An AI Powered AR Game Platform for Hand-Drawn Sketch Recognition

by

JAGAN K K (STM22CS027)

JIBIN P V (STM22CS028)

MUHAMMED FAYIZ V C (STM22CS036)

RHUTHOSHIKA K (STM22CS047)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ST. THOMAS COLLEGE OF ENGINEERING AND TECHNOLOGY

MATTANNUR

Abstract

Augmented Reality (AR) systems like Google's Quick, Draw! and Microsoft's Azure Spatial Anchors have advanced in using computer vision and AI for interactive ap-plications. However, these platforms face challenges in recognizing user drawings accurately due to variations in sketch styles, lighting conditions, and inconsistent rendering. In addition, they often have steep learning curves that impact user experience and accessibility. The main challenges of current systems include variability in user sketches, poor performance under different lighting, and non-intuitive interfaces, leading to frustrations like delayed responses and inaccurate recognition. These issues make existing systems less effective, especially when dealing with complex drawings in real time. A comprehensive system is proposed that combines feature extraction based on deep learning, AR frame detection, AR rendering, game physics, and real-time processing. It also integrates mobile AR control synchronization and AI-powered hand gesture recognition. This platform enables users to play games like multiple block games, and snake games from hand-drawn images, providing a seamless real-time experience by combining computer vision, deep learning, and AR.

Chapter 1

Introduction

This AI-powered Augmented Reality (AR) game platform leverages computer vision, deep learning, and AR technologies to recognize and process hand-drawn sketches in real time. The system enables users to interact with digital content by simply drawing on physical surfaces, allowing them to play multiple block-based games directly from hand-drawn images.

Using advanced deep learning models, the platform accurately interprets and converts sketches into game elements, ensuring an intuitive and interactive user experience. The integration of ARKit (for iOS) and ARCore (for Android) allows seamless overlay of virtual elements onto the real world, providing an immersive gaming experience across multiple devices. This cross-platform compatibility ensures a broad reach, making the platform accessible to a wide range of users.

By combining real-time sketch recognition with AR-enhanced gaming, this system represents a significant innovation in AI-driven interactive entertainment, enabling users to engage with digital content in a more natural and creative way.

1.1 Motivation

This project is our opportunity to bring something truly unique to life - blending the creativity of hand-drawn art with the cutting-edge world of augmented reality gaming. Not only will we be able to see our sketches transform into interactive elements, but we will also gain valuable skills in AR development in gaming that are highly sought after. Each step, from the first sketch to the final game mechanics, is a chance to push our limits and showcase our ability to combine art with technology. This is our moment to create something innovative and impressive.

1.2 Problem definition

Existing systems struggle to accurately recognize complex and hand-drawn game pat-terns. These systems often fail in varying lighting and environmental conditions. Current systems support only one game at a time, lacking multigame functionality.They lack immersive features such as real-time physics and gesture controls, which reduces user engagement.

1.3 Organization of the report

The report shows an overview of AI Powered AR Game Platform for Hand-Drawn Sketch Recognition from architecture to working model. Chapter 1 discusses the designed system, motivation for our project, problems in existing systems. Chapter 2 includes the literature surveys. Chapter 3 evolves with requirement analysis. Chapter 4 outlines the design of the system. In Chapter 5 the framework of the system is provided. Chapter 6 offers insights collected from implementation and testing phases. Outcomes are analyzed and discussed in Chapter 7. Chapter 8 summarizes about the real time gaming experience and offers suggestions for future research works. Lastly, Chapter 9 provides a list of references used for verification.

Chapter 2

Literature Review

There are several ways to overlay AR inorder to play games.The paper suggest the overlay of AR and AI that allows user to play multiple games. It includes Maze game, Snake game and Tetris. The Snake game is implemented using grid detection as well as hand gestures is also applied.

2.1 SINS AR: An Efficient Smart Indoor Navigation System Based on Augmented Reality

The SINS AR (Smart Indoor Navigation System) is a cutting-edge solution that enhances indoor navigation through Augmented Reality (AR) and pathfinding algorithms. The system utilizes Theta* and A* algorithms to compute the shortest and most efficient path within a structured indoor environment. The Theta* algorithm improves upon traditional A* by considering diagonal shortcuts, reducing the overall path length and making movement more natural.

To begin the navigation process, the system employs QR code-based initialization, where users scan a QR code that marks their starting position. Once initialized, a grid-based spatial mapping system assists in object localization and navigation guidance. This structured approach divides the indoor space into a grid, allowing for precise positioning of objects, obstacles, and paths.

