Abstract
This document contains many important details that may help users to get a brief knowledge about the NLCT system. The very begging topic is Introduction; it gives the clear idea about the system. The next topic Methodology describes the system by using seven separate steps. Results and Discussion section points out all the possible executable and outputs. And a conclusion explains overall picture of the system. Finally references points out the helping materials, which have being used for this project.

Keywords:
Khepera robotic simulator.
Festival synthesizer.
HTK: Hidden Maroo model Took Kit.

I. INTRODUCTION

Our main goal was to develop existing Khepera simulator by adding several features. Khepera is a robotic simulator. That can travel through the given environment by detecting obstacles. But there were not any features to control by users. We developed it by adding user control features such as text inputs, voice commands, Warning messages.

II. METHODOLOGY

In this research we have identified seven major activities during the development. They are

- Background studies of base system
- Background studies of voice recognizer
- Background studies of voice synthesizer
- Develop the voice recognizer
- Develop the voice synthesizer
- Develop the base system
- Integrate both voice recognizer and base system.

But the analyzing part contained heavy work load, because all the components which we used were free developed versions. In order to develop those components to satisfy our requirements we had to do lot of background studies and analysis on codes and documentations of those components.

In here we can identify 3 separate components that we had to develop. They are

- Khepera simulator
- Voice recognizer
- Speech synthesizer

In analyzing phase we decided our voice recognizer as HTK tool kit and voice synthesizer as Festival tool kit.

In order to develop Khepera simulator, we had to rewrite the codings of the “graphic.c”, “sim.c” and “user.c” files. sim.c was the main program in the execution time and the user.c is responsible for taking the user commands. The responsibility of graphic.c is to represent those activities graphically.

Before use HTK voice recognizer inside NLCT we had to define a grammar file which contains formats of voice commands. And then we had to train the HTK to detect the voice commands that we are going to insert to the system.

Voice synthesizer is responsible for generating warning messages in a form of voice. The Festival synthesizer is able to convert a text input to a voice. What we had to do was, pass a text to the Festival when required to generate a warning. In addition to the warning message generation, it generates several other messages to increase the user friendliness.

Finally integrations part was done. HTK and NLCT interface detection is the basic part. HTK generates executable command. Always NLCT has to wait for new commands.

RESULTS

The NLCT was developed to perform specific actions by giving pre-define user commands. According to the user commands given in the following table the NLTC could be able to perform successfully.
Command Description

Forward Stepping robot until obstacle is detected. If not, it should stop after finishing its default steps completion.

Back Turn 180 degrees and stepping robot until obstacle is detected or If not, it should stop after finishing its default steps completion.

Slow If the current speed of the robot is not slow; then slow down the current speed of the robot.

Fast If the current speed of the robot is not fast; then accelerate the current speed of the robot.

Turn little left Turn robot 45 degrees to the left.

Turn little right Turn robot 45 degrees to the right.

Turn left Turn robot 90 degrees to the left.

Turn right Turn robot 90 degrees to the right.

Stop Stop the robot.

Exit Exit from the voice command mode.

By giving above pre-define user commands it can be demonstrated how the robot performs in graphically on main interface of the NLCT. While executing above mentioned commands, system can produce appropriate warning messages and instructions to the user in both voice and text format. Following table describes all the messages given by the system to the users.

<table>
<thead>
<tr>
<th>Message</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obstacle is near</td>
<td>When the NLCT is reaching to an obstacle</td>
</tr>
<tr>
<td>Obstacle detected</td>
<td>When detect an obstacle</td>
</tr>
<tr>
<td>You stop me</td>
<td>When user stop the NLCT</td>
</tr>
<tr>
<td>I decided to stop because pre-define two-hundred steps were run out.</td>
<td>When it is expired pre-defined steps.</td>
</tr>
<tr>
<td>The robot is turning to right</td>
<td>When the user gives the Turn right command to the robot.</td>
</tr>
<tr>
<td>The robot is turning to left</td>
<td>When the user gives the Turn left command to the robot.</td>
</tr>
</tbody>
</table>

