Technische Universität Berlin

Gesture Recognition for Human-Robot Interaction: An approach based on skeletal points tracking using depth camera

Masterarbeit
am Fachgebiet Agententechnologien in betrieblichen Anwendungen und der Telekommunikation (AOT)
Prof. Dr.-Ing. habil. Sahin Albayrak
Fakultät IV Elektrotechnik und Informatik
Technische Universität Berlin

vorgelegt von
Sivalingam Panchadcharam Aravinth

Betreuer: Prof. Dr.-Ing. habil. Sahin Albayrak,
Dr.-Ing. Yuan Xu

Sivalingam Panchadcharam Aravinth
Matrikelnummer: 342899
Sparrstr. 9
13353 Berlin

Statement of Authorship

I declare that I have used no other sources and aids other than those indicated. All passages quoted from publications or paraphrased from these sources are indicated as such, i.e. cited and/or attributed. This thesis was not submitted in any form for another degree or diploma at any university or other institution of tertiary education

Place, Date

Signature

Abstract

Human-robot interaction (HRI) has been a topic of both science fiction and academic speculation even before any robots existed [**?**]. HRI research is focusing to build an intuitive and easy communication with the robot through speech, gestures, and facial expressions. The use of hand gestures provides an attractive alternative to complex interfaced devices for HRI. In particular, visual interpretation of hand gestures can help in achieving the ease and naturalness desired for HRI. This has motivated a very active research concerned with computer vision-based analysis and interpretation of hand gestures. Important differences in the gesture interpretation approaches arise depending on whether 3D based model or appearance based model of the gesture is used [**?**].

In this thesis, we attempt to implement the hand gesture recognition for robots with modeling, training, analyzing and recognizing gestures based on computer vision and machine learning techniques. Additionally, 3D based gesture modeling with skeletal points tracking will be used. As a result, on the one side, gestures will be used command the robot to execute certain actions and on the other side, gestures will be translated and spoken out by the robot.

We further hope to provide a platform to integrate Sign Language Translation to assist people with hearing and speech disabilities. However, further implementations and training data are needed to use this platform as a full fledged Sign Language Translator.

Keywords

Human-Robot Interaction (HRI), NAO, Computer Vision, Depth Camera, Hand Gesture, 3D hand based model, Skeleton tracking, Gesture Recognition, Sign Language Translation, Naive Bayes Classifier, Gesture Recognition Toolkit (GRT)

Acknowledgements

Der Punkt Acknowledgements erlaubt es, persönliche Worte festzuhalten, wie etwa:

- Für die immer freundliche Unterstützung bei der Anfertigung dieser Arbeit danke ich insbesondere...
- Hiermit danke ich den Verfassern dieser Vorlage, für Ihre unendlichen Bemühungen, mich und meine Arbeit zu foerdern.
- Ich widme diese Arbeit

Die Acknowledgements sollte stets mit großer Sorgfalt formuliert werden. Sehr leicht kann hier viel Porzellan zerschlagen werden. Wichtige Punkte sind die vollständige Erwähnung aller wichtigen Helfer sowie das Einhalten der Reihenfolge Ihrer Wichtigkeit. Das Fehlen bzw. die Hintanstellung von Personen drückt einen scharfen Tadel aus (und sollte vermieden werden).

Contents

Statement of Authorship **II**

Abstract **III**

Contents **V**

1 Evaluation **1**

1.1 Mean and Standard Deviation 1

1.2 Classification and Prediction 2

1.3 Prediction Accuracy Vs Null Rejection Accuracy 4

2 Conclusion and Future work **8**

2.1 Future Work 9

Bibliography **11**

List of Figures **12**

List of Tables **15**

Abbreviations **16**

Chapter 1

Evaluation

We have used a machine learning toolkit named as GRT to recognize hand gestures using skeletal points tracking algorithm and Adaptive Naive Bayes classifier (ANBC) for classification and prediction.

The classifier is based on a statistical model of x,y,z coordinate positions of static hand gestures and provides a likelihood measure for recognized gesture. Furthermore, the gesture recognition pipeline uses two post processing modules such as Class Label Filter and Class Label Change Filter to exclude lower frequent spikes in the prediction results and trigger an output only when there is a change in prediction.