A key feature of the system is the real-time AR overlay, which enhances the user experience by providing visual navigation cues. These overlays dynamically display arrowheads and directional markers, guiding users toward their destination with seamless updates based on their movement. This intuitive method of navigation is particularly useful in complex indoor environments such as airports, shopping malls, hospitals, and corporate offices, where conventional signage may be insufficient or difficult to follow.

The SINS AR system enhances accessibility, efficiency, and user convenience, making indoor navigation faster, smarter, and more interactive. It offers potential applications in smart buildings, healthcare facilities, and public infrastructure, improving navigation experiences for a broad range of users, including visitors, employees, and people with disabilities.

2.2 Object Detection With Augmented Reality

This study explores the fusion of Object Detection and Augmented Reality (AR) to create a seamless real-time object recognition and interaction system. The system employs deep learning-based object detection models, including YOLO (You Only Look Once), Faster R-CNN (Region-based Convolutional Neural Networks), and SqueezeNet, each offering unique advantages for different computational environments. YOLO (You Only Look Once) is known for its real-time speed and accuracy, making it ideal for high-speed applications such as live AR interactions. Faster R-CNN, a two-stage detector, enhances detection precision by identifying regions of interest before classifying objects, making it well-suited for high-accuracy applications. SqueezeNet is optimized for mobile and lightweight AR devices, reducing computational overhead while maintaining reasonable accuracy. To optimize the object detection process, the system hybridizes YOLO and Faster R-CNN, leveraging YOLO's speed for quick initial detection and Faster R-CNN's accuracy for refining the output. This results in a system that detects objects in real-time while maintaining high precision.

One of the most critical aspects of the project is post-processing for AR alignment. This step ensures that digital overlays correctly align with real-world objects, preventing misplacement or distortion of AR elements. Accurate object localization and alignment techniques are implemented to maintain the integrity of the AR experience.

This object detection and AR integration finds applications in smart retail, interactive learning, industrial automation, and real-time information overlays. For instance, in smart shopping applications, the system can identify products on store shelves and provide instant AR-based product details. In education, AR can overlay labels on historical artifacts or biological specimens, enriching the learning experience. The system's adaptability also allows it to be deployed in manufacturing, healthcare, and security applications, demonstrating its versatility and impact.

2.3 Augmented Reality-based Sudoku Solver with Training Module to Improve Cognitive Skills

The AR-Based Sudoku Solver is an AI-powered gaming system designed to enhance cog-nitive skills and puzzle-solving abilities using Artificial Intelligence (AI) and Augmented Reality (AR). The system provides an interactive web-based interface where users can log in, select difficulty levels, and access hint mechanisms for real-time Sudoku solving.

A core feature of the system is its AI-driven Sudoku solving algorithms, which enable automated puzzle-solving with high efficiency. The system incorporates multiple algorithmic approaches:

Backtracking Algorithm – A brute-force method that systematically explores all possible solutions until it finds a valid Sudoku arrangement. Dancing Links Algorithm – A variation of Algorithm X that efficiently finds solutions to constraint satisfaction problems, such as Sudoku. Genetic Algorithm – A heuristic-based approach that mimics natural selection, evolving multiple Sudoku solutions over iterations to optimize results. Constraint Propagation – A logical deduction method that reduces the problem space by eliminating impossible values early in the solving process.

The AR integration enhances the gaming experience by allowing users to interact with Sudoku puzzles in real-time. The system recognizes handwritten Sudoku grids and overlays an interactive solution in an augmented environment. This feature is powered by Optical Character Recognition (OCR), using a CNN-MNIST model to extract handwritten digits from the user's sketch and convert them into digital Sudoku grids.

Users can scan Sudoku puzzles from books, newspapers, or sketches, and the system automatically detects, analyzes, and solves them. Additionally, the interactive AR interface provides step-by-step guidance, allowing users to learn Sudoku-solving techniques while engaging in the game.

The AR-based Sudoku Solver serves as an innovative educational tool, making Sudoku solving more engaging, accessible, and interactive. It finds applications in puzzle-solving platforms, brain-training apps, and AI-assisted education, catering to users of all skill levels. The system promotes logical reasoning, problem-solving, and mental agility, reinforcing its value as a next-generation AI-powered cognitive gaming experience.

