International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 06 | June - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

VIRTUAL JOYSTICK APPLICATION

MS.N.ANU LAVANYA, Assistant Professor,
K.TAMILARASAN , R.SUBBIRAMANI,D.MADHESH KUMAR, M.SIVA SHANTHANA BHATHI
Department of Computer Science and Engineering , SSM Institute of Engineering and Technology, Anna university

ABSTRACT

Virtual controller mobile apps have become increasingly popular in the gaming industry, offering users a virtual joystick on their mobile devices to control characters or vehicles in a game. These apps offer an alternative to physical joysticks, which can be cumbersome or unavailable for mobile games. This article provides an overview of virtual joystick mobile applications, including their history, development, and current uses. We discuss the pros and cons of virtual joysticks, such as the ability to customize control settings and the potential for imprecise controls. We also review the current state of mobile applications with virtual joysticks and highlight possible areas for future work, such as improving the user interface, adding support for more games and genres, integrating artificial intelligence and incorporating other technologies such as virtual or augmented technologies. Finally, we discuss the future of mobile apps with virtual joysticks and their potential impact on the gaming industry. We suggest that as mobile devices become more powerful and ubiquitous, virtual joystick applications will continue to play an important role in mobile games, and developers will need to look for new ways to improve the user experience and functionality of these applications. keywords - virtual joysticks and mobile gaming

Keywords: virtual joysticks and mobile gaming.

I. INTRODUCTION

virtual joystick is a software-based control input mechanism that replicates the functions of a physical joystick. It is often used in video games or other interactive applications where the user must control a character or object on the screen. The virtual joystick usually appears as a round or square area on the screen. A small dot or icon indicates the location of the joystick. The user can move this point or icon by sliding or touching the screen with a finger. I. INTRODUCTION This module monitors the position of the user's finger in the virtual joystick user interface and converts this movement into the corresponding movement direction in the application. Here are some basic terms that are commonly used when talking about virtual joysticks:

- Input: Refers to user motion or action detected by the virtual joystick, such as touching or swiping the screen.

- Dead zone: the area around the center of the virtual joystick where no movement is detected. It is used to prevent unintentional movement of the controlled character or object.
- Responsiveness: The degree to which the virtual joystick responds to user inputs. Greater sensitivity means that smaller movements of the virtual joystick result in larger movements in the application.
- Button Mapping: Assigning different functions or commands to specific buttons in the virtual joystick user interface.
- Haptic Feedback: Tactile or force feedback produced by a device when a virtual joystick is moved, giving the user a

physical response to input.

- Calibration: Adjusting the virtual joystick to ensure that it correctly registers user input and responds as expected.
- Axis: Different directions in which the virtual joystick can be moved, e.g. up, down, left and right.

II. Literature

Maud Marchal et al.,[1] proposed Joyman: A HumanScale Joystick for Navigating in Virtual Worlds in the year 2008 A new user interface called Joyman, designed for immersive movement in a virtual environment. While many previous user interfaces preserve or stimulate the user's proprioception, Joyman aims to preserve balance sensitivity to improve immersion during virtual movement tasks. The proposed user interface is based on the metaphor of a joystick on a human scale. The device has a simple mechanical structure that allows the user to express their virtual navigation intention by tilting accordingly. We also propose a control law inspired by the biomechanics of human movement, which converts the measured pitch angle into walking direction and velocity—that is, a virtual velocity vector. A preliminary evaluation was performed to evaluate the advantages and disadvantages of the proposed interface and to better define future expectations for such a device.

Ae-Kyoung Ko and Joon-Young Choi et al..,[3]proposed A Haptic Interface Using a Force Feedback Joystick in the year 2007 We propose a haptic interface algorithm for joystick users driving unmanned vehicles. The haptic interface algorithm is implemented with a force-assisted joystick equipped with low-cost encoder less DC motors. By generating specific forces based on the distance between the remote joystick column and obstacles, the haptic interface allows the user to perceive distance information through the sense of touch. When the joystick is not in use or there are no obstacles in the work area, we propose a starting point control algorithm that places the joystick at the starting point of the post. The algorithm prevents distant vehicles from moving incorrectly and provides the driver with a realistic force that opposes the movement of the steering column. Experimental results obtained in different scenarios show the validity of the proposed haptic interface algorithm and descent control algorithm. Keywords: haptic user interface, force feedback joystick, proportional derivative control