IV. DISCUSSION

As mentioned above the main purpose of the project is to control the robot via voice. The main reason for using voice is the user convenience. This system is controlled by not only voice but also key board inputs. In earlier stages of the project the device was fully controlled by automatically according to the code segments of the “Khepera” simulator. That was not enough to perform actions similar to human movements. During the next step of the project development team added basic command buttons to give few number of basic commands to the system for further testing and development. By adding these basic command buttons development team was able to reach the primary aim of controlling the device by giving basic commands. After doing further analysis the development team realized that it would be difficult to the users when the object is traveling. When the device was traveling it was needed to turn by various degrees. Also the system should be able to stop at anywhere when the user needs. But the system was fully controlled by the basic command buttons “Forward”, “Back”, “Turn Left”, “Turn Right” & “Stop” on the main interface. At the beginning stages, if the user needs to turn the device to the left, right or stop user had to press the “Turn Left”, “Turn Right” or “Stop” command buttons.

When user presses the “Turn Left” button the device turns 90 degrees to the left. This was same to the “Turn Right” button. Other than that
the system was stopped after traveling specific distance specified inside the “Khepera” main code segment. When traveling between two destinations this situation could not be acceptable. It was needed to turn the device by various degrees and stop the device according to the terrain and user needs. This was not a good approach for a more convenient and a safety system. Then the development team began to think a new way to turn the device by various degrees to the left or right and stop the device at once by giving simple commands while it is traveling. In order to achieve the above goal it was needed to analyze the main code of the “Khepera” very deeply.

“Khepera” main code can be easily modified according to the user requirements. Further analyzing the code development team came in to a solution to the problem of stopping and turning the robot while traveling. The solution was to define 300 steps for the “Forward” command button. The “Forward” command is executed once then the robot moves to the forward up to 300 steps. If another command received while executing the “Forward” command then it stops at the current position of execution and began to execute the other command. Scene behind the above solution was using multiple process handling. At last development team realized multiple process handling was the ultimate solution for the problem of controlling the device while doing a one process.

After overcoming the above problems the team again began to build the system more user convenience. After that the aim of the team was to input instruction commands via voice. To give voice instructions there should be mechanism. By doing several experiments the team came in to a last decision to use a voice recognizer. Through a voice recognizer can translate a specific voice instruction in to a text format. At the first step of experiments system was again modified to input instruction commands through command lines.

The purpose of above modification was to check the capabilities of reading text inputs by the system. NLCT was responded to the above modification very successfully. By analyzing the results of above experiments the team decided to use command lines to give instructions to the system instead of using command buttons.

By looking at previous analysis the system analyzers realized that it could be difficult to the users to control the device via keyboard inputs. The system is always using in a traveling background. Some times users may take very quick decisions to change the route, control the speed according to the terrain, and stop the device in an urgent situation like in an accident. If the system is fully controlled by command switches it could be very difficult to operate and control by clicking or pressing a button. In such situations users might take some time to control the device through switches. That would not good for a system building for user convenience and safety. On the other hand user should have well practice about keyboard and a little computer literacy. By taking all these facts in to the account development team though to control the device through natural voice. By using natural voice such kind of a device can be controlled very simply. If the system accepts natural voice users doesn’t need to have a well practice about the system and also doesn’t need to have a well computer literacy. The development team always thinks of the user convenience and safety. It would provide a grate safety to the user because user doesn’t need to change the concentration to input commands while operating the device.

At last the team developed the system with taking a voice recognizer. Because of taking a voice recognizer the system had to use a high sensitive microphone to input natural language voice commands given by the user to the system. This voice recognizer converts natural language voice commands in to a text format. Converted text messages pass in to the system like a command line interpreter. Then the system performs as an assistant of the user.

When it comes at to the last of the development phase the project team tried to improve the system to make it more user convenience. In order to achieve this goal the project team had to add new functions to the developed system. In earlier developed system displayed a prompt on the main interface regarding the current movements of the device such as “Robot Traveling”, “Obstacle Detected” and when the device is idle “What should I do”. These messages cannot be seen by blind people. As mentioned above wheel chair is one of the best applications for the developed system. If the system is using by blind people they cannot see the current status of the device. To overcome above problem system was enhanced to produce warning voice messages to the users. Voice synthesizer is used inside the system to produce warning signals to users. Theory behind a recognizer is read a specific text and generates appropriate voice to that voice. According to the theory system generated messages are input to
the voice synthesizer and get the appropriate voice signal. Because of this method system can be act as a guide to the user and it will become more users friendly and convenient to use.