Table 2.1: Literature Review

Method	Description	Author Details
SINS AR: Smart Indoor Navigation System	SINS AR navigates within a facility using Theta* and A* algorithms for shortest pathfinding. A QR code marks the start, while a grid-based system locates objects and guides characters. AR overlays an arrowhead pointing to the destination.	Yasmin Alkady,Rawy Rizk,Deema Mohammed Alsekait, Diaa Salama Abdelmi-naam
Object Detection With Augmented Reality	Object detection overlays real-time digital data using YOLO, Faster R-CNN, and SqueezeNet. The YOLO-Faster R-CNN hybrid boosts speed and precision, while SqueezeNet optimizes for mobile devices. Post-processing ensures AR alignment.	Arnav Tyagi, Dev Rajpal, Jagendra Singh, Hardeo Kumar Thakur, Kamal Upreti
AR-based Sudoku Solver	AR Sudoku resolver enhances cognitive skills via a web app with login, modules, difficulty levels, and hints, using Backtracking, Dancing Links, Genetic Algorithm, and Constraint Propagation. CNN-MNIST-model.h5 handles OCR for digit extraction.	Abhinav Durga, Sohanraj R, Sparsh Agarwal, Sanket Salvi, Mrunali Borkar

Chapter 3

System Analysis

The system analysis of the AI-based AR gaming platform focuses on understanding the requirements, components, and interactions necessary for real-time sketch recognition and AR integration. The system captures user-drawn sketches, processes them using a deep learning model (CNN-based, such as YOLO), and maps them to corresponding 3D game elements. The AI model must be trained with a diverse dataset to accurately recognize hand-drawn inputs and generate interactive game objects. The integration of AR technology (ARKit/ARCore) ensures seamless rendering of recognized sketches into the physical environment, allowing users to interact with them in real-time through gestures and voice commands. The system must support multiple game modes, requiring efficient procedural content generation and physics-based game mechanics. Additionally, the web-based interface must provide smooth communication between the AI model and the user, ensuring low latency and high responsiveness. Performance evaluation includes measuring frame rates, tracking accuracy, and rendering quality to optimize real-time gameplay. The system also requires ongoing dataset expansion, model retraining, and game logic updates to improve recognition accuracy and enhance user experience. Ensuring scalability, accessibility, and seamless integration of multiple game codes into a single platform is essential for delivering a dynamic, immersive, and interactive gaming experience.

3.1 Requirement Analysis

The AI-based AR gaming platform requires a combination of hardware, software, AI models, and user interface components to enable real-time sketch recognition and inter-active gameplay. The system includes a sketch recognition module using deep learning (CNN models like YOLO) to identify hand-drawn inputs and render them as AR objects. It supports gesture-based controls, voice commands, and multiplayer modes for enhanced interaction. The platform ensures smooth rendering and scalability to accommodate additional game modes.High-resolution cameras and GPU/cloud processing enable real-time inference. The software stack includes OpenCV for AI processing and game development. This system creates an engaging AR gaming experience by merging AI-driven sketch recognition with immersive, real-world interactions.

3.1.1 Functional Requirement

Hand-drawn sketch recognition for seamless interaction between players and the virtual environment, allowing users to create or manipulate game elements through their own sketches, enhancing the immersive experience. AR Overlay and Rendering for Integrating hand-drawn sketches into the real world. Game Interaction that enables players to directly influence the game world through their sketches. Real-time processing for instant analyzing and interpreting of sketches. Gesture Contol Integration that allows players to manipulate and control the game environment using natural hand movements, creating a more intuitive and immersive experience. Game Physics and Logic that ensure that virtual objects created through sketches behave realistically.

3.1.2 Non Functional Requirement

The system must ensure real-time sketch recognition and AR rendering with minimal latency, maintaining smooth gameplay. It should accurately identify sketches under varying conditions, including different lighting, sketch quality, and user input styles. The platform must support thousands of concurrent users with scalable resources to handle high traffic periods seamlessly. Security is essential, with end-to-end encryption and industry-standard practices protecting user data from unauthorized access. An intuitive and engaging user interface should allow players of all ages and skill levels to interact with the AR environment effortlessly. Reliability is crucial to minimize service disruptions and maintain a consistently smooth experience. The system must be compatible across a wide range of devices, including laptops, tablets, and smartphones, supporting different OS versions. Maintainability is also key, ensuring easy updates, bug fixes, and feature enhancements for long-term sustainability with minimal downtime.

3.1.3 Other Requirement

Hardware requirements

The laptop acts as the primary device for running AI models that process and recognize hand-drawn sketches, providing the computational power needed for training, inferenc-ing, and executing machine learning models like CNN. The webcam captures real-time images of hand-drawn sketches on paper or a drawing surface, which are then processed and recognized by the AI model to interact with the AR game. It also tracks user gestures, such as swipes or finger movements, enhancing interaction with the system. Various hand sketches, including maze and square grids, are created for playing Maze, Tetris and Snake games.