In this chapter, we present the experiments carried out to evaluate and validate our system to recognize hand gestures using skeletal points. The goal is to demonstrate the effectiveness of the classifier and to evaluate its potential for real time input at 30 fps. In the classification phase, input samples are normalized using Min-Max Scaling and Null Rejection is enabled to detect non-gestures. Therefore, the evaluation consists of computing the prediction accuracy for various null rejection coefficient and compare it with other supervised learning classifiers such as Support Vector Machine (SVM).

1.1 Mean and Standard Deviation

ANBC is a supervised learning algorithm that can be used to classify any type of N-dimensional signal. It fundamentally works by fitting an N-dimensional Gaussian distribution to each class when it is trained.

During the training phase, first all the input samples are normalized using Min-Max Scaling with the range from 0 to 1 and then GRT computes mean μ and standard deviation σ to create a model for each class. During the prediction phase, it basically computes the maximum a posterior probability of an input vector belonging to any of

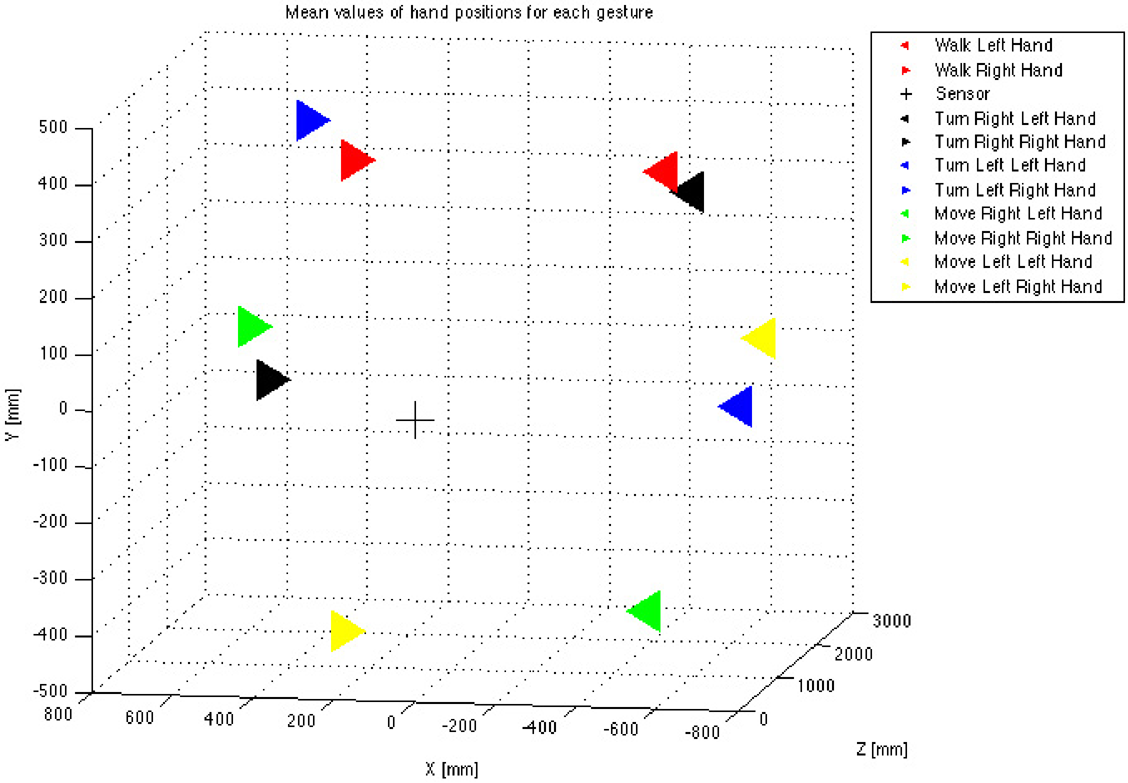


Figure 1.1: Mean values of hand positions for each gesture

the trained class. Figure 1.1 shows the mean positions of left and right hand for every gesture. Table 1.1 and 1.2 show mean and standard deviations of the labeled training data of all the five classes.

