Text

Description automatically generated with medium confidence

ARI2201

Individual Assigned Practical Task

Course Assignment 2022/2023

**Play Games via Gestures**

By Mark Dingli

mark.dingli.21@um.edu.mt

20703H

**Table of contents**

Statement of Completion ………………………….………………………………… 3

Introduction …………….………………………….………………………………… 4

Task 1 - Table-driven Lexer ……………………….………………………………… 5

Task 2 - Hand-crafted LL(k) parser ….…………….………………………………… 8

Task 3 - AST XML Generation Pass ...…………….……………………………….. 11

Task 4 - Semantic Analysis Pass ………………….………………...……………… 14

Task 5 - PixIR Code Generation Pass …………….………………………………… 16

Conclusion …………….………………………….………………………………… 19

Video .……………...….…………………………..………………………………… 20

References …………….…………………………..………………………………… 21

1. **Introduction**

Gaming is evolving at an unprecedented pace. As it continues to captivate a growing audience, the need for intuitive, accessible, and engaging games is greater than ever. One emerging frontier in this regard is gesture-based control, which offers a novel way of interacting with technology through physical hand movements. Despite its potential, gesture-based control has been largely unexplored in certain domains, such as gaming. Traditionally, gaming has been dominated by manual and button-based controls, often limiting the immersive experience and accessibility of games.

This method, while reliable, often fails to provide a truly immersive and engaging experience for the user. The reliance on such conventional control methods can limit accessibility, particularly for individuals who might find these controls physically challenging or unintuitive. In addition, the growing desire for more interactive and innovative ways of gaming calls into question the sustainability of these traditional mechanisms.

One such game that has been a cornerstone of the gaming industry yet remains confined to traditional control mechanisms is Pacman. With its simple gameplay yet captivating mechanics, Pacman offers an excellent canvas for exploring the potential of gesture-based control. However, integrating such a control system into an existing game poses a significant challenge. It requires not only an understanding of web development technologies but also the ability to leverage the power of machine learning in a creative and effective manner.

This project set out to tackle these challenges. It aimed to adapt the classic game of Pacman, integrating gesture-based controls in place of the traditional manual controls. The goal was to create a more immersive, interactive, and accessible gaming experience while exploring what is possible with current web development technologies and machine learning tools. By integrating more natural and intuitive control methods, it is possible to significantly enhance the user experience and open up gaming to a broader audience. This project thus acknowledges and addresses this gap, aiming to pioneer a shift in gaming interaction through the integration of gesture-based controls.

The report that follows will detail the journey of this project, from the initial stages of conceptualizing the idea, to the selection and utilization of specific tools and technologies, to the eventual implementation and testing of the final product. The hope is that it will not only provide an insight into the process and challenges of such an undertaking but also inspire further innovation in this exciting field.

1. **Literature Review**

Gesture-based control in gaming is a relatively new and unexplored area, with limited research available. However, there are several studies and publications that provide insights into the potential and challenges of integrating gesture-based controls into games, as well as the impact it can have on the user experience and accessibility.

One key area of research in gesture-based control is human-computer interaction (HCI). HCI studies focus on understanding how users interact with technology and how to design interfaces that are intuitive, efficient, and enjoyable. Research in HCI has explored various input modalities, including gesture recognition, and has highlighted the advantages of natural and intuitive interaction techniques in improving user engagement and immersion in gaming experiences [1].

In the realm of gaming, there have been some notable examples of gesture-based control implementations. For instance, researchers have investigated the use of motion sensors, such as Microsoft Kinect, to enable gesture-based control in games. Studies have demonstrated the potential of using gesture recognition to enhance gameplay and create more immersive experiences [2].

Machine learning techniques play a significant role in gesture recognition and tracking. Deep learning algorithms, in particular, have shown promise in accurately recognizing and interpreting gestures from sensor data. These algorithms have been used in various domains, including sign language recognition and human pose estimation [3]. Applying machine learning to gesture-based control in gaming allows for the creation of personalized gesture classifiers that can adapt to individual users, enhancing the precision and reliability of the control system.

Accessibility is a critical consideration when exploring gesture-based control in gaming. Traditional control mechanisms often present challenges for individuals with disabilities or limited mobility, excluding them from fully engaging in gaming experiences. Gesture-based control has the potential to provide a more inclusive gaming environment by accommodating various physical abilities and preferences. Research in accessible gaming emphasizes the significance of considering diverse user needs and implementing alternative input methods, such as gesture recognition, to ensure equal accessibility [4].

In conclusion, the existing literature on gesture-based control in gaming demonstrates its potential to revolutionize the way users interact with games. The studies conducted in the field of human-computer interaction, machine learning, and accessible gaming provide valuable insights into the benefits and challenges associated with gesture-based control. This literature review highlights the need for further research and innovation in this area to refine the integration of gesture-based control into gaming experiences, ensuring enhanced user immersion, accessibility and engagement.

