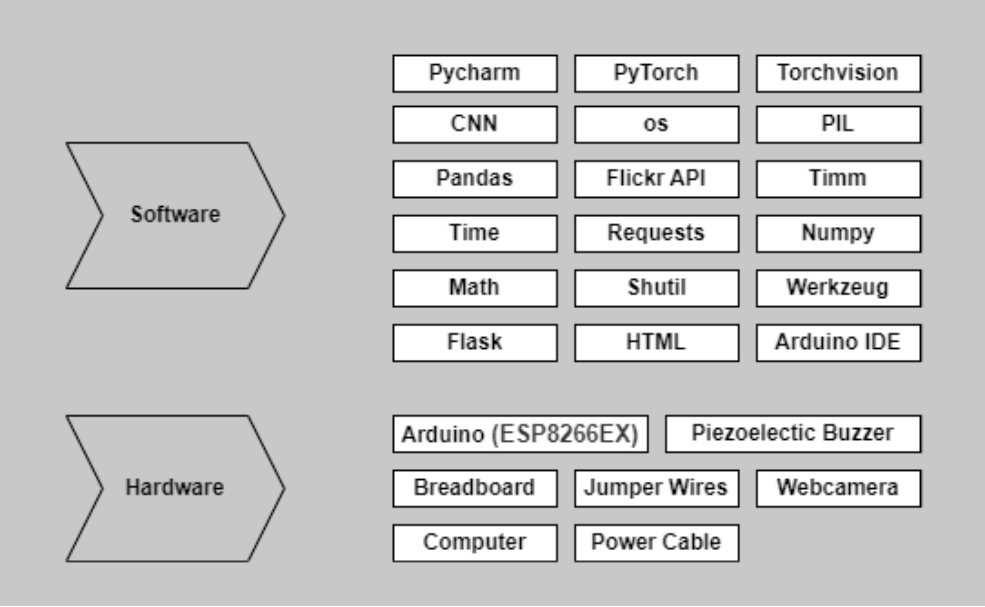


Figure 2. System Architecture

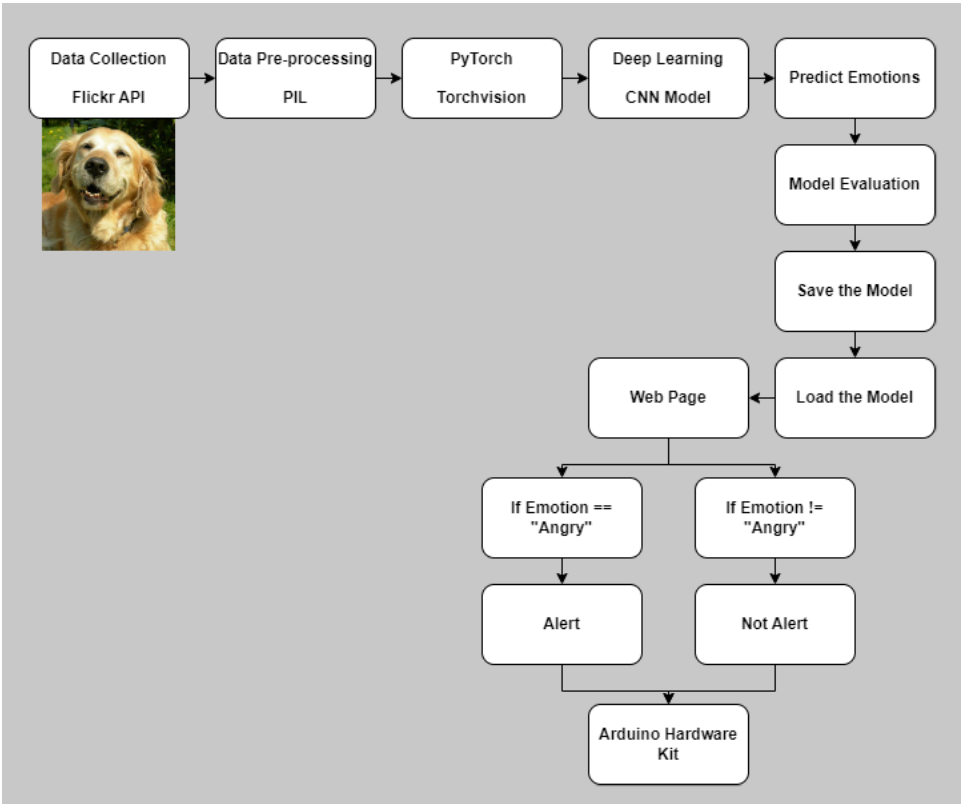


Figure 3. System Flowchart

DATASET COLLECTION

Dataset is collection of data which is used to train and test the machine learning model. It turns to be a good example to educate the machine learning process how to make it to predictions. The datasets are in the form of various types like text, image, audio, video, and numeric data. Many ML projects can go wrong due to several reasons, one of the major reasons will be usage of poor datasets. Selecting the proper dataset for the ML project is one of the most critical preliminary phases of positively creating and deploying an ML model (Shehmir Javaid).

Dataset for the project has been downloaded from an open-source platform namely Flickr using a python package called Flickr API. Flickr API is the extremely complete, and actively built Python interface which is very easy to use. It comprises of technical support for both authorized and non-authorized admission, uploading and switching pictures, and all Flickr API tasks (Stuvel). Flickr is a platform which contains massive collections of images. As this project research is completely based on the images, by means of an API key, we have used Flickr to download relevant images for the research. Flickr can be used by both Free and Pro members. My research dataset consists of 3 major emotions of dogs as variables such as angry, happy, and sad. In each variable, there are almost more than 2000 images to train and test the system.

The emotions of the dogs are downloaded by specifically mentioning in the syntax to search for the emotions (i.e.,) search_field = emotion + " dog" that helps to picks images based on its emotions and facial expressions from Flickr platform, each image was also analyzed manually based on their facial expressions. The dog's emotions are analyzed manually by studying facial expressions as a system of communication between conspecifics. Eyebrow and ear movements of dogs are majorly examined parts to recognize and interconnect aversive circumstances (stated to as an occurrence triggered by a harmful provocation such as ache, anguish, or any negative emotion). For good example, flattening the ears or modifying the opening of the eyes during challenging stimuli. In the same way, the movement of the lips, such as drawing forward the short lip and retraction of the labial commissure (in the long lip), communicate menacing states proportionally to the degree or intensity of the animal's reaction (Mota-Rojas *et al.* 2021). Happy dogs can make signs like delicate, well-behaved, perked-up cozy relaxed ears, wide-open soft-shaped merry-looking eyes, wriggling dog showing you their belly and tongue, slurpy pats and an exposing smile.

Dogs smile can be identified easily by its mouth relaxed or slightly open; the corners of the mouth are pointed upwards and in some instances some teeth may be visible but not in an aggressive way (Anon. a). Sad dogs can look alike keep their gaze lowered, no eye contact (Anon. m) or cast down eyes, and their heads will return to a lower position (Walden 2019). Angry dogs can show characteristics like forward or close to head position ears, pricky or staring eyes, open mouth with exposing teeth bared in a snarl and jaw snapping (Anon. 2019b).

```
API_KEY = "d7d6ef2e1bccccdc6dc15b310337ac41"
API_SECRET = "42905369ea168c42"

IMAGE_DIRECTORY = "dogs_images"
MIN_SIZE_IMAGES = 384
FOLDERS = ["happy", "sad", "angry"]
VAL_EXTENSIONS = ["jpg", "png", "gif", "jpeg"]
TAR_EXTENSION = "jpg"

DEVICE = "cpu"

import os
if os.path.exists("private.py"):
    from private import *

from flickrapi import FlickrAPI
import time
import os
import requests
import math
```

Figure 4. Packages used for Downloading Data from Flickr API

Table 3. Characteristics of Dog's Emotions (Kowalczuk, Czubenko and Żmuda-Trzebiatowska 2022)

Dog's Emotions	Head	Eyes	Ears	Mouth
Angry	Tilted	Open (or) squared	Tilted sideways	Open (or) visible teeth
Happy	Raised	Open	Raised	Open (or) visible tongue
Sad	Lowered	Closed (or) lowered	Relaxed	Closed

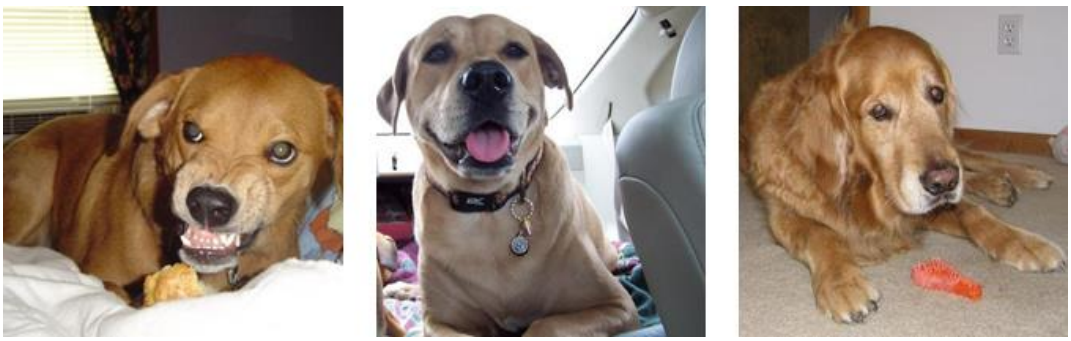


Figure 5. A few sample images from the dataset: Angry (Left), Happy (Center), Sad (Right)

DATA CLEANING AND PRE-PROCESSING

Data pre-processing is a traditional ML saying which delivers a prominent ML fact, unless the input data are of high quality and relevant data, even the best computer vision architectures will deteriorate. It comprises of the procedures needed to configure images prior to model training and evaluation. This also includes data cleaning, data resizing, data orienting, and image color corrections. Also, it is a necessary stage to cleanse image data prior to implementation in computer vision prototype. Downloaded images are cleaned with the help of a Python package named PIL – Python Imaging Library. It enhances image processing abilities to the Python interpreter. This package supports file format extensions, an effective internal interpretation, and strong image handling capacity. The core image library is aimed for speedy approach to data accumulated in a limited basic pixel format. It will provide a strong base for a common image processing tool (Anon. i). We have also used python packages like time, requests, and math. The Python time module supports various methods of indicating time in code, such as objects, numbers, and strings. It as well supports functionality more than indicating time, like expecting for the duration of code execution and evaluating the efficacy of your code (Anon. e). The requests module permits you to send out HTTP requests with Python. The HTTP request reverts a Response Object with every bit of the response data (Anon. k). The math module is a traditional module in Python and is constantly accessible. To make use of mathematical functions, you can import the module math which provides access to C library functions (Anon. c).

```
from settings import *
from flickrapi import FlickrAPI
import time
import os
import requests
import math
```

Figure 6. Python Packages for Data Downloading

Data cleaning is the procedure done in this research for the purpose of removing irrelevant images based the scope. It has done by the following packages of Python: Shutil and PIL.