Class Label	Left X	Left Y	Left Z	Right X	Right Y	Right Z
1	0.55	0.76	0.76	0.57	0.76	0.78
2	0.48	0.73	0.78	0.75	0.51	0.79
3	0.36	0.42	0.78	0.67	0.8	0.8
4	0.58	0.13	0.73	0.79	0.57	0.79
5	0.29	0.52	0.79	0.58	0.24	0.72

Table 1.1: Normalized mean values of 3 dimensions of left and right hand

1.2 Classification and Prediction

Our gesture recognition pipeline is trained with 11918 input samples of 6 dimensional vector for 5 classes. Class 1,2,3,4,5 are mapped to Walk, Turn Right, Turn Left, Move

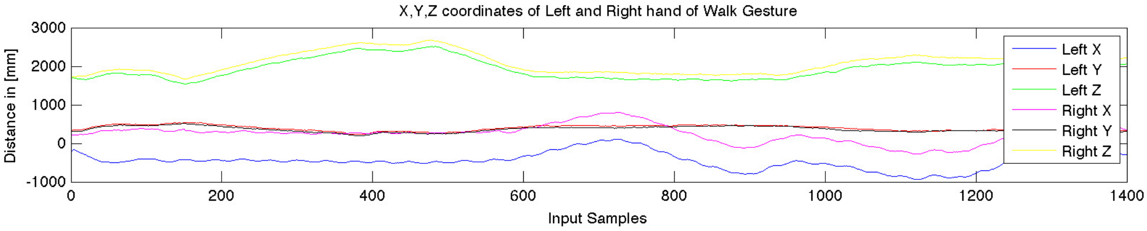
Class Label	Left X	Left Y	Left Z	Right X	Right Y	Right Z
1.000	0.261	0.116	0.079	0.225	0.092	0.083
2.000	0.189	0.086	0.075	0.149	0.053	0.080
3.000	0.178	0.072	0.088	0.141	0.079	0.093
4.000	0.182	0.060	0.076	0.159	0.070	0.089
5.000	0.128	0.102	0.088	0.114	0.061	0.083

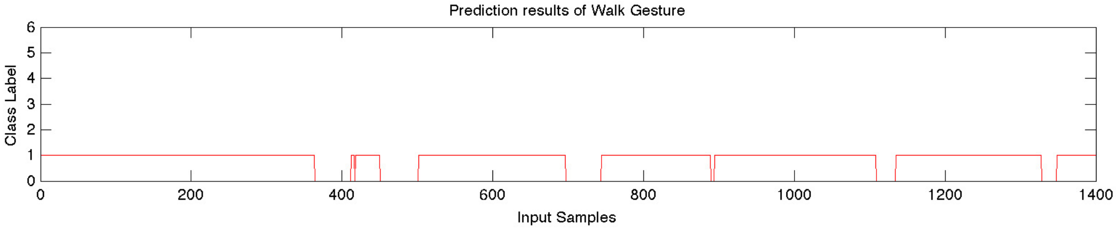
Table 1.2: Standard deviations of 3 dimensions of left and right hand

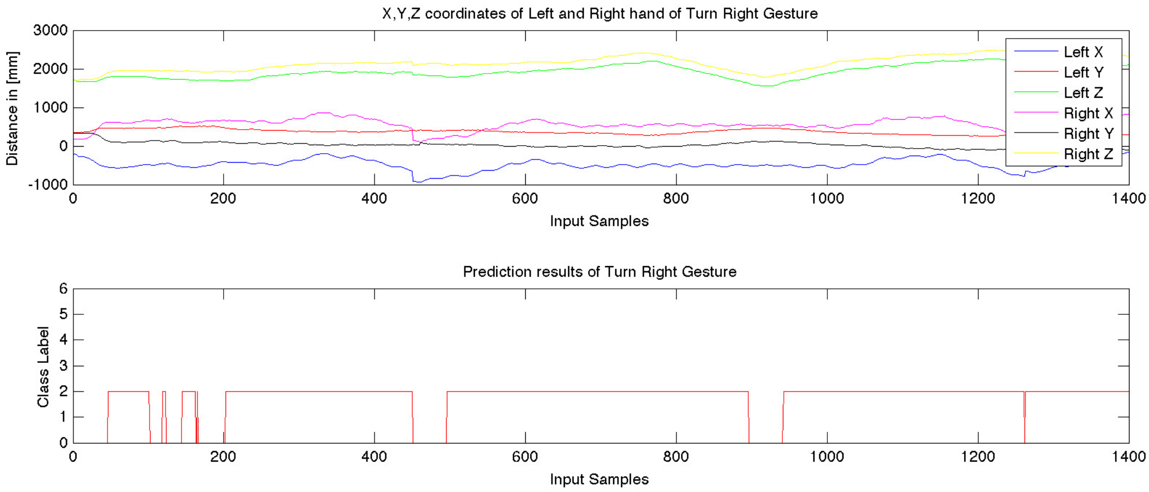
Right, Move Left gestures respectively. We have carried out experiments to evaluate the classification, prediction and post processing efficiency of our system. Graph 1.2 show change is positions of left and hand in Cartesian coordinates while test data was recorded and corresponding prediction for every input sample.

