Gesture-Based Control of MIDOG Quadruped Robot: A Neural Network Approach

Group 1, Construction Robotics, WS 2023/24

Harsh Prabhakar Sharma (561451),
Ayush Vasantbhai Patel (564382),
Ishan Sandeep Mohite (561464)

***Abstract*—This paper presents an approach for performing actions using the Petoi Bittle Robot dog based on the hand gesture detected by the camera on the robot.**

This approach for controlling the robot is beneficial for medical assistance where the robot can perform challenging tasks based on the gestures. Also, it proves beneficial in smart home integration for the control of various home automation tasks.

The hand gestures are detected using a pre-trained neural network and open-source MediaPipe framework for hand landmark detection. The actions are executed using the action commands from the Robot Operating System (ROS) framework.

The actions corresponding to the different gestures are defined beforehand. The image feed received from the robot camera is passed through this neural network and it selects the gesture with the highest confidence based on the labeled data provided.

***Index Terms*—Hand gestures, Computer Vision, efficient Neural Network, robot movement**

I. Introduction

The field of robotics has found its way into many different sectors in recent years. Different robots are used for a variety of automation and assistance tasks. Particularly, mobile robotics is an emerging field due to their mobility, flexibility, and wide range of tasks they can perform. Quadruped robots that mimic dog-like movement provide a wide range of motion that can be used to perform various movements on different terrains. This can be very beneficial in the medical assistance of patients and also in automation tasks in smart home systems. This paper presents a way to perform such actions based on the hand gestures detected by the robot camera [6].

The method defined in this paper provides a way to detect gestures using a neural network. The network first detects the hand landmarks using the MediaPipe framework and a convolutional neural network. Then a feed-forward neural network is implemented and it classifies the gesture from the set of gestures based on the training from the labeled data.

However, when performing such tasks, there always remains a question of computational efficiency and limitations. As the memory available on the robot to perform real-time recognition and command execution is limited, the neural network should be efficient whilst performing the actions with acceptable delay.

This paper dives deeper into the background and previous work, methodology used in the framework, results, and encountered challenges in further sections.

II. Background

A. MediaPipe Hand Detection

The MediaPipe Hand Landmark detection is an open-source framework by Google to detect the landmarks of the hands in an image [1] & [5]. It can be used to identify the hands' important points and apply visual effects to them. It uses a machine learning (ML) model to operate on image data, which can either be a static image or a continuous stream as in this project. It outputs the handedness of several detected hands as well as hand landmarks in both image and world coordinates. For this project, only single hand gestures are used to control the robot. Hence, the maximum number of detected hands at a time is set to one.

B. mIDOG setup

The mIDOG setup used in the hardware is a Petoi Bittle Robot dog equipped with a Raspberry Pi 3A+ that acts as a micro-controller, nine Servo Motors to perform various motions, and sensors like Raspberry Pi Camera and Internal Measurement Unit (IMU). When the IMU detects a drastic change in motor angles, the controller stops the motors, and the robot is re-calibrated. If the robot is in the middle of executing a task, it is stopped. It consists of a Li-ion battery which has a battery life of around 70 minutes.

The software environments Gazebo and RViz in conjunction with the ROS framework are used to simulate the robot's behavior before trying it in the real world. Due to the limited computation capability and memory of Raspberry Pi, the robot is connected to a remote server using SSH where the computations are carried out. This allows for more computational memory and complex tasks like image segmentation and executing neural networks can be performed on the robot.

III. Method and Approach

This section includes the method and approach the team followed to achieve its goal. It is divided into five parts. First, the hand landmarks, neural networks, and robot control are explained. After, the data acquisition and interfacing of modules are presented.

A. Hand Landmarks

The precise localization of 21 key points over the hand is achieved inside the detected hand region. following are the 21 key points defined as hand landmarks.

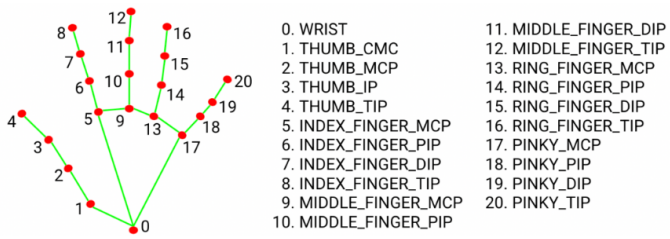


Fig. 1: Hand landmarks (MediaPipe)

Each key point has coordinates composed of X, Y, and Z. The Z coordinate gives the depth of the landmark, and the X and Y coordinates are first converted to relative coordinates concerning key point zero. These coordinates are then flattened to a one-dimensional array and normalized to a maximum value (absolute value). The normalized coordinates obtained from this process are used for further applications involving the analysis or interpretation of hand landmarks.

B. Neural Networks

Neural networks are widely used for tasks related to image recognition. In particular, Convolutional Neural Network (CNN) is a deep learning algorithm used for image processing and analysis [3] & [4]. It performs image segmentation and mathematical operations on the image to extract relevant features and classify images. It works efficiently as its convolutional layers have fewer parameters compared to the fully connected layers of a traditional neural network.