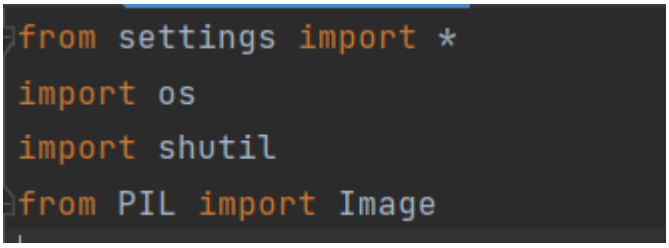


Figure 7. Python Packages for Data Cleaning

The package Shutil is a module that proposes high-level functions on images and collection of images. Operations that are offered helps for image copying and elimination (Anon. o). For functions on individual images, os module is used. The PIL is an incredible Python package utilized for image processing. This package supports numerous features for operating images using Python. It is majorly used in this research to help wide-ranging image formatting, professional image interpretation, resizing images, applying filters to images and requires few robust image handling facilities (Anon. j)

Removing data is a step followed to eliminate the images which are not containing dog's images. This procedure is done with the help of package Timm. PyTorch Image Models (timm) is a library for state-of-the-art image classification, containing a collection of image models, optimizers, schedulers, augmentations and much more (Hughes 2022).

DEEP LEARNING MODEL

DATASET PREPARATION

We are going to train a deep learning model CNN using PyTorch that can predict emotions from the collected images. We will be evaluating its performance and try to boost the accuracy. The dataset collection is done using the Jupyter Notebook, which is an original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience (Anon. 2019a). The collected dataset is specified as "IMAGE_DIRECTORY", this directory will create a folder called "dog_clean_images", from where the images will be pulled to train the model. Then, the folders inside the directory are defined with emotions namely angry, happy, and sad. And then the images are stored in its respective folder, where each emotion will take in account as binary values (0, 1, 2). As per this project research code, angry emotion is assigned as 0, happy emotion is assigned as 1 and sad emotion is assigned as 2. The reason behind

assigning binary values are machines normally works on a code, it cannot predict based on the strings.

```
IMAGE_DIRECTORY = "dog_clean_images"
MINIMUM_SIZE = 384
FOLDERS = ["angry", "happy", "sad"]
```

Figure 8. Pulling Images

The basic procedure to kickstart the deep learning library PyTorch is by defining the device. Devices can be of several types based on the computer operating system, for example, MAC system users can set up “mps”, Nvidia GPU users can set up “cuda” and windows users can set up “cpu”. The system used for this project is windows operating system, so the device is set up as “cpu”.

```
DEVICE = "cpu"
```

Figure 9. Device Setup for Windows Operating System

Next, dataset is to be loaded, emotions must be labelled and associated with each dog image. For that purpose, a list called “image_files” is created which hold path to each image; a list called “image_labels” is created which will hold emotions; a list called “image_code” is created which implements text labels to an actual binary code (0, 1, 2) since the code cannot predict text.

```
image_files = []
image_labels = []
image_code = []

for folder in FOLDERS:
    file_name = os.path.join(IMAGE_DIRECTORY, folder)
    for im in os.listdir(file_name):
        image_path = os.path.join(file_name, im)

        image_files.append(image_path)
        image_labels.append(folder)
        image_code.append(FOLDERS.index(folder))
```

Figure 10. Syntax for Creating Lists and Loop

Then, a package called Pandas is imported to create a data frame in which image_files, image_labels and image_code lists are pulled. Pandas are a top, rapid, dominant, malleable, and easy-to-use open-source data analysis and manipulation tool, built on the Python programming language. As shown in Figure 10, 9560 individual images are loaded and for each image we have the label based on emotion that the dog in the image is showing whether it is angry, happy, or sad.

```
dataset = pd.DataFrame(dict(filename=image_files, label=image_labels, code=image_code))
print(dataset)
```

Figure 11. Syntax for Pandas Data Frame

```
                                                filename  label  code
0      dog_clean_images\angry\100035066_bb2792d5c8_b.jpg  angry     0
1      dog_clean_images\angry\10069504164_436856871c_...  angry     0
2      dog_clean_images\angry\1007803967_b4b8ebc62c_b...  angry     0
3      dog_clean_images\angry\10131474585_997d88e066_...  angry     0
4      dog_clean_images\angry\10222334643_1153586191_...  angry     0
...                                                  ...    ...   ...
9555    dog_clean_images\sad\9953954964_4330fdf081_b.jpg    sad     2
9556    dog_clean_images\sad\9959873844_67dce9ca9f_b.jpg    sad     2
9557    dog_clean_images\sad\9982050784_3765f84803_b.jpg    sad     2
9558    dog_clean_images\sad\9983838563_5683af739a_b.jpg    sad     2
9559    dog_clean_images\sad\9997069306_1c36187ca9_b.jpg    sad     2

[9560 rows x 3 columns]
```

Figure 12. Output of Loaded Data Frame

Next, the deep learning framework PyTorch is imported because it's a little more flexible python package. PyTorch is a publicly accessible machine learning context that speeds up the pathway from investigation prototyping to final positioning. It is built on the Python programming language and the Torch library. Torch is a publicly accessible machine learning library used for generating deep neural networks and is engraved in the Lua scripting language. It is one of the desired platforms for deep learning research (Anon. l). A random seed is created to implement randomness where the random numbers are

generated in a certain order and if the same code is executed twice also, the results are same.

```
# set randomness at certain order
torch.manual_seed(0)
my_device = torch.device('cpu')
```

Figure 13. Setting Randomness

Torchvision package is imported to read images, PyTorch that enables to work with images of computer vision and that is why it is called torch vision. Torchvision is an individual computer vision package of PyTorch which comprises of numerous significant and effective datasets as well as algorithms and conversion functions that are repeatedly applied around computer vision (Chakraborty 2022). Torchvision read_image and transforms libraries are used in this research work. The read_image function reads JPEG and PNG images and returns a 3D RGB or Grayscale Tensor. The three dimensions of the Grayscale tensor correspond to [C, H, W] where C is the number of channels, W and H are the width and height of the image. For RGB, the number of channels is 3 and it corresponds to [3, H, W] (Anon. g). The transforms' function is remarkably familiar in deep learning to operate image data augmentation. These are universal image transformations applied in various purpose of computer vision and image processing (Anon. h).

```
from torchvision.io import read_image
from torchvision import transforms as T
```

Figure 14. Torchvision Python Package

PyTorch consists of two important data primitives: torch.utils.data.DataLoader and torch.utils.data.Dataset which allows to use pre-loaded datasets as well as your own data. For that sake, Dog Dataset is created in the code. Dataset stores the examples and their subsequent labels, and DataLoader binds an alterable all over the Dataset to facilitate easy-going approach to the models (Anon. f). Deep learning techniques have the capability to work with millions and millions of images, since the images collected for this research is

not up to that level, the data augmentation method is used to create new images using our existing images. The data augmentation techniques are done by the installed package transforms. By applying this transforms function, it fundamentally modify the original image into a larger dataset. This can be useful to enhance the accuracy of the model. Random Horizontal Flip set a probability where the original image flips horizontally (i.e.,) the right side of the image becomes the left side, and the left side of the image becomes the right side which creates a new image. Next is Random Rotation which rotates the original image from 0 degrees to 90 degrees and form a new image. Then Auto Augment which is a PyTorch class that applies a set of predefined augmentations to automatically create a new image. By this data augmentation techniques, every image in the dataset is created into three additional images, which increases the size of the data.

The Convert Image D-Type which is a sub-category of Normalizer – Compose function that is applied in the code to read in each of those dog images and it will use the integers to represent each intensity value in the image. The Normalizer is used to standardize the image intensities and standard deviations of the dataset which just makes training a little bit faster. The standard values for the Normalizer function are taken by referring the ImageNet dataset (Anon. 2021). The original dataset and the mentioned augmentations multiply it by the length which means that our dataset will be four times as large as our original dataset. So, the 9560 original images are augmented into 38240 images.

The length of our original dataset and all of the augmentations will define a method called classes that is supposed to return the number of items that we are trying to predict so called three different labels/emotions that is angry, happy, or sad. This classes method will just return the number of unique labels, the most important method Define is used to call the length function, it will index the items in the dataset, it will iterate across the dataset and finally get each item individually with values like 0, 1, 2, 3, 4.

```
class Dog_Dataset(Dataset):
    def __init__(self, data_set):
        self.data_set = data_set
        self.augments = [T.RandomHorizontalFlip(1), T.RandomRotation(90), T.AutoAugment()]
        self.normalize = T.Compose([
            T.ConvertImageDtype(torch.float),
            T.Normalize((0.485, 0.456, 0.406), (0.229, 0.224, 0.225))
```

Figure 15. Syntax used for Data Augmentations

MODEL IMPLEMENTATION

The crucial step of all machine learning projects is splitting training and test data size. The train-test split technique is a way of evaluating the performance of machine learning models. Training set is used to train the model and testing set is used to check how accurate the model is performed (Prabhakaran 2022).