Test data consists of 1400 samples which are recorded under supervised arrangement. Input vectors that is not containing left and right hand are removed from the test data. Furthermore, input vectors which were recorded when the hand is at the field of view of the camera are also excluded. A program was implemented with GRT to read all the test data and execute prediction on every input sample and then results are stored to CSV file. Finally results are plotted using MATLAB.

Prediction results shown in the graph 1.2 frequently falls down to Class Label 0 that is reserved for non-gestures. Non-gestures are detected with the help of Null Rejection thresholds. Therefore, this prediction is based on normalized classification data for Adaptive Naive Bayes classifier with Null rejection coefficient of 1.0.







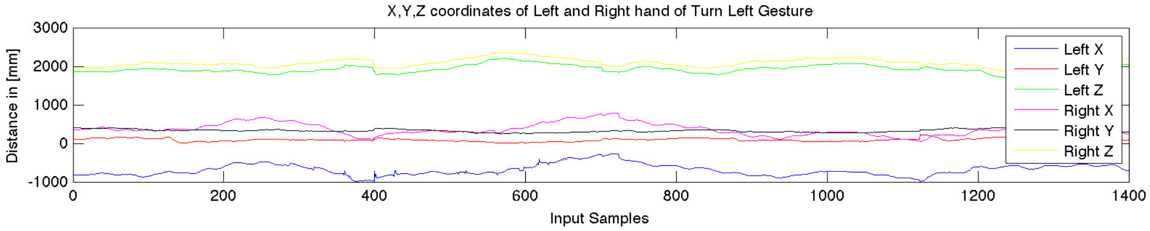
1.3 Prediction Accuracy Vs Null Rejection Accuracy

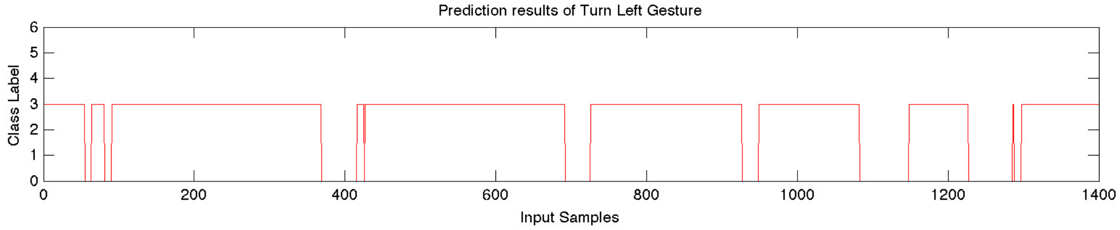
Classifiers of GRT offers various customization that could produce different results for the same test data. Accuracy of a gesture recognition system does not depend only on th precise predictions of trained gestures, but also differentiating them from unintended hand gestures. Graph 1.2 shows the prediction and null rejection accuracies of Adaptive Naive Bayes Classifier (ANBC). Graph shows the accuracies of 5 trained gestures and a non-gesture. Non-Gesture data was recorded with both the hands are pointing downwards. GRT allows us to set null rejection coefficient for the classifier and therefore, graph contains the dataset of accuracies with null rejection coefficients range from 0 to 10.