David R. Loker et al., [2] proposed Joystick Control of Finch Robot the virtual joystick application in the year 2007. Junior students in the electrical and computer engineering program complete a 3-credit course in measurement and instrumentation. The course has three main parts: (1) LabVIEW application programming, (2) Data acquisition, sensors and signal processing, and (3) Design of measurement systems. Weekly laboratory activities reflect lecture material. Part of the course requirements includes an end-of-semester team design project, with one possible option being the design and implementation of Finch Robot software. Students are provided with LabVIEW Subsist for all low-level functions of the robot (beep, three-color LED, left/right motor control, light sensors, resistance detectors, temperature sensor, and three axis accelerometers) and corresponding DLL files to run the. Under YOU The goal of the project is to use my LabVIEW programming skills to design a joystick for the speed and direction of the robot, show the lift and roll of the robot, and warn the user with a sound of an obstacle in front of the robot. a robot a robot This document contains a detailed list of technical requirements for the project. An example of the students' work and the evaluation of the project are presented. Recommendations are included to help ensure student success in the project.

2.1PROPOSED SYSTEM

Here we create a single app with fully customizable controls ie; we can add multiple layouts depending on the device being controlled, making the app more compatible.

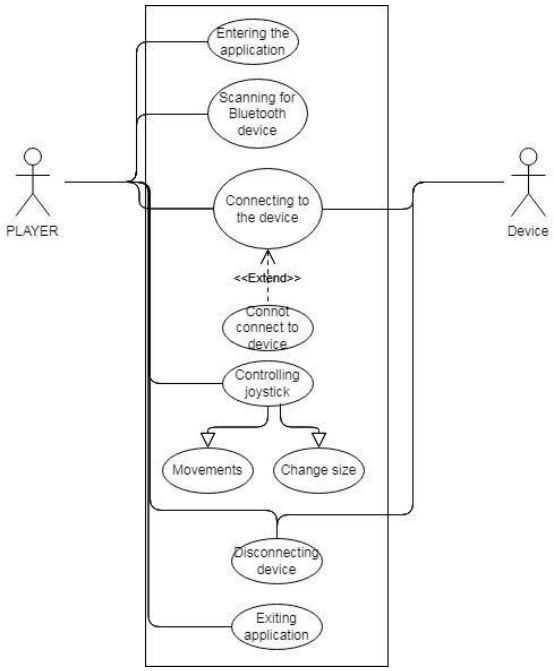
It also has a multi-touch feature that helps with gaming and controlling devices. The user interface of this program has a virtual joystick that

can be moved in any direction to control the movement of a character or object. It also has a set of buttons to perform various functions.

The sensitivity of the virtual controller can be adjusted according to the user's preferences. The sensitivity setting can vary from low to high

depending on the skill of the user. This application has a calibration function to adjust the centre of the virtual joystick. This ensures that the joystick stays centred and responds to the user's movements. Its responses combined with good haptic feedback provide an immersive gaming experience.

2.2.System Architecture



*2.3.*MODULE DESCRIPTION

There are mainly 2 modules namely

1. User module
2. Admin module

User module

The proposed system can be divided into a number of modules as shown below: 1. Touch Input Module

2. Joystick Display Module

3. Data Transmission Module

4. Settings Module

5. Networking Module

2.4.Touch Input Module

The touch input module for this proposed system involves detecting the users touch in the screen and translating that touch into a

direction or movement for the virtual joystick.

1. It detects the user's touch on the screen by monitoring touch events on the device. This can typically be done using the touch events provided by the operating system or touch event libraries.

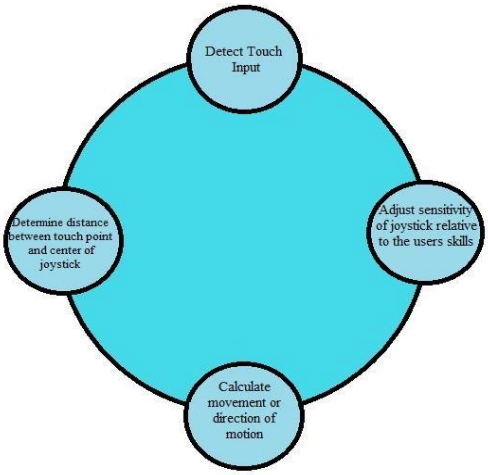
2. Determine the location of the touch relative to the centre of the virtual joystick. You can do this by calculating the distance between the touch point and the centre point of the joystick.

