**1**

Hello everyone, we are the third team of Monty Matlab. Today, through our presentation, we will provide a detailed introduction to our project. We are excited to share our code achievements with you. Our team has been working diligently to bring you a comprehensive overview. So, without further ado, let's dive into the exciting world of our project.

**2**

We will discuss several key aspects to give you an overview of our project. First, let's dive into the introduction and motivation behind our work.

**3**

In this project, our main focus is on the classification of walking patterns, specifically distinguishing between normal walks and Monty Python’s Silly Walk using acceleration data. We believe that this topic holds great importance due to the accuracy required for such classification and its relevance in capturing human movement effectively.

**4**

Next is our motivation. By classifying walking patterns, we can contribute to the improvement of human activity recognition, which is crucial for health monitoring. This technology can be widely used to develop many applications, such as fitness tracking, gait analysis, and abnormality monitoring. Therefore, the motivation behind our project is to develop an automated system that can accurately distinguish between normal walks and silly walks.

**5**

Moving on to our workflow and timeline, we carefully planned and executed our project in a systematic manner. We divided our project into different stages, each with specific tasks and timelines. This allowed us to streamline our work. The timeline covered tasks such as data collection, data pre-processing, model training, report writing, video creation, and the development of a graphical user interface (GUI). This organized approach helped us stay on track and complete our project successfully.

**6**

Besides, team collaboration is essential for our project's success. We have implemented effective strategies to ensure strong collaboration throughout the project.

Weekly meetings are our platform for progress updates, problem-solving, and assistance. Each team member is responsible for a specific part of the project, providing focused expertise and regular updates.

Collaboration and support are key. We help each other with difficulties, share knowledge and resources, and foster a supportive environment.

Our project timeline is carefully planned for efficiency. Through teamwork, regular communication, and coordination, we aim to complete the project one week before the deadline.

In summary, our team's collaboration includes weekly meetings, individual responsibilities, collaboration and support, and adherence to the project timeline. Together, we strive for exceptional results and successful project delivery.

**7**

Next, let's explore data preprocessing. First is the data collection. We collected acceleration data in directions X,Y,Z, using MATLAB Mobile, an APP that allowed us to capture the necessary information for our analysis.

**8**

Then is the data extraction phase. During this process, we cut off pre-and post walks, ensuring that only effective walks were contained in the data set. This step was essential to obtain accurate results of our classification models.

**9**

Next is the part of model selection, in which we evaluated two algorithms, namely the k-nearest neighbors (k-NN) algorithm and the long short-term memory (LSTM) neural network.

**10**

The k-NN is one of the most common used machine learning algorithms, which classifies the object by a purely vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. One of the advantages of k-NN is its shorter training time compared to LSTM. However, as a machine learning algorithm, feature extraction is an essential preprocessing step. The classification results highly depends on what kind of features is chosen to train the model.

**11**

This is our settings for the k-nn algorithm. After fine-tuning, we finally chose k=10 to show the best result. We selected the mean values, standard deviation, sum of magnitude, range of magnitude and also the maximal magnitude of X, Y, Z data, respectively, as the features used to train the model.

**12**

This is the classification result. The accuracy is about 88.6%. A more intuitive confusion chart will also be shown after running the corresponding MATLAB script.

**13**

On the other hand, The LSTM stands for Long Short-Term Memory, and it excels at capturing long-term dependencies in sequences. One of the remarkable features of the LSTM is its ability to handle variable-length sequences. Whether you're dealing with short or long sequences, the LSTM can adapt and process them effectively. Another advantage of the LSTM is its capability to automatically extract relevant features from raw data. This means that you don't have to manually engineer features; the LSTM can learn to identify the most important information for your analysis. Furthermore, the LSTM is well-suited for learning from large datasets. With its capacity to handle vast amounts of data, the LSTM can uncover complex patterns and relationships that might be hidden in the data.

**14**

Our LSTM neural network was trained with a learning rate of 0.001, a batch size of 32, 120 hidden units, and a dropout rate of 0.5.

**15**

The training process and unit test results demonstrated the effectiveness of the LSTM model in accurately classifying normal walks and silly walks, with an accuracy about 97.7%.

**16**

To provide a user-friendly interface for our project, we also developed a graphical user interface (GUI) that allows users to interact with the classification system.

**17**

Welcome to Silly Walk Detection Lab. In this Lab, you are able to check your gait, normal or silly. And also if you get interested in knowing what exactly is silly walk, we offer a GIF to present the typical Silly Walk, what you need to do is to click the ‘How does Silly Walk look like’ button and watch it.

Besides this functionality, the most attractive thing is that you can import your gait data by clicking ‘Sure! Import my gait!’ button, then you will be directed to the second part of our GUI and be able to import your expected gait file. Afterwards, the gait data and test result will be displayed. This top illustrated diagram is your acceleration data represented in their directions and the image in the bottom left corner will tell you if your uploaded gait is normal or not. If your gait is considered as normal, this image will display a person standing normally, and in the middle, it will show the corresponding result next to it and the following text: "Congratulations! Based on our experimental results, your gait is considered normal. Please continue to maintain it!". On the contrary, if your gait is tested as silly, the figure in the bottom left corner will be a person with a high leg lift and the middle text will display ‘Based on our experimental results, your gait is considered as silly, and we recommend reducing the amplitude of body sway’. Furthermore, on the bottom right side, the elapsed classification time is displayed. If you still want to test another gait data, you are supposed to click ‘Choose another data’ button. The GUI we designed is primarily aimed at user convenience, providing a simple interface for users to quickly get started.

**18**

In conclusion, our project successfully achieved the goal of classifying walking patterns into normal walks and silly walks. Through a comprehensive analysis, we compared the performance of the k-NN algorithm and the LSTM neural network, ultimately selecting the LSTM model due to its higher accuracy.

In terms of future work, there are several areas we can explore. For instance, we can further improve the k-NN model by finding better features. Additionally, we can also extend our classification system to differentiate between different types of silly walks.