V. CONCLUSION

Main purpose of starting this project was introducing a new technology to the world to control a mobile device via voice and more user convenience system. Main fact for above aim it’s very expensive to develop a device which can be control remotely. Still there is not a device is being produced to control remotely via voice in the world. The project team mainly considered only to develop the simulator. To control such a robot or device due to limited time period and resources. After developing such software it can be easily embedded into such a device or a robot through a microchip. This is a very common technology using nowadays all over the world.

To over come above problems project team developed a system successfully. When developing the system priority was given to the user convenience. Considering of the user convenience system was developed to produce warning messages. These warning messages show the current status of the system to the user. Though the system shows warning text messages if the system is using by blind people they can’t see these text messages. This situation was not satisfied the user convenience. Project team solved above problem by producing the system to generate warning voice message even a blind person can here. Currently the system produces both warning text message and also warning voice messages. Hence their will be no conflicts occur if the system is used by visually impaired or hearing impaired person. After the above step system came too close to users.

On the other hand the system was able to control through voice by giving appropriate commands. Currently developed simulation software simulates the movements to appropriate commands given from the users. This was one of main goal achievements of the project team. The developed system responds to voice successfully. Hence the system was developed by using the khepera simulator as the base system it would be applied to any real world applications including obstacle detecting robots. The other reason was the system became more and more user friendly. Obstacle detecting robots and electronic wheel chairs are taking the leading place among applicable areas. More user convenience is the closest reason to apply the system in to number of real world applications. After applying the system to a real world application it can be operate very simply by giving natural voice commands and the device perform moving actions very similar to human movements with safety. This is the main target achievement of the project team.

NLCT was developed using C language. This is a one reason that the system can be applied in to many real world applications. C language can deal with hardware devices very easily.

At the beginning of the project the team expected movements from the NLCT which is very similar to human movements and a satisfied response from the system to natural voice commands. Human movements is a word used by the project team to generalized the basic idea to the audience for better understanding and regarding what the system does.

When take humans they can move ahead, back, turn to any ware while traveling, stop at any ware, increase or decrease the speed of the movement. Also when detect an obstacle they can travel by evading obstacles. Humans can perform all of these actions because they have a brain to think. But NLCT does not. Hence the team had to train the NLCT to perform some actions for specific commands. When comparing NLCT with humans they have eyes to detect and avoid obstacles. But NLCT uses sensors. There was another fact to think to the project team. Blind people can also use the system when it is applied in to application like wheel chair. In such situations NLCT must be able to inform about obstacles found on the way while the device is traveling. Normally humans don’t put their lives in to risks and dangers. They always are trying to avoid risks and danger situations from their lives. When the system applied in to application it should perform actions underlying these facts and provide enough safety to the users. When taking the NLCT it is performing best of its class.

ACKNOWLEDGMENT

In hear we would like to give our special thanks to the project supervisor Dr.Koliya Pulasinhe, subject in charge Mr.Jyantha lal Amararachchi and some of our friends who gave their friendly support to implement the NLCT by giving us advices, resources, sending us reminders, sharing the knowledge and for guiding us to the correct way.
REFERENCES

Ref Book 1:
The HTK Book (for HTK Version 3.3)
COPYRIGHT 2001-2005 Cambridge University
Engineering Department.

Ref Link 1:
http://www.cs.cmu.edu/afs/cs/project/ai
repository/ai/areas/speech/doc/notes/htk.txt

Ref Link 2:
http://www.cstr.ed.ac.uk/downloads/festival/1.9
5/

Ref Link 3:
http://www.speech.cs.cmu.edu/pocketsphinx/

Ref Link 4:
http://www.cstr.ed.ac.uk/projects/festival/

Ref Link 5:
http://www.faqs.org/docs/Linux-
HOWTO/Speech-Recognition-
HOWTO.html#BASICS