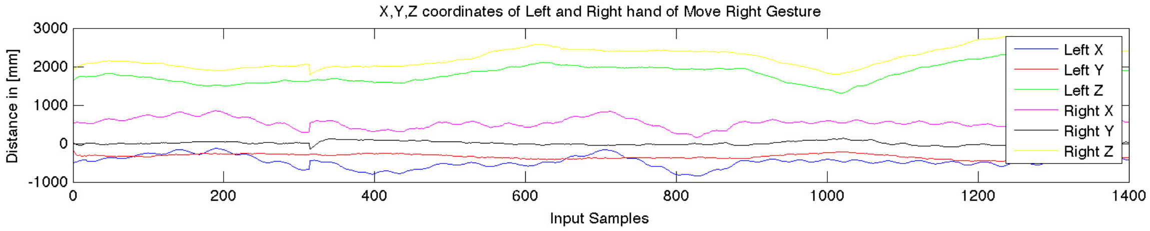
Graph 1.2 shows that increase in null rejection coefficient causes an increase in the accuracy of trained gestures, however, causes a decrease in the accuracy of non-gesture. Therefore, it is optimal to use a null rejection coefficient of 2.0 with ANBC.

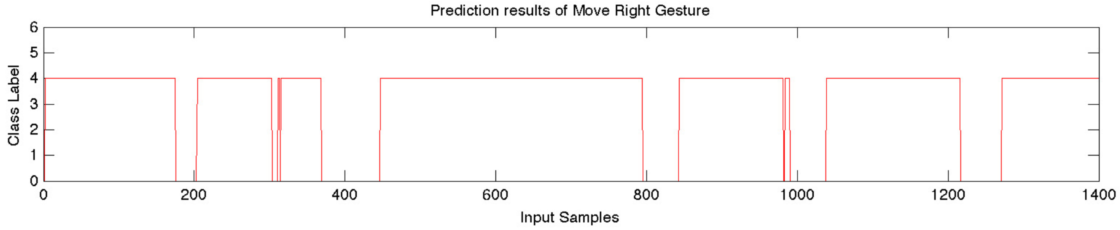
Graph 1.3 shows prediction results of Support Vector Machine classifier with RBF Kernel. It is apparent that accuracy of trained gestures are consistently above 95 percent. However, null rejection accuracy is constantly lesser than 10 percent, thus, denoting that SVM is not useful in our case, because the robot should not execute any unintended commands.

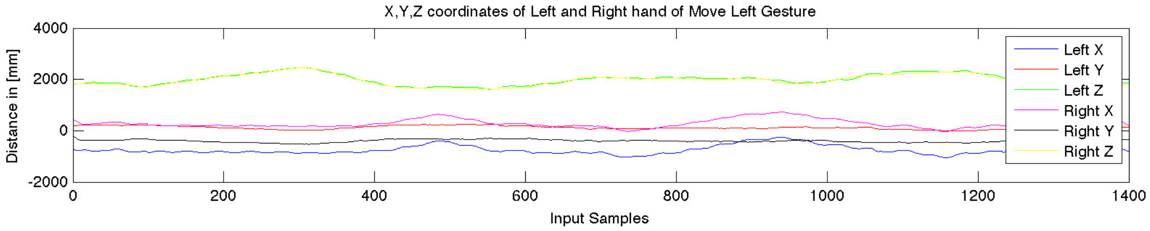
Graph 1.4 shows some interesting results of Minimum Distance classifier with 4 clusters. Graph shows that prediction results are unpredictable as there is increase in null rejection coefficient. However, with coefficient of around 6.3, MinDist shows compromising accuracy above 90 percent for trained gesture and non-gesture. This shows that MinDist is a better alternative to ANBC.