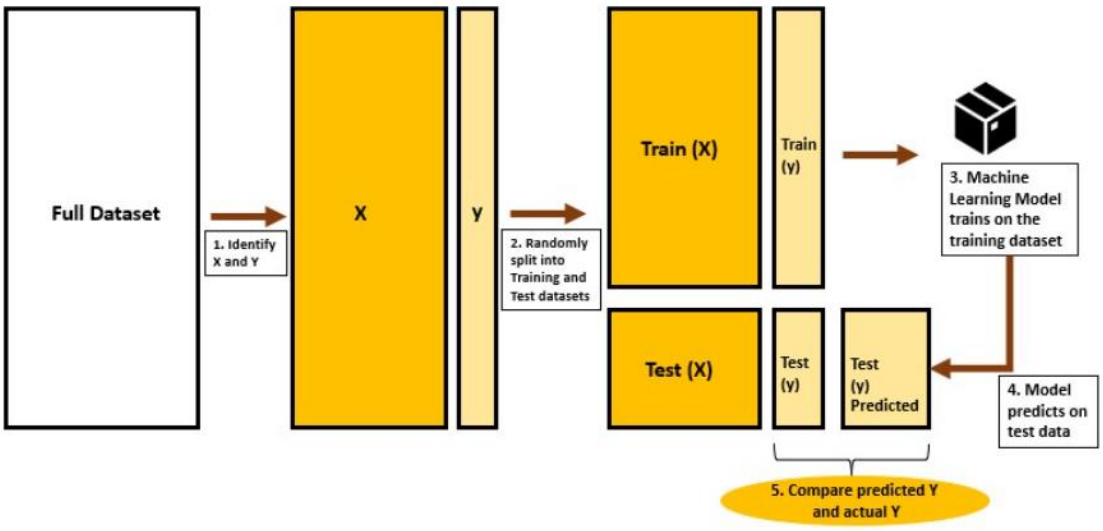


Figure 16. Train-Test Split Sample Methodology (Prabhakaran 2022)

The training size used for this research is 60% whereas the testing size is 40%. The reason behind using 60:40 ratio is the images in each emotion folder does not seems to in equal number. If the training size is increased, the random split function The function called Random Split is used to split the data randomly from train and test set. This function is passed in the generator and the reason behind is the split should be consistent for every execution. Again, random seed is used here to run this random split in the same order and the same values will be assigned to the train and the test set.

```
train_data, test_data = torch.utils.data.random_split(data, [train_size, test_size], generator=torch.Generator().manual_seed(1))

BATCH_SIZE = 16
EPOCHS = 100

train = DataLoader(train_data, batch_size=BATCH_SIZE, shuffle=True)
test = DataLoader(test_data, batch_size=BATCH_SIZE, shuffle=True)
```

Figure 17. Python Syntax for Train-Test Split and Random Split Function

Here the data loader is used to help load the data to train the deep learning model, so that it will create batches and able to be call the training methods. The batch size used

in this project is sixteen, this is the number of images that is used to train the DL model at one time so that less memory is consumed by the CPU. The number of images in the dataset seems to be very less compared to the capability of a DL model. Also, the higher batch size can end up with unstable results. The epochs used for this model is one hundred, an epoch is a complete number of repetitions of the whole train set in one sequence for model training. Shuffle function is enabled True, so that shuffling is performed in the order of the data.

```
BATCH_SIZE = 16
EPOCHS = 100

train = DataLoader(train_data, batch_size=BATCH_SIZE, shuffle=True)
test = DataLoader(test_data, batch_size=BATCH_SIZE, shuffle=True)
```

Figure 18. Batch Size and Epochs

The promising phase of the project is training the neural network which is defined within the class “Neural Network”. A convolutional neural network (CNN or ConvNet) is a complex structural design for deep learning that acquires straightaway commencing information. CNNs are valuable aimed at verdict patterns in images to spot objects, classes, and categories. They can likewise be an effect on categorizing audio, time series, and signal data. It consists of tens otherwise hundreds of strata that respectively learns in the direction of sense diverse structures of a picture. Screens exist useful to respective training picture for dissimilar purposes, and the result of respective convolved portrait is based as the response to the subsequent film. Screens can initialize as exact guileless sorts, such as illumination and ends, and add in complication to sorts that exclusively describe the objective.

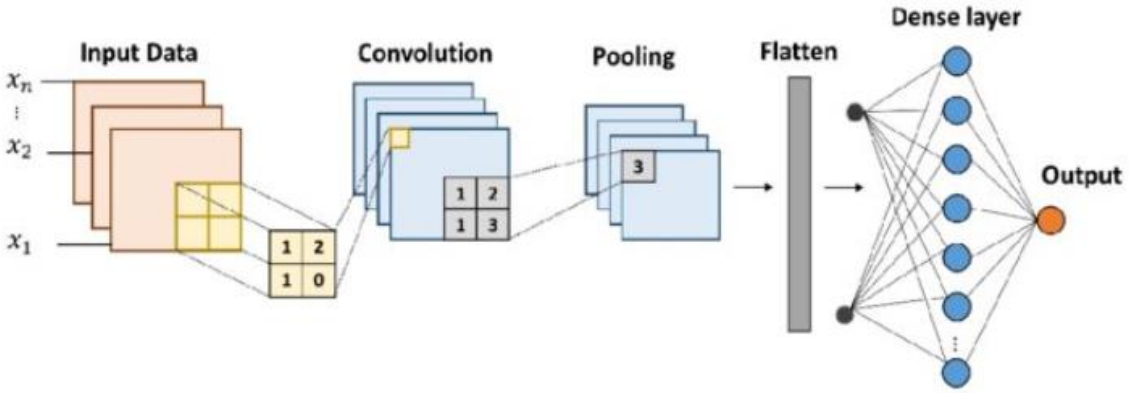


Figure 19. CNN Architecture (Ibrahim and Rabelo 2021)

```python
class NeuralNetwork(nn.Module):
    def __init__(self, classes):
        super(NeuralNetwork, self).__init__()

        self.bn = nn.BatchNorm2d(16)

        self.cnn = nn.Sequential(
            nn.Conv2d(3, 16, 4, stride=2),
            self.bn,
            nn.ReLU(True),
            nn.MaxPool2d(2, 2),
            nn.Conv2d(16, 16, 2),
            self.bn,
            nn.Conv2d(16, 16, 2),
            self.bn,
            nn.ReLU(True),
            nn.MaxPool2d(2, 2)
        )

        self.dense = nn.Sequential(
            nn.Linear(16 * 46 * 46, 16),
            nn.Linear(16, len(classes))
        )

    def forward(self, x):
        x = self.cnn(x)
        x = torch.flatten(x, 1)
        x = self.dense(x)
        return x
```

Figure 20. Python Syntax for Convolutional Neural Network

Loss function is applied to optimize the neural network and Cross Entropy Loss is a common way to get loss for classification task and this predict the correct label for each image. To prevent the algorithm predicting the wrong labels, we need an optimizer to optimize the weights of our neural network against this loss function. For that purpose, Stochastic Gradient Descent (SGD) is an optimizer which pass parameters and learning rate is given as 0.001 for our model. The lower learning rate executes the model slowly which helps to improve accuracy. Here comes the epochs execution and with it the loss functions values will printed. At the end, the pretrained model will be saved. The execution of this pretrained model almost took 12 hours to train and save the model. This training is performed in a high-definition system with specifications of i9 processor, 14 core, 32gb RAM and 2T storage.

```
loss_function = nn.CrossEntropyLoss()
optimizer = torch.optim.SGD(my_model.parameters(), lr=.001)

size = len(train.dataset)

for epoch in range(EPOCHS):
    for batch, (images, image_labels, img_paths) in enumerate(train):
        optimizer.zero_grad()

        images = images.to(my_device)
        prediction = my_model(images.float())

        image_labels = image_labels.to(my_device)
        loss = loss_function(prediction, image_labels)

        loss.backward()
        optimizer.step()

    loss, current = loss.item(), batch * len(images)
    print(f"loss: {loss:>7f}  [{epoch}]")

torch.save(my_model, 'pretrained_dog_model.pth')
```

Figure 21. Python Syntax for Loss Function and Saved Model

LOADING THE MODEL

After 12 hours of training time, the pretrained model is saved, it should be loaded to evaluate the testing set.

```
NeuralNetwork(
  (bn): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (cnn): Sequential(
    (0): Conv2d(3, 16, kernel_size=(4, 4), stride=(2, 2))
    (1): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (4): Conv2d(16, 16, kernel_size=(2, 2), stride=(1, 1))
    (5): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (6): Conv2d(16, 16, kernel_size=(2, 2), stride=(1, 1))
    (7): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (8): ReLU(inplace=True)
    (9): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (dense): Sequential(
    (0): Linear(in_features=33856, out_features=16, bias=True)
    (1): Linear(in_features=16, out_features=3, bias=True)
```

Figure 22. Structure of the Loaded Model

```python
my_model = torch.load('pretrained_dog_model.pth')

my_model.to(my_device)

all_predictions = list()
all_labels = list()
all_paths = list()

count = 0
with torch.no_grad():
    for batch, (images, labels, img_paths) in enumerate(test):
        count = count + 1
        print(count)
        images = images.to(my_device)
        outputs = my_model(images.float())

        _, preds = torch.max(outputs.data, 1)

        all_labels.append(labels)
        all_predictions.append(preds)
        all_paths.append(img_paths)
```