3. Based on the location of the touch relative to the centre of the joystick, calculate the direction or movement that should be applied to the virtual joystick. For example, if the touch is to the right of the centre, the joystick should move to the right

4. Update the position of the virtual joystick based on the calculated direction or movement. You can do this by updating the joystick's position on the screen or by sending commands to the game or application that is using the joystick.

5. Sensitivity controls are used to adjust how quickly the joystick should respond to the touch input.

6. Repeat steps 1-5 as needed to continue tracking the user's touch input and updating the virtual joystick accordingly.

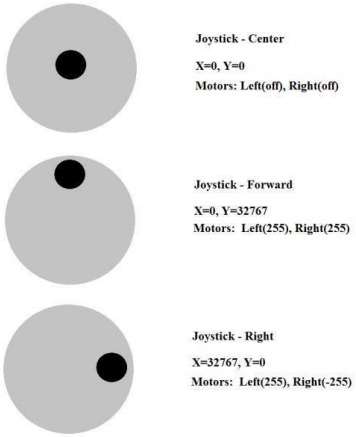


2.5 Joystick Display Module

A joystick display module for a virtual joystick would provide a visual representation of the joystick's position and movement on an application or other device. This module would typically consist of a graphical user interface (GUI) that displays a visual representation of the joystick, along with real-time updates of the joystick's position and any changes in its movement.

The user could move the joystick by dragging a virtual representation of the joystick handle on the interface, and the position of the handle would be displayed in real-time on the screen. The display module could also incorporate various visual cues to help the user better understand the position and movement of the joystick.

Overall, a joystick display module for a virtual joystick would help to enhance the user's experience by providing a clear and intuitive way to interact with the joystick and visualize its movements. This type of module could be useful in a variety of applications, such as video games, virtual reality environments, and remote-controlled devices.



2.6 Data Transmission Module

The data transmission module for a virtual joystick is be responsible for transmitting the joystick's position and movement data from the user's device to the device controlled by the virtual joystick. This module would typically be used in applications where the user is controlling a remote device, such as a robot or drone, using a virtual joystick on their mobile device.

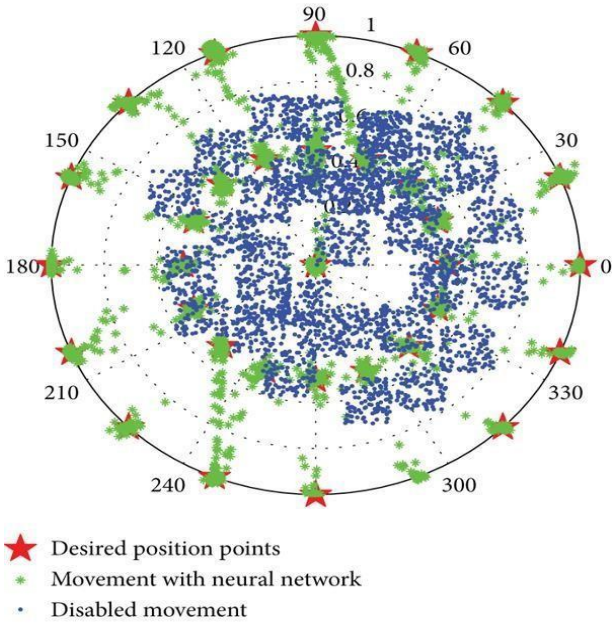
The data transmission module could be implemented in a variety of ways, depending on the specific requirements of the application. Here in this proposed system, it uses Bluetooth to connect with the device and to transmit data in real time. To ensure reliable and accurate transmission of the joystick data, the data transmission module would need to include error-checking and correction mechanisms. This could involve using checksums or other algorithms to verify the integrity of the data before it is sent, as well as providing a mechanism for retransmitting any data that is lost or corrupted during transmission.