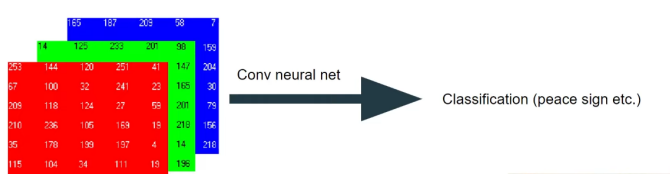


Fig. 2: Classifying hand gestures directly using CNN

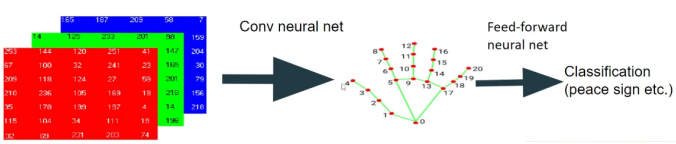


Fig. 3: Classifying hand gestures using CNN and feed-forward net

Training a network to predict hand gestures from the image using CNN can be achieved using the following approaches:

- Extracting RGB color matrices from the image and applying a CNN to classify the hand gestures by extracting features directly from these matrices
- Extracting the RGB color matrices from the image and applying a CNN to get hand landmarks. Then, training and applying a feed-forward neural net to detect hand gestures from these landmarks

Training the network to identify images using the latter approach is significantly easier and more efficient than training the network just on pixels. Different images have different camera resolutions, variable hand shapes and sizes, and contrasting backgrounds and lighting. Due to these factors, training the network just on pixels requires much more training data.

For this project, a CNN is used to get the hand landmarks from the image data or RGB values. Instead of passing the whole image to the CNN, the image is first cropped and only the significant part with the hand data is passed while the rest of the image with just the background is removed. This is achieved using a hand-detection neural network. Implementation of both these networks is done using the MediaPipe library [ref].

Then a small feed-forward net is trained to identify the gestures from the landmarks. Here, the neural network from [2] is used as a base for the model. It is further modified as per the required functionality and more gestures and training data are added. The training data required to train the network by this approach is way less and the model is more efficient and computationally effective.

C. Robot Control

The mIDOG is a quadruped robot with eight servos in its legs which allows a variety of actions and movements. The robot controller has some predefined functions that can be directly used to perform commands and gaits using the servos and the sensors. These functions are used to execute some basic and viable actions based on hand movement.

The actions executed by the robot for the corresponding hand gestures are as mentioned in Table I.

TABLE I: Hand Gestures and corresponding movements of the robot

Hand Gesture	Executed Actions
open palm	stop
close palm	walk forward
OK sign	greeting
rock and roll action	stretch
thumbs up	sit
index finger	crawl left
pinky finger	crawl right
index and middle fingers	sleep
3 fingers	walk backward

The number of gestures that the robot can react to can be increased and corresponding actions can be assigned. However, as the number of labels or gestures increases, the network requires more data to correctly classify the labels.

Otherwise, the accuracy of the network decreases and it will predict incorrect gestures.

Also, the robot controller is programmed with a delay to detect a second gesture after the first command is received. If the delay is not included, the controller will try to change actions quickly when the gestures are changed rapidly, which will lead to re-calibration of the robot or even damage to the servo motors. Also, there might be delays in micro-controller processing and publishing of the image or due to neural network processing.

D. Data Acquisition

The normalized coordinates of the hand landmarks are stored as a CSV file in a training dataset enabling simple manipulation and access. This file serves as the important input for the machine learning model improving its ability to learn and anticipate scenarios based on the variety of hand gestures that were recorded throughout the training phase. A portion of the CSV file that is created for new hand gestures can be seen in the image below.

```
6,0.0,0.0,-0.15853658536585366,-0.10975609756097561,-0.24390243902439024,-0.32926829268292684
6,0.0,0.0,-0.16265060240963855,-0.0963855421686747,-0.25301204819277107,-0.3253012048192771,-
6,0.0,0.0,-0.15217391304347827,-0.09782608695652174,-0.22282608695652173,-0.32065217391304346
6,0.0,0.0,-0.14871794871794872,-0.1128205128205128,-0.2153846153846154,-0.3128205128205128,-
6,0.0,0.0,-0.13636363636363635,-0.11616161616161616,-0.20202020202020202,-0.30808080808080807
6,0.0,0.0,-0.14646464646464646,-0.11616161616161616,-0.20707070707070707,-0.30303030303030304
6,0.0,0.0,-0.155,-0.105,-0.22,-0.3,-0.145,-0.445,-0.045,-0.54,-0.185,-0.55,-0.29,-0.75,-0.34,
6,0.0,0.0,-0.15639810426540285,-0.0947867298578199,-0.22748815165876776,-0.27488151658767773,
7,0.0,0.0,-0.09289617486338798,-0.09289617486338798,-0.1092896174863388,-0.2459016393442623,0
7,0.0,0.0,-0.08695652173913043,-0.09239130434782608,-0.10326086956521739,-0.2554347826086957,
7,0.0,0.0,-0.085561497326203 2,-0.085561497326203 2,-0.10695187165775401,-0.25133689839572193,0
7,0.0,0.0,-0.09183673469387756,-0.07653061224489796,-0.11224489795918367,-0.23469387755102042
7,0.0,0.0,-0.0861244019138756,-0.07655502392344497,-0.1100478468899521 6,-0.2200956937799043 2,
7,0.0,0.0,-0.08597285067873303,-0.07239819004524888,-0.10407239819004525,-0.22171945701357465
7,0.0,0.0,-0.09913793103448276,-0.06896551724137931,-0.1336206896551724,-0.21551724137931033,
7,0.0,0.0,-0.10121457489878542,-0.06072874493927125,-0.13765182186234817,-0.19838056680161945
```