Figure 23. Python Syntax for Loading Predicted Values

```
preds = np.concatenate([p.cpu().numpy() for p in all_predictions])
labels = np.concatenate([p.cpu().numpy() for p in all_labels])
paths = np.concatenate([p for p in all_paths])

print(((preds == labels).sum()) / len(labels))

predictions = pd.DataFrame(dict(pred=preds, label=labels, path=paths))
predictions["correct"] = (predictions["pred"] == predictions["label"])

predictions["prediction"] = predictions["pred"].apply(lambda x: FOLDERS[int(x)])

predictions["actual"] = predictions["label"].apply(lambda x: FOLDERS[int(x)])

print(predictions.groupby("prediction").apply(lambda x: x["correct"].sum() / x.shape[0]))

disp = predictions.iloc[:50000, :].copy()

print(disp)


def image_formatter(path):
    return f'<img src="{path}">'
```

Figure 24. Python Syntax for Displaying Actual and Predicted Values

GRAPHICAL USER INTERFACE (GUI)

The output for this research project is shown in the form of Graphical Use Interface (GUI). This GUI is developed based on the Python library Flask and Jinja. Jinja is a full-featured template engine for Python and is the default template engine used by Flask. The Jinja template engine can generate any text-based format, e.g., HTML, XML, CSV, etc (Kraczkowsky).

```
import cv2
from flask import Flask, flash, request, redirect, url_for, render_template,  Response
import os
from werkzeug.utils import secure_filename
import requests
```

Figure 25. Python Packages for Developing GUI

This webpage is very user-friendly and consists of two options to detect the dog's emotions (angry, happy, or sad) are:

1) **Image Upload**
2) **Video Feed**

As shown in Figure 24, the image upload option will be useful for researchers or even dog owners who likes to predict their dog's emotions using pictures. First, the user should

upload the image using **“Choose File”** and click **“Submit”**. Once the image is uploaded successfully, it will be displayed in the webpage. Then, click **“Detect”** button to find out the emotion of the dog. The video feed option can be used to upload recorded videos of dogs, open web camera or any external camera connected to the system. This will be helpful for user who want to analyze the dog’s live emotions. The emotions of the dogs will be displayed in the video feed.

Figure 26. GUI Home Page

(Image Courtesy: https://assets3.thrillist.com/v1/image/3010071/1024x768/scale;jpeg_quality=60;progressive.jpg)

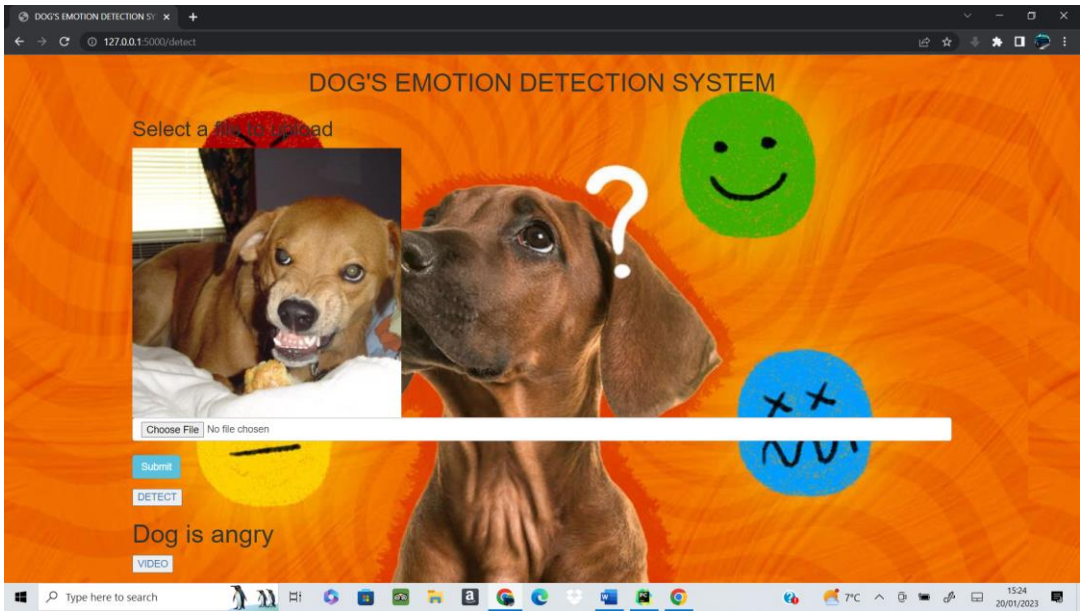


Figure 27. Predicting Angry Emotion

(Image Courtesy: https://assets3.thrillist.com/v1/image/3010071/1024x768/scale;jpeg_quality=60;progressive.jpg)

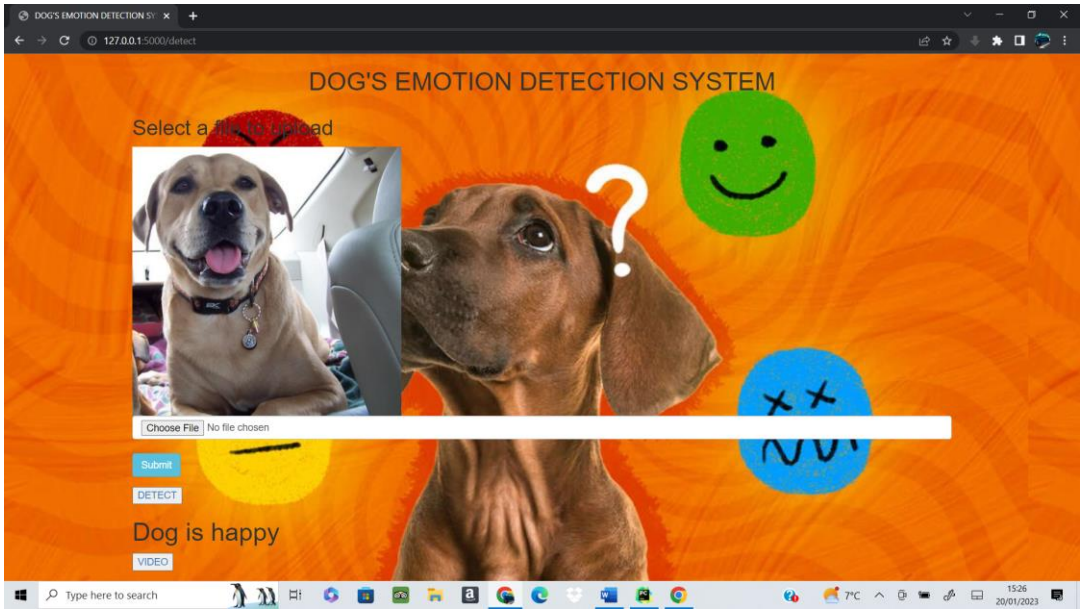


Figure 28. Predicting Happy Emotion

(Image Courtesy: https://assets3.thrillist.com/v1/image/3010071/1024x768/scale;jpeg_quality=60;progressive.jpg)

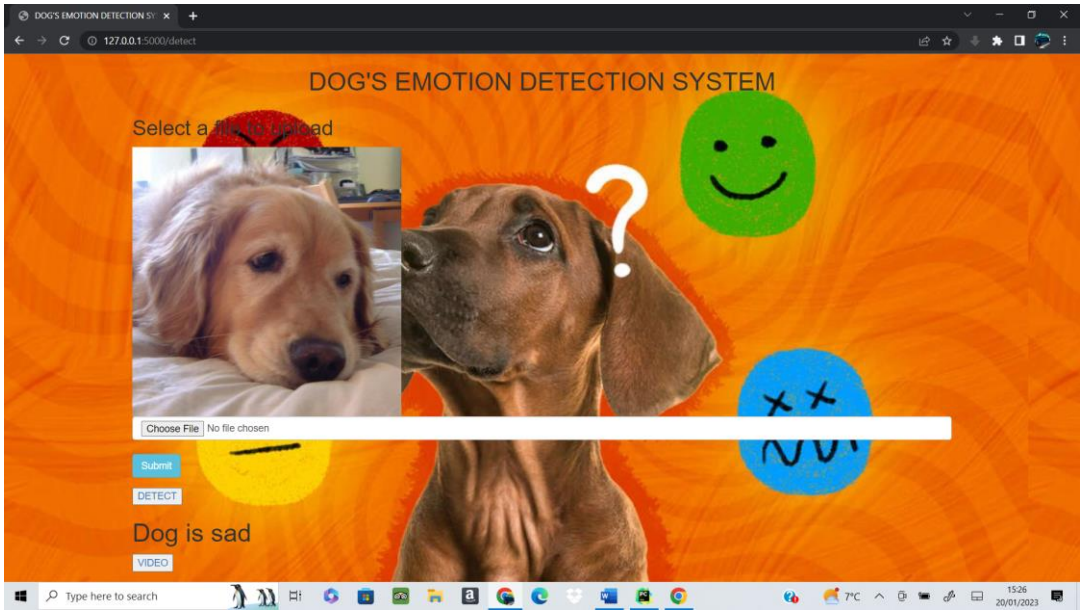


Figure 29. Predicting Sad Emotion

(Image Courtesy: https://assets3.thrillist.com/v1/image/3010071/1024x768/scale;jpeg_quality=60;progressive.jpg)

ARDUINO ALERT SYSTEM

The Arduino Alert System is used to detect aggressive dogs and gives alert in the form of alarming noise signals. How does this alert system work?

Requirements

1) **Arduino IDE:** The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as **Windows, Mac OS X, and Linux**. It supports the programming languages C and C++.