1. **Design of the Solution**

The design of the solution for integrating gesture-based controls into the Pacman game follows a systematic approach that aims to enhance the user experience and accessibility. The solution consists of several key components, including the input module, gesture recognition module, game control module, and user interface module. Each component plays a crucial role in enabling seamless and intuitive interaction between the user and the game.

The input module serves as the interface between the user and the game. It captures the user's hand movements using the camera. These movements are then processed and passed on to the gesture recognition module for further analysis. The input module ensures the accurate and real-time detection of the user's gestures, forming the foundation for the gesture-based control system.

The gesture recognition module is responsible for interpreting the user's hand movements and translating them into specific commands for the game. This module utilizes machine learning techniques, specifically deep learning algorithms, to train a gesture classifier. The classifier is trained using a dataset of labeled gesture samples, allowing it to recognize and classify different gestures made by the user. By leveraging the power of machine learning, the gesture recognition module achieves high accuracy in detecting and interpreting user gestures.

The game control module receives the recognized gestures from the gesture recognition module and maps them to corresponding actions within the game. It controls the movement and behaviour of the Pacman character based on the user's gestures. For example, a swipe gesture to the right can trigger the Pacman character to move in that direction, while a tap gesture can make Pacman perform a specific action, such as eating pellets or avoiding ghosts. The game control module ensures a seamless integration of gesture-based controls into the game mechanics, enhancing the user's immersion and interactivity.

The user interface module provides the visual representation of the game and allows users to interact with it using gestures. It displays the game elements, such as the Pacman character, ghosts, and the maze. Users can perform gestures, such as swipes, taps, or hand movements, to control the Pacman character and navigate through the maze. The user interface module ensures a visually appealing and intuitive interface that aligns with the gesture-based control system.

By combining these components, the solution successfully integrates gesture-based controls into the Pacman game, creating a more immersive and interactive gaming experience.

1. **AI Techniques Used**

This project takes a novel approach to game interaction. By using a combination of HTML, CSS, and JavaScript, alongside the powerful p5.js library, a web-based adaptation of Pacman that moves beyond traditional control mechanisms was developed. It replaces these control mechanisms with a more immersive, intuitive method of interaction: gestures. The p5.js library facilitates the creation of interactive graphics and animations, making it an ideal tool for building the visual aspects of the game

The real magic behind this project lies in its integration of Google's Teachable Machine. This innovative tool provides a user-friendly platform for the creation of machine learning models, removing the need for complex coding or deep understanding of machine learning algorithms. By leveraging Teachable Machine's capabilities, a straightforward, user-specific classifier that recognizes specific hand gestures from the user was trained. These gestures are then mapped to corresponding actions within the game, creating a unique and interactive gaming experience.

In adapting Pacman’s game code to include gesture-based controls, the project not only extended its functionality but also opened up new avenues for user interaction. This adaptation demonstrates the potential of combining web development technologies with machine learning tools, creating applications that are both sophisticated and accessible to a wide audience. The process of incorporating these changes posed unique challenges, requiring innovative solutions that further underline the complexity and potential of this field

This project serves to highlight the versatility and power of Google's Teachable Machine and the p5.js library. By incorporating these tools into a gaming application, the project showcases how they can be used beyond their typical educational or experimental use cases. This application stands as a testament to the versatility of machine learning and the power of creative coding, underlining the possibility of their widespread application.

1. **Implementation**
2. **Evaluation & Results Obtained**
3. **Analysis of the results**
4. **Conclusion**

**References**

[1] K. W. Siovi, C. W. Kipruto, and A. Mindila, “Design thinking for gesture-based human computer interactions,” International Journal of Computer Sciences and Engineering, vol. 7, no. 3, pp. 919–925, 2019. doi:10.26438/ijcse/v7i3.919925

[2] D. M. Shafer, C. P. Carbonara, and L. Popova, “Spatial presence and perceived reality as predictors of motion-based video game enjoyment,” *Presence: Teleoperators and Virtual Environments*, vol. 20, no. 6, pp. 591–619, Dec. 2011. doi:10.1162/pres\_a\_00084

[3] B. Xie, B. Li, and A. Harland, “Movement and gesture recognition using Deep Learning and wearable-sensor technology,” Proceedings of the 2018 International Conference on Artificial Intelligence and Pattern Recognition, Aug. 2018. doi:10.1145/3268866.3268890

[4] S. Vickers, H. Istance, and M. J. Heron, “Accessible gaming for people with physical and cognitive disabilities,” CHI ’13 Extended Abstracts on Human Factors in Computing Systems, Apr. 2013. doi:10.1145/2468356.2468361

[5]

[6]

[7]

[8]

[9]

[10]