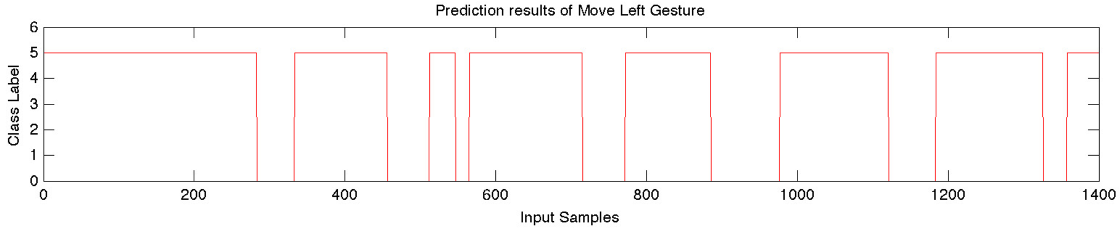

X,Y,Z coordinates of Left and Right hand of Turn Left Gesture
Distance in [mm]
3000
2000
1000
0
-1000
0
200
400
600
800
1000
1200
1400
Input Samples
Left X
Left Y
Left Z
Right X
Right Y
Right Z


Prediction results of Turn Left Gesture
Class Label
6
5
4
3
2
1
0
0
200
400
600
800
1000
1200
1400
Input Samples


X,Y,Z coordinates of Left and Right hand of Move Right Gesture
Distance in [mm]
3000
2000
1000
0
-1000
0
200
400
600
800
1000
1200
1400
Input Samples
Left X
Left Y
Left Z
Right X
Right Y
Right Z


Prediction results of Move Right Gesture
Class Label
6
5
4
3
2
1
0
0
200
400
600
800
1000
1200
1400
Input Samples


X,Y,Z coordinates of Left and Right hand of Move Left Gesture
Distance in [mm]
4000
2000
0
-2000
0
200
400
600
800
1000
1200
1400
Input Samples
Left X
Left Y
Left Z
Right X
Right Y
Right Z


Prediction results of Move Left Gesture
Class Label
6
5
4
3
2
1
0
0
200
400
600
800
1000
1200
1400
Input Samples