Here, IDE stands for **Integrated Development Environment**. The program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino' (Anon. d)

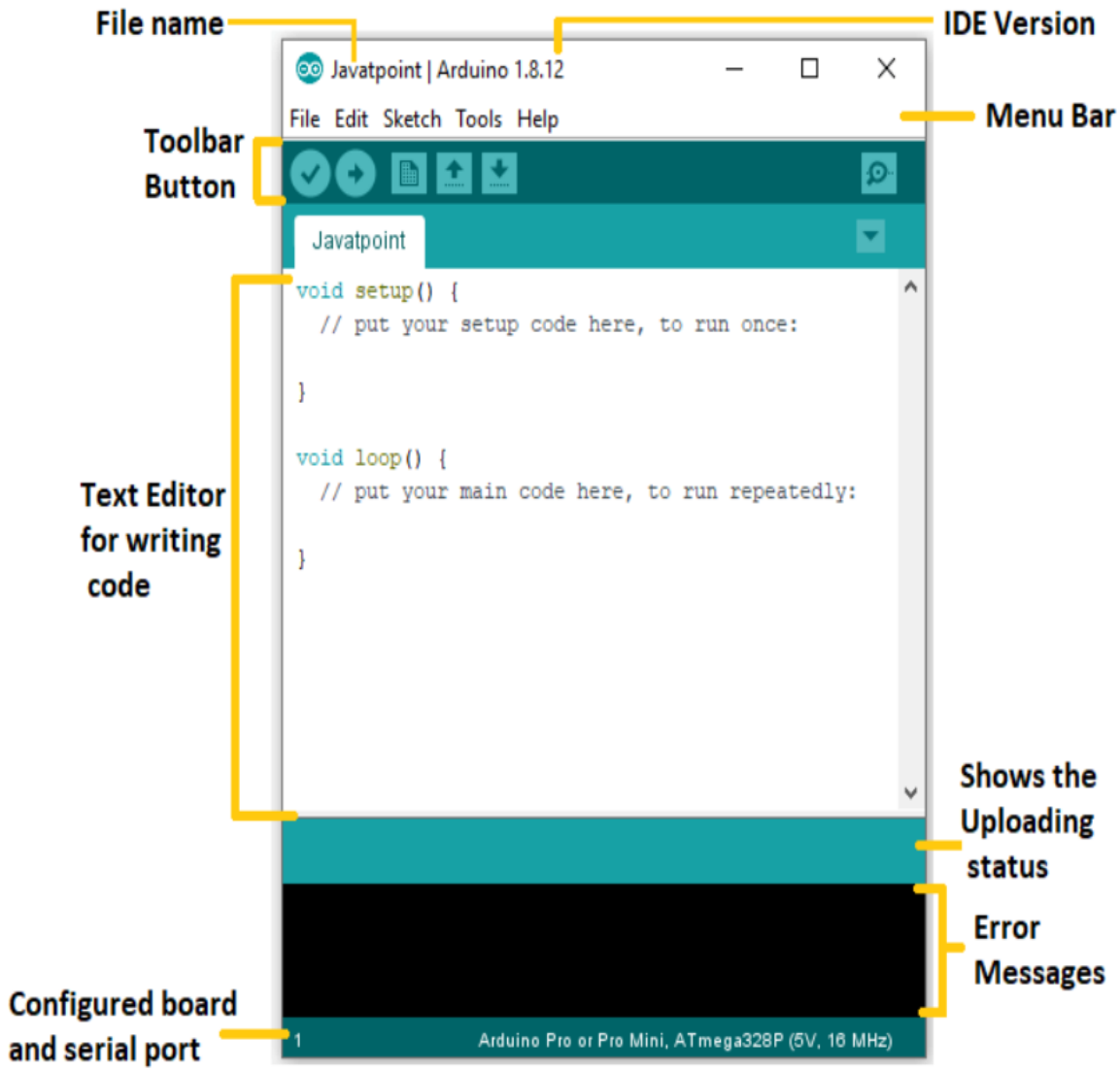


Figure 30. Arduino IDE Display (Anon. d)

2) **TECNOIOT ESP-12F WeMos D1 WiFi uno based ESP8266:** This is D1 R2 V2.1.0 is a mini-Wi-Fi card based on the ESP-8266EX. It consists of 11 digital input or output pins, all pins have interruption/Pulse Width Modulation (PWM)/I2C/ supports one cable (except D0); 1 analog input (maximum input of 3.2 V); micro–Universal Serial Bus (USB) connection; power connector, power input 9-24 V. It is consistent with Arduino. Its performance is processed by 3.3 V operating voltage; Clock speed 80MHz/160MHz; Flash 4M bytes.

3) **Piezoelectric Buzzer:** It is made up of high-quality ABS raw material, environmentally friendly, flame retardant and durable. An integrated electronic vocalizer comprising of multi-harmonic oscillator, piezoelectric buzzer chip, impedance matching unit, resonator, and housing. Its specifications are Rated Voltage: 12 V; Voltage range: 3-24 V; Rated current(max): 12 mA, Wire length: 100 mm/4 inch; Diameter: 23 mm/0.9 inch; 12-volt buzzer works with DC 3-24 V, Sound pressure level: 87 dB. It is very easy to operate when it is connected to the rated power supply, the buzzer beeps continuously, simple, and convenient to use. It is widely used as a buzzer for the circuit of many electronic products as an audible device.

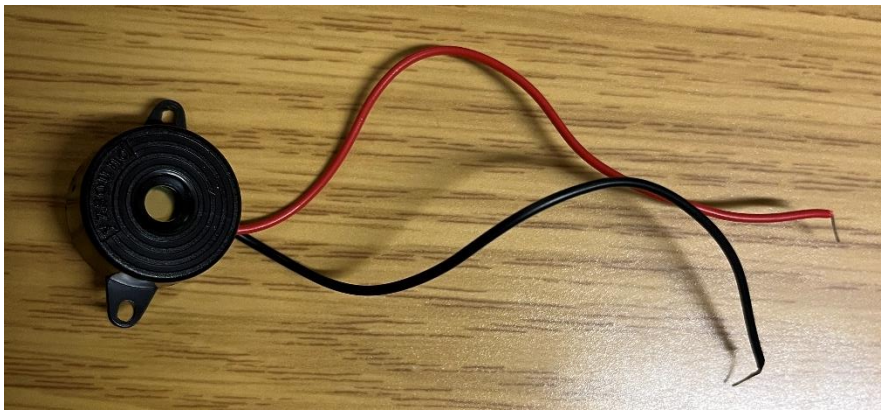


Figure 31. Piezoelectric Buzzer

4) **Breadboard, Power Cable, and Jumper Wires:** A dedicated Solderless Breadboard with 830 Tie-Points, 630 Tie-Point Terminal Strip and 200 Tie-Point Power Strip. Power cable is used to connect computer system with **ESP8266** device. The flexible jumper wires are basic component that connects two or more devices. They are produced in many colors so you can easily pick up the right one when lots are plugged. They provide an easy way to build your own circuitry on a breadboard as well as Arduino.

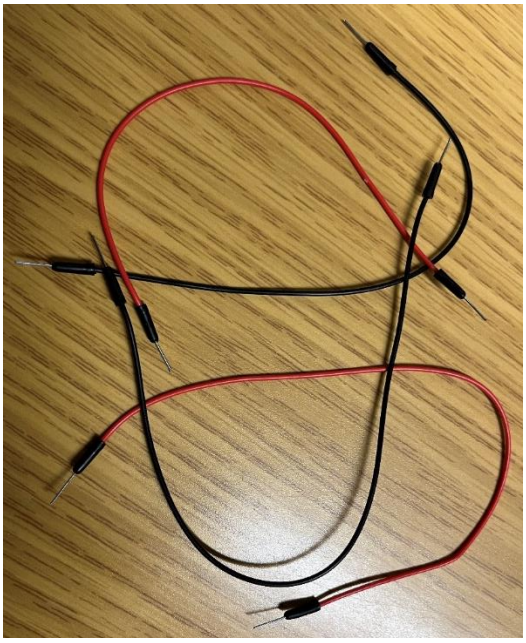


Figure 32. Jumper Wires

System Design

The ESP-8266EX device is connected to the computer system with the help of power cable. When it is connected, the in-built LED lights start to glow, which confirms that the device is connected to computer system. In the computer system, the software Arduino IDE should be installed to synchronize the python programming with the hardware device. Using the breadboard and jumper wires, the ESP-8266EX device is connected with the piezoelectric buzzer. The connected pins are to be mentioned in the Arduino IDE to take up the control of the hardware.

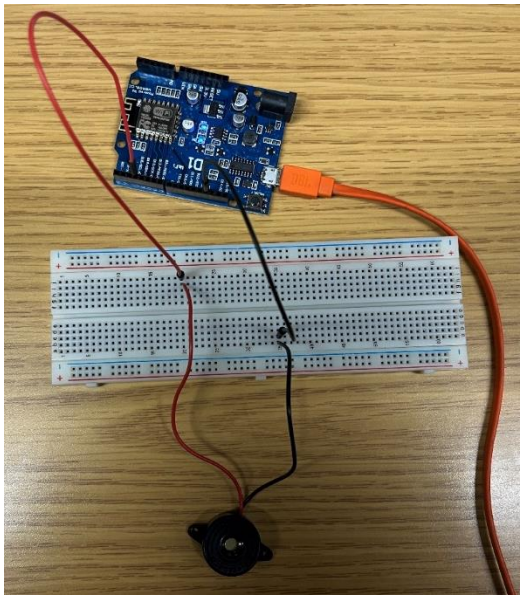


Figure 33. Design of Arduino Alert System

Python and Arduino IDE

The whole setup is interconnected by means of same Wi-Fi connection that is same server.

```
global flag
if predictables[predicted] == "angry" and flag == 1:
    requests.get("http://192.168.0.71/inline")
    print("Dogs")
    flag = 0
ret, buffer = cv2.imencode('.jpg', frame)
```

Figure 34. Python Syntax for Arduino IDE

```
server.on("/inline", []() {
  server.send(200, "text/plain", "this works as well");
  Serial.println("Dog is angry");
  // digitalWrite(4, 1);
  tone(4, 1000, 10000);
```

Figure 35. Syntax used in Arduino IDE

Output

Once the “Angry” emotion is detected in the GUI either by means of Image Upload option or Video Feed option, the Arduino alert system starts to produce alarming noise signals. The alert will last for 10 seconds. The frequency set for this device is 10 kHz which can be audible to human beings. There is also an option to set the frequency to 35 kHz – 45 kHz which can only be audible to dogs, so that it won’t cause disturbance to human beings.