Fig. 4: Training dataset stored as a .csv file

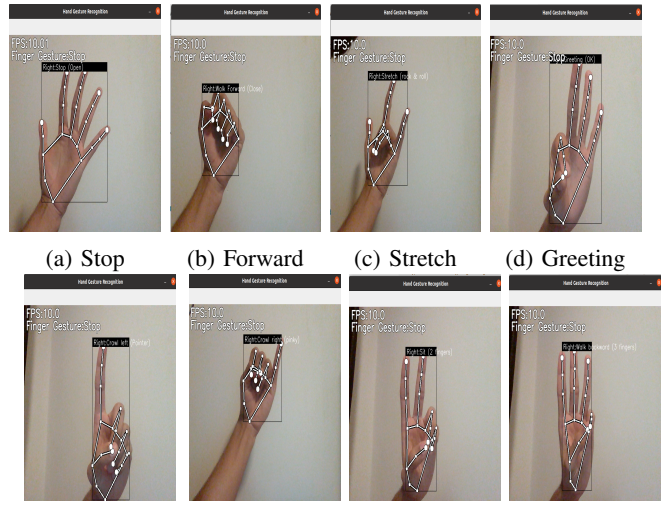
Another file in the repository contains various hand gesture labels with their respective index. The first column in the above key points file denotes the gesture labels, while the 2nd and subsequent columns show keypoints coordinates. . By selecting 'K' on the keyboard, the implemented code allows the entry of training data and switches to logging or recording mode. When a user clicks on an index number that corresponds to a certain hand gesture label, a snapshot is taken and the hand landmarks are saved in the training dataset under the corresponding gesture label. This allows for effective model learning.

For this project, nine labels of hand gestures are created as mentioned in Table I. ~~Then,~~ the training data set for all nine labels by the above method was created considering different backgrounds, lighting, and other factors and the model was successfully trained with the new training data set. Following are the results obtained after training the model with the new data set.

E. Interfacing Modules

The neural networks, OpenCV library, and robot control commands are combined into a system and executed using the ROS framework.

The robot camera publishes the live image data with a bit of delay which is subscribed to by a subscriber node. This



(a) Stop (b) Forward (c) Stretch (d) Greeting

(e) Crawl left (f) Crawl right (g) Sleep (h) Backward

Fig. 5: results from the trained model

data is then processed and OpenCV methods are used to get a bounding rectangle, landmark list, info text, drawing landmarks, and other required information, and the image is cropped to relevant information. The CNN is applied to this image and hand landmarks are obtained. It is then fed to the feed-forward net for gesture recognition.

Now, the identified gesture with maximum confidence is published onto a topic. The robot controller node then subscribes to this topic and gets the gesture identified. It then commands the Raspberry Pi to execute actions like walking or crawling corresponding to the gesture.

IV. Results

V. Conclusion

Acknowledgement

The authors of this paper are extremely grateful to the IDAC at TUHH for providing the 3D model and technology needed to carry out their work.

References

[1] Ali Suryaperdana Agoes, Indriani, Moh. Harris, "Applying Hand Gesture Recognition for User Guide Application Using MediaPipe", 2nd International Seminar of Science and Applied Technology (ISSAT 2021).

[2] GitHub: Hand Gesture Recognition - https://github.com/kinivi/hand-gesture-recognition-mediapipe

[3] Gongfa Li, Heng Tang, Ying Sun, et. al. "Hand gesture recognition based on convolution neural network", Published: 29 December 2017

[4] Hsien-I Lin, Ming-Hsiang Hsu, Wei-Kai Chen, "Human hand gesture recognition using a convolution neural network", 2014 IEEE International Conference on Automation Science and Engineering (CASE)

[5] Vaibhav Mudgal Real-Time Gesture Recognition Using GOOGLE'S MediaPipe Hands - https://mudgalvaibhav.medium.com/real-time-gesture-recognition-using-googles-mediapipe-hands-add-your-own-gestures-tutorial-1-dd7f14169c19

[6] Elena Nazarova; Ildar Babataev; Nipun Weerakkodi; Aleksey Fedoseev; Dzmitry Tsetserukou, "HyperPalm: DNN-based hand gesture recognition interface for intelligent communication with quadruped robot in 3D space", 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)