Overall, a data transmission module is a crucial component of a virtual joystick system that enables users to control remote devices from their mobile devices. The reliability and performance of the module are critical to ensuring that the joystick movements are accurately transmitted to the remote device and that the user has a seamless and responsive control experience.

2.7 Settings Module

The settings module for a virtual joystick would typically contain userdefined layouts and configurations that control the behaviour of the virtual joystick.

The specific parameters and configurations will depend on the specific requirements of the virtual joystick application



2.8 Networking Module

A networking module in a virtual joystick system would allow users to connect and communicate with other devices or systems that are involved in the control of the joystick.
This module would be responsible for establishing and maintaining the network connections, transmitting data between devices, and managing any network-related issues that may arise.

The networking module could be implemented using a variety of networking protocols, depending on the specific requirements of the application. One important consideration when designing a networking module for a virtual joystick system is latency. Since even slight delays in transmission can result in significant lag in the control of the joystick, it is important to minimize latency as much as possible. This could involve implementing various optimizations such as packet compression, caching, or buffering.

Another important consideration is security. Depending on the nature of the application, it may be important to ensure that the joystick data is transmitted securely and that unauthorized access to the system is prevented. This could involve implementing encryption, access control mechanisms, or other security measures. Overall, a networking module is a critical component of a virtual joystick system that enables users to control remote devices over a network

The reliability, performance, and security of the networking module are critical to ensuring that the joystick data is transmitted accurately and that the user has a seamless and responsive control experience.

2.9 Admin module

The admin module for a virtual joystick app would typically include functions for managing the app's settings, user accounts, and other aspects of the app's operation. Here are some possible features and functionalities that could be included in an admin module for a virtual joystick app:

1. User management: The admin can create, edit, and delete user accounts, as well as manage user permissions and roles

2. Settings management: The admin can adjust various settings for the app, such as the default joystick mode, sensitivity, and other performance related settings.

3. Data management: The admin can view and manage the app's data, including user profiles, joystick usage data, and other relevant metrics.

4. Reporting and analytics**:** The admin can generate reports and analytics based on the app's usage data, providing insights into user behaviour, performance, and other key metrics.

5. App updates and maintenance: The admin can manage app updates and perform maintenance tasks, such as server backups and system updates.

6. Security management: The admin can manage the app's security features, such as user authentication and authorization, data encryption, and other security-related settings.

7. Support and helpdesk**:** The admin can provide support and helpdesk services to users who encounter issues or need assistance with the app.

The starting of a virtual joystick application would be the lunching of the app interface. The application would typically start by initializing the necessary components, such as libraries, APIs, or other dependencies that are required for the virtual joystick to function properly.

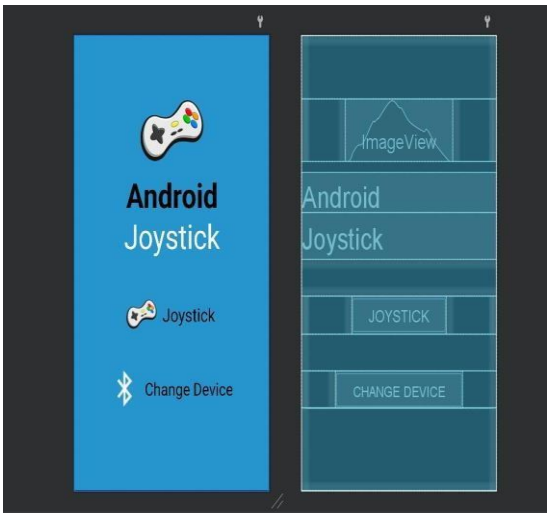


Figure 5.6 Menu.xml

The menu page of a virtual joystick application is a screen or window that provides access to various options, settings, and functionalities related to the virtual joystick. It typically serves as a central hub for managing different aspects of the virtual joystick application.