Purpose of this system

There are many life-threatening issues are there in the society because of dog’s attack. In many countries, street and straw dogs are roaming around. If this kind of system is installed in those areas, it will be preventing from human beings from dog’s attack. I have researched and found that many researchers had already implemented projects like dog food/treat dispenser machines, dog pampering, pet-managed doors, and automatic ball launchers.

SECTION 4: RESULTS

The training and testing size of the dataset are used in a ratio of 60:40. The testing accuracy of each emotion like “Angry” is 79.1%, “Happy” is 67.1% and “Sad” is 75.2%. The experimental method of this research paper effectively shows 71.5%.

SECTION 5: DISCUSSIONS

The research is completely based on images dataset, so the major challenge I faced is training time. The training time for one training took around 12 hours. Also, a high-definition computer system with specifications of i9 processor, 14 core, 32gb RAM and 2T storage to train the model. Even if slight modifications are to be done in the model, I have to train from the initial stage and time of around 12 hours. I trained this model using ratios of 90:10, 80:10, 70:30, 60:50, 50:50 training and testing test. Compared to all the training, only 60:40 train & test set achieves the 70 – 80 % accuracies.

SECTION 6: CONCLUSION AND FUTURE WORKS

This research intends to examine whether a deep-learning model can identify dog’s emotions. sensations. The testing accuracy of each emotion like “Angry” is 79.1%, “Happy” is 67.1% and “Sad” is 75.2%. The experimental method of this research paper effectively shows 71.5%.

Forthcoming experiments aim to elaborate the data set in terms of demonstrative attitudes for each pose and emotional class and hardware can be expanded with a web camera and sensor to build a complete dog emotions recognition device. This full-fledged device can be used, and it will be great protection for human beings from dog’s attack.

SECTION 7: CHALLENGES AND LIMITATIONS

1) Reasons for eliminating dog’s body postures data: The explanations for not using dog’s body posture dataset are: (1) the facial expressions dataset is more useful as well as more precise compared to body posture data to predict the emotions of the dogs. (2) In the dataset which I have downloaded from Flickr API is a combination of both dog’s facial expressions and body postures dataset, in that, facial expressions images seem to be huge compared to body postures. Since, the body postures dataset is less, we eliminated those images and worked with only facial

expressions images. Added to, facial expressions require different DL algorithms and body postures require different DL algorithms, to avoid confusion and complication, we are sticking to one relevant DL model and relevant framework based on the facial expressions' dataset for this project.

2) Dataset Availability: There are only limited number of dog's datasets available in the open-source websites. As we are working with an image dataset, the model requires a maximum number of images to predict accurately. More than 10K images are required for each emotion category. However, hardly 2K – 5K images are collected by means of Flickr API and open-source websites. Because of this, we are not able to define the exact number of images that are required for each feature which will be major reason for not achieving an acceptable accuracy rate (70% - 90%). Since the images dataset are numerous and downloaded directly using Flickr API which took almost 3/4th day to complete.
3) Quality of the dataset: Only good-quality images can be used for model training. Even if we have more than 2K images with excellent quality for each emotion, it won't be able to predict or give a good accuracy rate unless the images are of high quality. The dataset we are using is a mixture of high and medium quality. Since we are detecting dog's emotions from its facial expressions, in many images, face of dogs is not in front view. These kinds of images have less possibility to predict the emotion of the dogs. It is crucial that the dataset is prepared in such a way that our model can fully comprehend the data. The model will then be able to successfully use that dataset for learning.
4) Training the model: Since we are using images as dataset and the device seems to be cpu based system, it almost took 2 days for training the dataset. If we make any minor changes to the code also, we again need to train the data freshly for 2 days. Because of this condition, the system runs continuously.
5) Some dogs like pugs, bullmastiff, bulldog, Chinese shar-pei, bloodhound, basset hound, a Neapolitan mastiff, and the Pekingese are naturally having sad-looking face and droopy-like appearance, it was difficult to find exact emotions in these dogs. One important thing is to be noted is these kinds of dogs can be very ferocious and dangerous as well (Russel 2021).

BIBLIOGRAPHIES

12 signs of an extremely happy pooch | by Adaptil | Happy Hound | Medium. a.

30 Reasons Why Dogs Are Better Than Humans. b.

All Mathematical Functions Defined under Math Module in Python 3. c.

Arduino IDE - JavaTpoint d. Available from: https://www.javatpoint.com/arduino-ide#:~:text=The%20Arduino%20IDE%20is%20an%20open-source%20software%2C%20which

A Beginner's Guide to the Python time Module – Real Python. e.

BONEH-SHITRIT, T. *et al.,* 2022. Explainable automated recognition of emotional states from canine facial expressions: the case of positive anticipation and frustration. *Scientific Reports,* 12

CHAKRABORTY, A., 2022. *Intro to Computer Vision with PyTorch's TorchVision* Available from: https://medium.com/jovianml/intro-to-computer-vision-with-pytorchs-torchvision-46c4ff43ce34

Datasets & DataLoaders — PyTorch Tutorials 1.11.0+cu102 documentation f. Available from: https://pytorch.org/tutorials/beginner/basics/data_tutorial.html

FERRES, K., T. SCHLOESSER and P.A. GLOOR, 2022. Predicting Dog Emotions Based on Posture Analysis Using DeepLabCut. *Future internet,* 14(4), 97

FRANZONI, V. *et al.,* Oct 14, 2019. A Preliminary Work on Dog Emotion Recognition. ACM, pp.91-96

*How to normalize images in PyTorch ?*2021. Available from: https://www.geeksforgeeks.org/how-to-normalize-images-in-pytorch/

*How to read a JPEG or PNG image in PyTorch?*g. Available from: https://www.tutorialspoint.com/how-to-read-a-jpeg-or-png-image-in-pytorch

HUGHES, C., 2022. *Getting Started with PyTorch Image Models (timm): a practitioner's guide* Available from: https://towardsdatascience.com/getting-started-with-pytorch-image-models-timm-a-practitioners-guide-4e77b4bf9055

IBRAHIM, B. and L. RABELO, 2021. A Deep Learning Approach for Peak Load Forecasting: A Case Study on Panama. *Energies,* 14

Important TorchVision Random Transforms used in Image Augmentation - Python PyTorch h. Available from: https://www.binarystudy.com/2022/07/important-torchvision-random-

transforms-image-augmentation.html#:~:text=Wondered%20what%20Torch%20Vision%20transforms%20actually%20do%3F%20These

KRACZKOWSKY, C., *Building modern user interfaces with Flask | by Colin Kraczkowsky | Medium* [viewed Jan 20, 2023]. Available from: https://ckraczkowsky.medium.com/building-modern-user-interfaces-with-flask-23016d453792

MENDL, M., O.H.P. BURMAN and E.S. PAUL, 2010. An integrative and functional framework for the study of animal emotion and mood | Proceedings of the Royal Society B: Biological Sciences. *Proceedings of the Royal Society B: Biological Sciences,* 277(1696), 2895-2904

MOTA-ROJAS, D. *et al.,* 2021. Current Advances in Assessment of Dog's Emotions, Facial Expressions, and Their Use for Clinical Recognition of Pain. *Animals (Basel),* 11(11), 3334

Pillow: Python Imaging Library (Fork) i. [viewed Jan 14, 2023]. Available from: https://python-pillow.org

PRABHAKARAN, S., 2022. *Train Test Split - How to split data into train and test for validating machine learning models?* Available from: https://www.machinelearningplus.com/machine-learning/train-test-split/

Project Jupyter 2019a. Available from: https://jupyter.org/

Python Imaging Library (PIL) Tutorial | Aman Kharwal j. [viewed Jan 17, 2023]. Available from: https://thecleverprogrammer.com/2021/08/21/python-imaging-library-pil-tutorial/

Python Requests Module. k.

PyTorch l.

R. O. SINNOTT *et al.,* 2021. Run or Pat: Using Deep Learning to Classify the Species Type and Emotion of Pets. - *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE).* pp.1-6

RUSSEL, 2021. *10 Dog Breeds With Sad Looking Faces (Pictures & Facts)* Available from: https://zooawesome.com/home-pets/10-dog-breeds-with-sad-looking-faces/#:~:text=Most%20dog%20breeds%20with%20a%20sad-looking%20face%20usually

Sad Dog - What Makes Dogs Sad, And How to Cheer Them Back Up. m.

Search - PDSA. n.

SHEHMIR JAVAID, Quick Guide to Datasets for Machine Learning in 2023.

shutil — High-level file operations — Python 3.11.1 documentation o. [viewed Jan 17, 2023]. Available from: https://docs.python.org/3/library/shutil.html

SINGH, B. *et al.,* 2020. Animal Emotion Detection and Application. *Midas,*

STUVEL, S.A., *flickrapi: The Python interface to the Flickr API.* [viewed Jan 13, 2023]. Available from: https://stuvel.eu/flickrapi

TSAI, M. and J. HUANG, 2021. *Sentiment Analysis of Pets Using Deep Learning Technologies in Artificial Intelligence of Things System.*

WALDEN, L., 2019. *8 Key Dog Emotions and How To Identify Them* [viewed Jan 14, 2023]. Available from: https://www.countryliving.com/uk/wildlife/pets/a28407333/dog-body-language/

You Can Identify Your Dog's Emotion Through Its Facial Expressions | Manuka Vet Hospital 2019b. [viewed Jan 14, 2023]. Available from: https://manukavet.com.au/you-can-identify-your-dogs-emotion-through-its-facial-expressions/

AGARWAL, V., 2020. *Complete Architectural Details of All EfficientNet Models* [online] Available from: https://towardsdatascience.com/complete-architectural-details-of-all-efficientnet-models-5fd5b736142

ANON., n.d. Convolutional Neural Networks (CNN) with Deep Learning [online] Available from: https://www.happiestminds.com/insights/convolutional-neural-networks-cnns/#:~:text=Within%20Deep%20Learning%2C%20a%20Convolutional

ANON., n.d. Emotion Detection Using CNN | Emotion Detection Deep Learning Project |Machine Learning | Data Magic [online] Available from: https://www.youtube.com/watch?v=UHdrxHPRBng