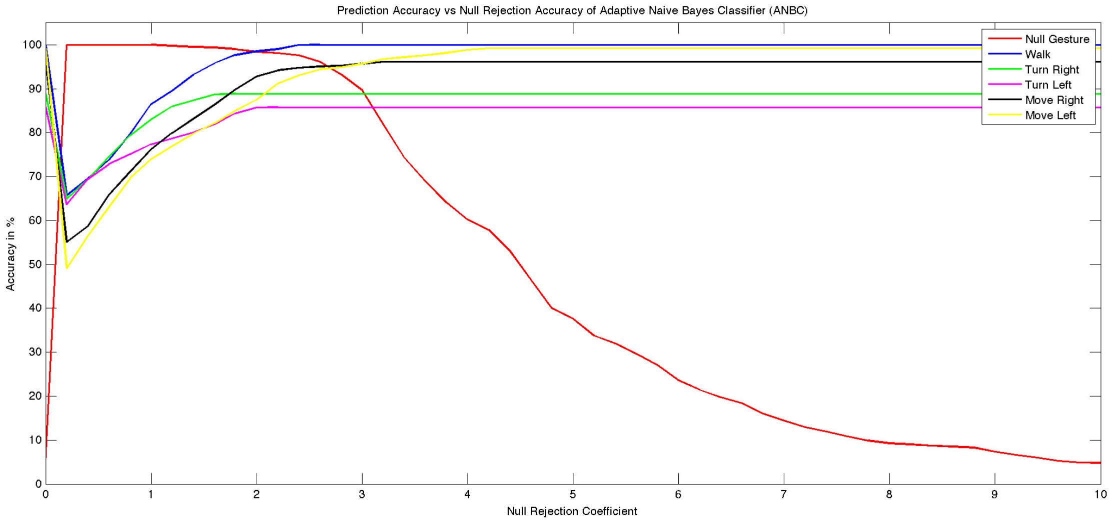


Figure 1.2: Prediction vs Null Rejection of ANBC

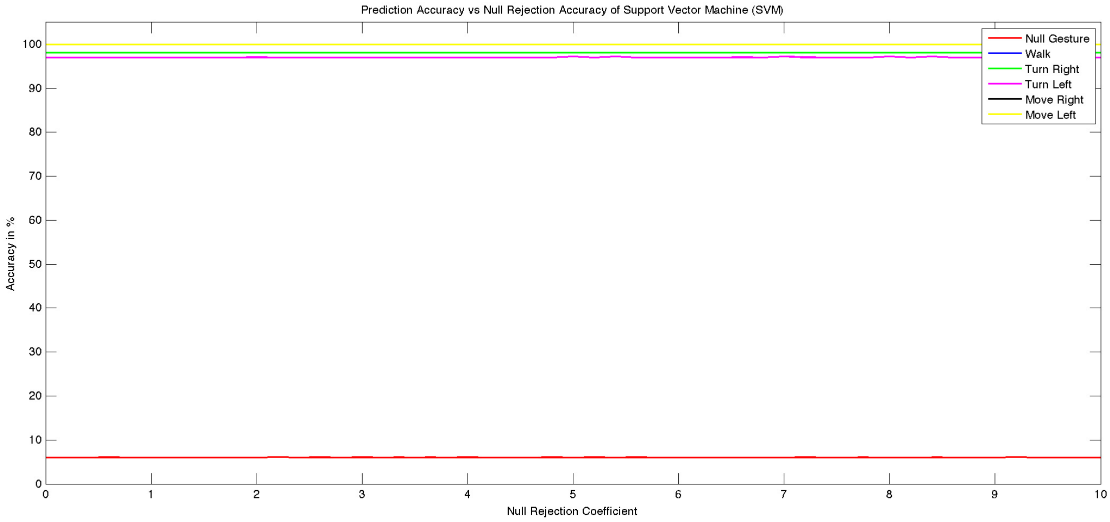


Figure 1.3: Prediction vs Null Rejection of SVM

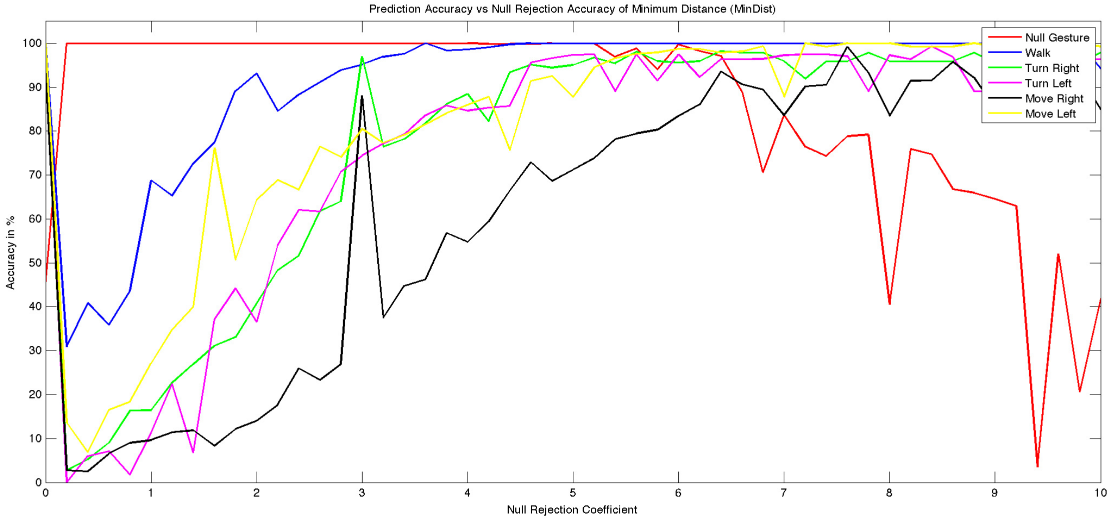


Figure 1.4: Prediction vs Null Rejection of MinDist

Chapter 2

Conclusion and Future work

During this thesis, we have proposed a promising system to recognize hand gestures based on skeletal points tracking using depth camera. This system is built for the purpose of human-robot interaction (HRI) with the humanoid robot named as NAO. We have validated this approach by training the system with five static gestures and we have obtained good results.

We have partitioned our goal into 4 modules and reached the goal by implementing them in a decoupled environment, and finally, integrated all of them into one system. We believe that the proposed approach is sufficiently robust and flexible to deal with static and temporal hand gestures.