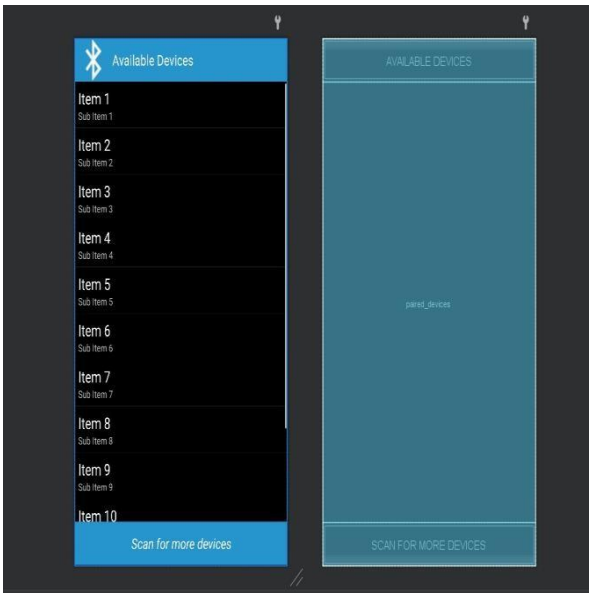


Figure 5.7 Bluetoothscan.xml

The Bluetooth scan feature of a virtual joystick application allows the application to search for and establish wireless connections with Bluetooth enabled devices, such as game controllers, smartphones, or other devices that can be used as input devices for controlling the virtual joystick.

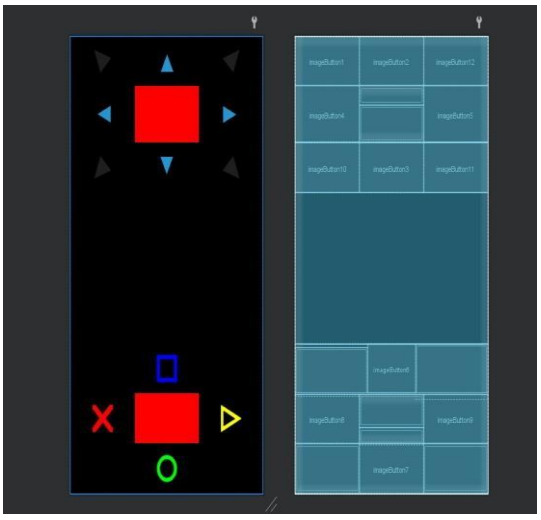


Figure 5.9 Joystick.xml

The joystick component of a virtual joystick application is the graphical representation of a joystick on the user interface (UI) of the application. It serves as the primary control element that the user interacts with to control the virtual joystick's behavior.

III.CONCLUSION

There are several potential areas for future work for mobile applications with virtual joysticks, including:

1. Improved user interface: One area of focus could be improving the virtual joystick user interface to make it more intuitive and friendly. This could include trying different control systems, adding haptic feedback or animations, or adjusting the layout and size of the users' joystick.

2. Improved Game Support: Another area of improvement could be to add support for more games and genres. Virtual joystick applications are currently most commonly used in action and adventure games, but there may be opportunities to expand into other genres such as racing, simulation or puzzle games.

3. Advanced Artificial Intelligence: Artificial Intelligence (AI) can be integrated into virtual joystick applications to provide users with advanced control capabilities. For example, the AI can automatically adjust the joystick sensitivity based on the current game situation, or predict the user's next move and adjust the joystick accordingly.

4. Multiplayer support: Virtual joystick apps can also add multiplayer support, allowing players to compete against each other or cooperate in games that require precise control.

5. Integration with other technologies: Virtual joystick applications can be integrated with other technologies, such as virtual or augmented reality, to provide a more immersive and immersive gaming experience.

REFERENCES

[1] Maud Marchal, Joyman "A Human-Scale Joystick for Navigating in Virtual Worlds" ,2008

[2] David R. Loker "Joystick Control of Finch Robot the virtual joystick" application, 2007.

[3] Ae-Kyoung Ko and Joon-Young Choi "A Haptic Interface Using a Force- Feedback Joystick" ,2007

[4] Android Developers documentation on creating a custom view for virtual joystick: https://developer.android.com/training/customviews/create-view

Code Project tutorial on creating virtual joystick for Android games: *https://www.codeproject.com/Articles/1088652/I mplementing-Virtual- Joystick-Android-Application*

[5] Medium article on creating a virtual joystick for Android games using Kotlin: *https://medium.com/swlh/create-virtualjoystick-for-android-game-with- kotlin-6a8e6e62f6c7*

[6] GitHub repository with open-source virtual joystick implementation for Android: *https://github.com/control9/android-joystick*