ANON., n.d. Dog Depression: Signs, Causes, and Treatment [online] Available from: https://www.petmd.com/dog/behavior/can-dogs-get-depressed

ANON., n.d. Papers with Code - EfficientNet Explained [online] Available from: https://paperswithcode.com/method/efficientnet

ANON., n.d. What Is Kaggle? [online] Available from: https://www.datacamp.com/blog/what-is-kaggle

ANON., n.d. PyTorch [online] Available from: https://www.techtarget.com/searchenterpriseai/definition/PyTorch

ANON., n.d. Assistance Dogs a Guide for All Businesses [online] Available from: https://www.equalityhumanrights.com/sites/default/files/assistance-dogs-a-guide-for-all-businesses.pdf

ANON., n.d. How Many Pets Are There in the UK? [online] Available from: https://www.pdsa.org.uk/what-we-do/pdsa-animal-wellbeing-report/uk-pet-populations-of-dogs-cats-and-rabbits#:~:text=Our%202022%20findings%20showed%20that%3A%201%2052%25%20of

ANON., 2022. 8 Dogs Faces Set, Year of the Dog Stock Vector - Illustration of Lettering, Spring: 107812293 [online] [viewed 24 Nov 2022]. Available from: https://www.dreamstime.com/dogs-faces-set-year-dog-traditional-celebration-china-spring-festival-animal-image107812293

ANON., 2021a. How to Normalize Images in PyTorch? [online] Available from: https://www.geeksforgeeks.org/how-to-normalize-images-in-pytorch/

ANON., n.d. Datasets & DataLoaders — PyTorch Tutorials 1.11.0+Cu102 Documentation [online] Available from: https://pytorch.org/tutorials/beginner/basics/data_tutorial.html

ANON., 2019a. You Can Identify Your Dog's Emotion through Its Facial Expressions | Manuka Vet Hospital [online] [viewed 14 Jan 2023]. Available from: https://manukavet.com.au/you-can-identify-your-dogs-emotion-through-its-facial-expressions/

ANON., n.d. Quick Guide to Datasets for Machine Learning [online] Available from: https://research.aimultiple.com/datasets-for-ml/#:~:text=A%20machine%20learning%20dataset%20is

ANON., 2022a. Training an Emotion Detection System Using PyTorch [online] [viewed 24 Nov 2022]. Available from: https://neuraspike.com/blog/training-emotion-detection-system-pytorch/

ANON., 2015. 30 Reasons Why Dogs Are Better than Humans [online] [viewed 10 Jan 2023]. Available from: https://www.theodysseyonline.com/why-dogs-are-better-than-humans

ANON., 2022. Dog Emotions Prediction TF EfficientNetB4+ResMLP [online] [viewed 24 Nov 2022]. Available from: https://www.kaggle.com/code/jiaowoguanren/dog-emotions-prediction-tf-efficientnetb4-resmlp/data

ANON., 2022b. Real-Time Emotion Detection System with PyTorch and OpenCV [online] [viewed 24 Nov 2022]. Available from: https://neuraspike.com/blog/realtime-emotion-detection-system-pytorch-opencv/

ANON., 2021b. What Is TensorFlow? How It Works [with Examples] [online] [viewed 24 Nov 2022]. Available from: https://www.upgrad.com/blog/what-is-tensorflow/

ANON., 2022. Dog Emotions F1 Score=45% [online] [viewed 24 Nov 2022]. Available from: https://www.kaggle.com/code/gpiosenka/dog-emotions-f1-score-45/notebook

ANON., 2022. Dog Emotion Classify Model by MobileNetV2 & VGG16 [online] [viewed 24 Nov 2022]. Available from: https://www.kaggle.com/code/sasakitetsuya/dog-emotion-classify-model-by-mobilenetv2-vgg16

ANON., 2020. Streamlit - Introduction and Setup [online] [viewed 24 Nov 2022]. Available from: https://www.geeksforgeeks.org/streamlit-introduction-and-setup/

ANON., 2019b. Why Are Dogs Important to Humans? | Ask Dr. Universe | Washington State University [online] Available from: https://askdruniverse.wsu.edu/2019/08/08/dogs-important-humans/

ANON., 2022. Detect Dog Emotions with Deep Learning [Full Walkthrough W/Code] [online] [viewed 24 Nov 2022]. Available from: https://www.youtube.com/watch?v=HMAIoVQel1c

ANON., 2022. Anaconda Navigator — Anaconda Documentation [online] [viewed 24 Nov 2022]. Available from: https://docs.anaconda.com/navigator/index.html#:~:text=Anaconda%20Navigator%20is%20a%20desktop

ANON., 2022. TensorFlow Intro [online] [viewed 24 Nov 2022]. Available from: https://www.w3schools.com/ai/ai_tensorflow_intro.asp

ANON., 2022. Arduino Uno REV3 [A000066]: Amazon.co.uk: Computers & Accessories [online] [viewed 25 Nov 2022]. Available from: https://www.amazon.co.uk/Arduino-A000066-ARDUINO-UNO-REV3/dp/B008GRTSV6/ref=asc_df_B008GRTSV6/?tag=googshopuk-21&linkCode=df0&hvadid=310855475891&hvpos=&hvnetw=g&hvrand=6613287324141922442&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9045800&hvtargid=pla-457497319401&th=1

ANON., 2023. How to Read a JPEG or PNG Image in PyTorch? [online] [viewed 19 Jan 2023]. Available from: https://www.tutorialspoint.com/how-to-read-a-jpeg-or-png-image-in-pytorch

ANON., 2023. Arduino IDE - JavaTpoint [online] [viewed 20 Jan 2023]. Available from: https://www.javatpoint.com/arduino-ide#:~:text=The%20Arduino%20IDE%20is%20an%20open-source%20software%2C%20which

BONEH-SHITRIT, T. et al., 2022. Deep Learning Models for Automated Classification of Dog Emotional States from Facial Expressions. arXiv:2206.05619 [cs] [online] [viewed 24 Nov 2022]. Available from: https://arxiv.org/abs/2206.05619

CANGAROO6, 2022. Dog-Emotions [online] [viewed 24 Nov 2022]. Available from: https://github.com/cangaroo6/Dog-Emotions

CHAKRABORTY, A., 2020. Intro to Computer Vision with PyTorch's TorchVision [online] [viewed 19 Jan 2023]. Available from: https://medium.com/jovianml/intro-to-computer-vision-with-pytorchs-torchvision-46c4ff43ce34

EDANZ-LEARNING-TEAM, 2022. Understanding a PRISMA Flow Diagram [online] Available from: https://learning.edanz.com/prisma-flow-diagram/

FERRES, K., T. SCHLOESSER and P.A. GLOOR, 2022. Predicting Dog Emotions Based on Posture Analysis Using DeepLabCut. Future Internet, 14(4), 97

FRANZONI, V. et al., 2019. A Preliminary Work on Dog Emotion Recognition. IEEE/WIC/ACM International Conference on Web Intelligence - Companion Volume

HARVARD HEALTH PUBLISHING, 2019. Having a Dog Can Help Your Heart — Literally - Harvard Health [online] Available from: https://www.health.harvard.edu/staying-healthy/having-a-dog-can-help-your-heart--literally

HUGHES, C., 2022. Getting Started with PyTorch Image Models (Timm): A Practitioner's Guide [online] Available from: https://towardsdatascience.com/getting-started-with-pytorch-image-models-timm-a-practitioners-guide-4e77b4bf9055

ILYENA, 2017. How to Build Dog Detection Systems with Arduino [online] [viewed 24 Nov 2022]. Available from: https://ilyena.com/2017/09/01/doguino/

JUPYTER, 2019. Project Jupyter [online] Available from: https://jupyter.org/

KAZARINOFF, P.D., 2018. Using Python to Control an Arduino [online] [viewed 24 Nov 2022]. Available from: https://pythonforundergradengineers.com/python-arduino-LED.html#:~:text=To%20communicate%20with%20the%20Arduino

LINK, G. et al., 2023. Important TorchVision Random Transforms Used in Image Augmentation - Python PyTorch [online] [viewed 19 Jan 2023]. Available from: https://www.binarystudy.com/2022/07/important-torchvision-random-transforms-image-augmentation.html#:~:text=Wondered%20what%20Torch%20Vision%20transforms%20actually%20do%3F%20These

MATPLOTLIB, 2012. Matplotlib: Python Plotting — Matplotlib 3.1.1 Documentation [online] Available from: https://matplotlib.org/

MCQUERREY, L., 2014. The Importance of Dogs in Our Lives [online] Available from: https://dogcare.dailypuppy.com/importance-dogs-lives-6622.html

MENDL, M., O.H.P. BURMAN and E.S. PAUL, 2010. An integrative and functional framework for the study of animal emotion and mood. Proceedings of the Royal Society B: Biological Sciences, 277(1696), 2895–2904

PANDAS, 2018. Python Data Analysis Library — Pandas: Python Data Analysis Library [online] Available from: https://pandas.pydata.org/

PRABHAKARAN, S., 2022. Train Test Split - How to Split Data into Train and Test for Validating Machine Learning Models? [online] [viewed 19 Jan 2023]. Available from: https://www.machinelearningplus.com/machine-learning/train-test-split/

RIAZ, M.N. et al., 2020. eXnet: An Efficient Approach for Emotion Recognition in the Wild. Sensors, 20(4), 1087

RUSSEL, 2021. 10 Dog Breeds with Sad Looking Faces (Pictures & Facts) [online] [viewed 14 Jan 2023]. Available from: https://zooawesome.com/home-pets/10-dog-breeds-with-sad-looking-faces/#:~:text=Most%20dog%20breeds%20with%20a%20sad-looking%20face%20usually

SAHA, S., 2018. A Comprehensive Guide to Convolutional Neural Networks — the ELI5 Way [online] Available from: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

SEABORN, 2012. Seaborn: Statistical Data Visualization — Seaborn 0.9.0 Documentation [online] Available from: https://seaborn.pydata.org/