Human-Robot Interaction module deals with integrating the depth camera into the robot and processing the depth information to track skeletal points of the user, and finally send them via network to Brain module. Brain module supports the core functionalities of this system by receiving the skeletal point information of the user, and recording them as training data in the training mode or computing the prediction results in the prediction mode. Control Center module plays vital role in visualizing these interactions, therefore, it is considered as the eye of the system. Finally, Command module comprehends the predicted gesture and translating them to a robotic Motion or Speech or Gesture itself.

Finally, we have carried out several experiments and provided the results illustrates the robustness of the system. Furthermore, we have done evaluations on the test data and plotted them on graph to visually understand the performance of the system. Moreover, we have proposed alternatives to improve the efficiency and the effectiveness of our system. For instance, the gesture recognizing performance can be then improved incrementally by learning from on-line training feature of Adaptive Naive Bayes classifier. Additionally, evaluation results show that Minimum Distance classifier could be a better alternative.

2.1 Future Work

The proposed design for gesture recognition based on skeletal points tracking using depth camera can be improved in several ways. In this section, we overview the future work by discussing the limitations of the proposed methods and proposing the alternatives.

- **Networking** : 4 modules of this system is connected via several networking components such as UDP Server-Client, Websocket Server-Client and NAOqi proxy. Server-Client networking topology involving different communication protocol is a big limitation, since every module have implemented their own server or client functionalities of UDP or WebSocket. Furthermore, data is serialized as JSON strings and must be parsed at the receiver side to extract the data. Hence, we propose Open Sound Control (OSC) protocol that covers all these requirements in one framework with distributed networking topology.

- **Graphical User Interface** : Since the components of this system are modularized and developed independently, the development environment varies with each other. Even though WebGL is easier and flexible graphics library, it takes a lot of processing power as it runs on the browser. Instead of using several programming languages, the system could be implemented with C++ GUI using QT or Microsoft Visual C++ and OpenGL. We propose to integrate the existing code into such framework to build this system as standalone application.

- **Skeleton Joints** : Due to computational limitations of NAO, in this thesis, we have used only hand joints of the user to train and classify the gestures. Classification based on positions of only hand joints does not allow us to train many gestures because the gesture model will higher rate of confusion. For instance, in Cartesian coordinates "Hands Up" gesture trained at different distances from the sensor will be confused with "Hands Wide". Therefore, we propose to make use of other skeleton joints such as shoulder and arm to calculate the orientation of hand in polar coordinates.

- **Computational Limitations of NAO** : In this thesis, HRI module is deployed to general purpose computer of NAO. HRI module is responsible for tracking the hand or full skeleton joints with the help of NiTE framework. NiTE uses computationally intensive algorithms and causes higher CPU utilization of NAO. Therefore, we propose to transmit the depth information from OpenNI device completely to NiTE application on an off-board computer. This could be achieved by using Robot Operating System (ROS) framework that already has a solution to transmit depth information via network.

Bibliography

List of Figures

1.1 Mean values of hand positions for each gesture 2
1.2 Prediction vs Null Rejection of ANBC 6
1.3 Prediction vs Null Rejection of SVM 6
1.4 Prediction vs Null Rejection of MinDist 7

List of Tables

1.1 Normalized mean values of 3 dimensions of left and right hand 2
1.2 Standard deviations of 3 dimensions of left and right hand 3

Abbreviations

HRI	Human-Robot Interaction
OpenNI	Open Natural Interaction
NiTE	Natural Interaction Technology for End-user
GRT	Gesture Recognition Toolkit
CC	Control Center
UDP	User Datagram Protocol
WLAN	Wireless Local Area Network
FOV	Field Of View
JSON	JavaScript Object Notation
DOF	Degrees Of Freedom
TTS	Text-To-Speech
API	Application Program Interface
DP	Dynamic Programming
MAP	Maximum A Posterior Probability
CSV	Comma Separated Values
DTW	Dynamic Time Warping
HMM	Hidden Markov Models
KNN	K-Nearest Neighbor
SVM	Support Vector Machines
PCA	Principal Component Analysis
GUI	Graphical User Interface
IDE	Integrated Development Environment