SHARMA, T., 2020. Facial Expression Recognition Using PyTorch [online] [viewed 24 Nov 2022]. Available from: https://blog.jovian.ai/facial-expression-recognition-using-pytorch-b7326ab36157

SINGH, J.R., 2020. Emotion Detection Using PyTorch [online] [viewed 24 Nov 2022]. Available from: https://medium.com/swlh/emotion-detection-using-pytorch-4f6fbfd14b2e

SINGH, J.R., 2022. Emotion-Detection-PyTorch [online] [viewed 24 Nov 2022]. Available from: https://github.com/Conero007/Emotion-Detection-Pytorch

SINNOTT, R.O. et al., 2021. Run or Pat: Using Deep Learning to Classify the Species Type and Emotion of Pets [online] [viewed 24 Nov 2022]. Available from: https://ieeexplore.ieee.org/document/9718465

TEAM, J.D., 2022. Jupyterlab: JupyterLab Computational Environment [online] [viewed 24 Nov 2022]. Available from: https://pypi.org/project/jupyterlab/

TENSORFLOW, 2019. TensorFlow [online] Available from: https://www.tensorflow.org/

TSAI, M.-F. and J.-Y. HUANG, 2021. Sentiment Analysis of Pets Using Deep Learning Technologies in Artificial Intelligence of Things System

WALDEN, L., 2019. 8 Dog Body Language Signs and the Emotions They Portray [online] Available from: https://www.countryliving.com/uk/wildlife/pets/a28407333/dog-body-language/

WEBSITE, W., 2022. Electronic Ultrasonic Emitter - Basic Version [online] [viewed 24 Nov 2022]. Available from: https://www.instructables.com/Electronic-Ultrasonic-Emitter-Basic-Version/

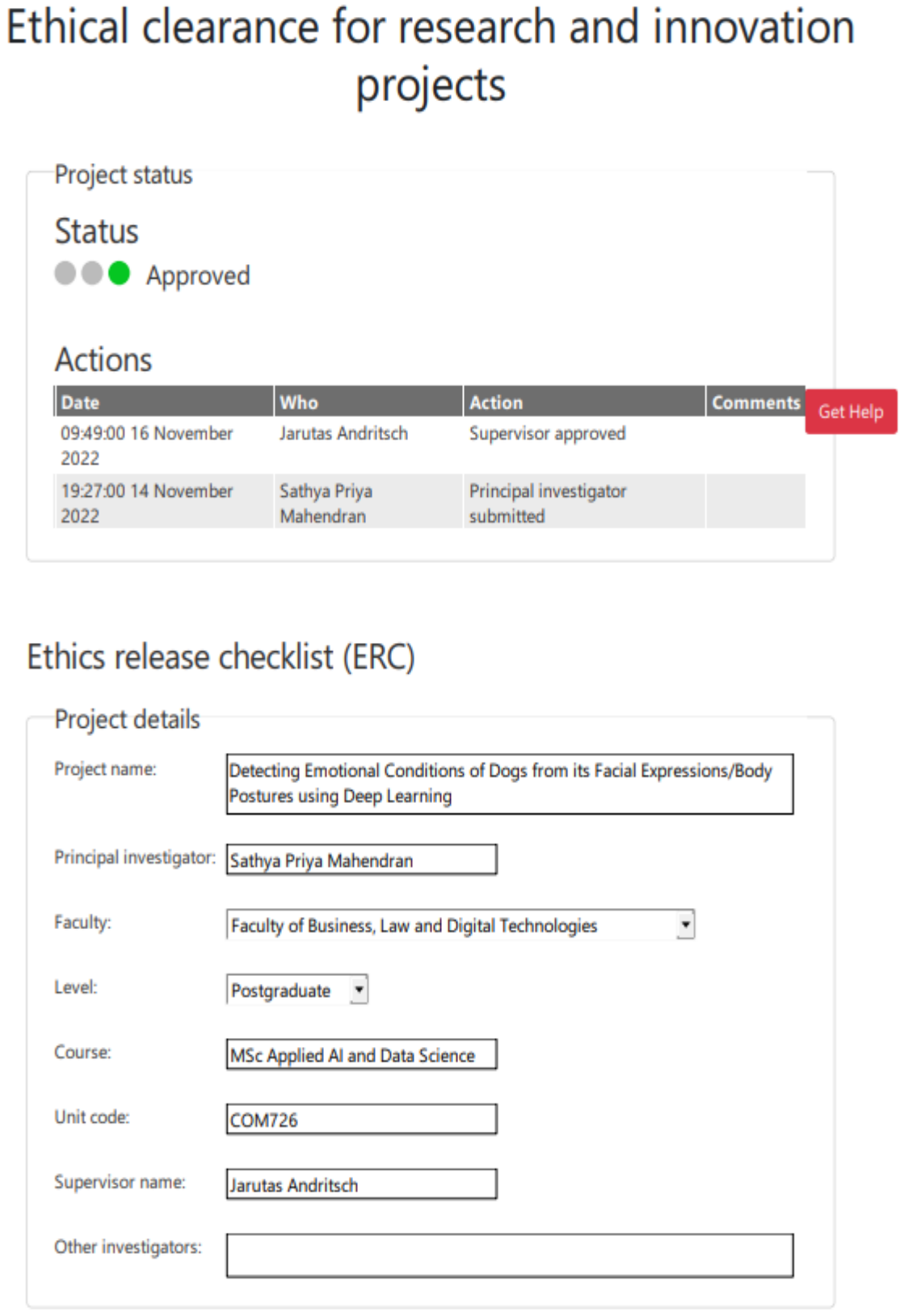
WIKIPEDIA CONTRIBUTORS, 2019a. Research [online] Available from: https://en.wikipedia.org/wiki/Research

WIKIPEDIA CONTRIBUTORS, 2019b. Gantt Chart [online] Available from: https://en.wikipedia.org/wiki/Gantt_chart

YUTAKA_KUN, 2019. Python Beginner Tried to Make Dog Cat Recognition AI [online] [viewed 24 Nov 2022]. Available from: https://medium.com/lsc-psd/python-beginner-tried-to-make-dog-cat-recognition-ai-bbefc80bddcb

APPENDICES

Appendix 1: Ethics Application Approval



Ethical clearance for research and innovation projects

Project status

Status

Approved

Actions

Date	Who	Action	Comments
09:49:00 16 November 2022	Jarutas Andritsch	Supervisor approved	
19:27:00 14 November 2022	Sathya Priya Mahendran	Principal investigator submitted	

Get Help

Ethics release checklist (ERC)

Project details

Project name:	Detecting Emotional Conditions of Dogs from its Facial Expressions/Body Postures using Deep Learning
Principal investigator:	Sathya Priya Mahendran
Faculty:	Faculty of Business, Law and Digital Technologies
Level:	Postgraduate
Course:	MSc Applied AI and Data Science
Unit code:	COM726
Supervisor name:	Jarutas Andritsch
Other investigators:	

Figure 36. Ethics Approval Form (Page - 1)

Checklist

Question	Yes	No
Q1. Will the project involve human participants other than the investigator(s)?	○	◉
Q1a. Will the project involve **vulnerable participants** such as children, young people, disabled people, the elderly, people with declared mental health issues, prisoners, people in health or social care settings, addicts, or those with learning difficulties or cognitive impairment either contacted directly or via a **gatekeeper** (for example a professional who runs an organisation through which participants are accessed; a service provider; a care-giver; a relative or a guardian)?	○	◉
Q1b. Will the project involve the use of **control groups** or the **use of deception**?	○	◉
Q1c. Will the project involve any **risk to the participants' health** (e.g. intrusive intervention such as the administration of drugs or other substances, or vigorous physical exercise), or involve psychological stress, anxiety, humiliation, physical pain or discomfort to the investigator(s) and/or the participants?	○	◉
Q1d. Will the project involve **financial inducement** offered to participants other than reasonable expenses and compensation for time?	○	◉
Q1e. Will the project be carried out by individuals unconnected with the University but who wish to use staff and/or students of the University as participants?	○	◉
Q2. Will the project involve sensitive materials or topics that might be considered offensive, distressing, politically or socially sensitive, deeply personal or in breach of the law (for example criminal activities, sexual behaviour, ethnic status, personal appearance, experience of violence, addiction, religion, or financial circumstances)?	○	◉
Q3. Will the project have detrimental impact on the environment, habitat or species?	○	◉
Q4. Will the project involve living animal subjects?	○	◉
Q5. Will the project involve the development for export of 'controlled' goods regulated by the Export Control Organisation (ECO)? (This specifically means military goods, so called dual-use goods (which are civilian goods but with a potential military use or application), products used for torture and repression, radioactive sources.) Further information from the Export Control Organisation	○	◉
Q6. Does your research involve: the storage of records on a computer, electronic transmissions, or visits to websites, which are associated with terrorist or extreme groups or other security sensitive material? Further information from the Information Commissioners Office	○	◉

Figure 37. Ethics Approval Form (Page - 2)

Declarations

I/we, the investigator(s), confirm that:

☑ The information contained in this checklist is correct.

☑ I/we have assessed the ethical considerations in relation to the project in line with the University Ethics Policy.

☑ I/we understand that the ethical considerations of the project will need to be re-assessed if there are any changes to it.

☑ I/we will endeavor to preserve the reputation of the University and protect the health and safety of all those involved when conducting this research/enterprise project.

☑ If personal data is to be collected as part of my project, I confirm that my project and I, as Principal Investigator, will adhere to the General Data Protection Regulation (GDPR) and the Data Protection Act 2018. I also confirm that I will seek advice on the DPA, as necessary, by referring to the Information Commissioner's Office further guidance on DPA and/or by contacting information.rights@solent.ac.uk. By Personal data, I understand any data that I will collect as part of my project that can identify an individual, whether in personal or family life, business or profession.

☑ I/we have read the prevent agenda.

Figure 38. Ethics Approval Form (Page - 3)