Software requirements

Python is used develop a deep learning model (YOLO) to recognize and interpret hand-drawn sketches for games. OpenCV is employed to process images of these sketches, detecting edges, contours, and shapes, which serves as a pre-processing step for the AI model to recognize the drawing. HTML is used to structure static web pages and, when combined with JavaScript, enables the creation of the game's front-end by handling user input, providing sketching tools, and displaying real-time feedback from the AI model. While HTML sets up elements like buttons and canvases, JavaScript manages dynamic behavior, handling user interactions and updating the interface with AI feedback. YOLO, a powerful object detection model, is utilized to detect specific objects in the hand-drawn sketches, enhancing the game's recognition capabilities.

3.2 Feasibility Study

Technical feasibility assesses whether the hardware and software components required for a project are suitable, accessible, and compatible, ensuring efficient functionality without performance bottlenecks or integration issues. The hardware evaluation considers devices like laptops, webcams, and AR headsets, which must be powerful enough to handle real-time sketch recognition, augmented reality rendering, and AI-based tasks such as YOLO object detection. These processes require high-performance CPUs, GPUs, or specialized AR hardware to ensure smooth execution. Software compatibility is equally crucial, as AI models like YOLO must run efficiently on the chosen hardware, supported by cross-platform frameworks like MediaPipe, which provides optimized solutions for real-time computer vision tasks. Additionally, AR development tools such as AR.js or Unity with AR Foundation must integrate seamlessly with the existing system architecture to avoid performance conflicts. Beyond hardware and software, integration is essential to ensure smooth communication between AI-based object detection, AR rendering, and the user interface. The system must function cohesively, interpreting sketches accurately and providing real-time feedback while avoiding compatibility issues like mismatched data formats or unsupported APIs. By addressing these factors, technical feasibility ensures the project can be implemented effectively without hardware limitations, software incompatibilities, or integration challenges.

3.2.1 Operational Feasibility

Operational feasibility evaluates how well the system can be integrated into the existing infrastructure and overall workflow. It involves ensuring seamless integration with exist-ing systems, such as security frameworks and network protocols, to prevent disruptions. For instance, if the platform includes multiplayer support, it must be compatible with existing game servers or online systems without requiring significant modifications. Additionally, support for security infrastructure is crucial to prevent vulnerabilities, ensuring that the AI-powered AR game platform secures communication channels and protects sensitive user data, such as sketches, from unauthorized access. Implementing robust encryption, access controls, and compliance with security standards strengthens the system's reliability. Another key factor is user adoption and training, which determines how easily administrators and users can adapt to the platform. A user-friendly interface, intuitive controls, and proper documentation help minimize the learning curve, ensuring that existing staff can manage and maintain the system without extensive training or effort. By addressing these aspects, operational feasibility ensures a smooth transition, security compliance, and effective usability of the AI-powered AR system.

3.2.2 Economic Feasibility

Economic feasibility focuses on the financial aspects of the project, including the budget required for both procurement and maintenance of hardware, software, and infrastructure. It involves evaluating initial procurement costs, which include expenses for acquiring necessary hardware such as laptops, webcams, and AR devices, as well as software licenses for AI frameworks and AR development tools. Selecting cost-effective yet reliable equipment that meets performance standards is essential to optimize expenses without compromising efficiency. Additionally, ongoing maintenance costs must be considered, covering long-term operational expenses such as software updates, AI model training, hardware upgrades, security measures, and technical support. Ensuring sustainable financial planning helps maintain system performance while minimizing unexpected costs and ensuring long-term viability.

3.2.3 Performance Feasibility

Accuracy and detection speed are critical factors in evaluating the performance of the AI model for recognizing hand-drawn sketches. Accuracy refers to the model's ability to correctly identify a wide range of sketches under different conditions, including variations in drawing styles, lighting, and angles. High accuracy is essential to ensure a seamless interactive experience and maintain user satisfaction. Detection speed measures how quickly the AI model can process and recognize sketches in real time. For an AR game, rapid response to user inputs is crucial to delivering a fluid and engaging experience without noticeable delays. Optimizing both accuracy and speed enhances the overall usability and effectiveness of the system.

Chapter 4

System Design and Schedule

The AI-powered AR game platform integrates computer vision and deep learning techniques to recognize hand-drawn sketches and interact with them in real-time. Utilizing CNN-based models and custom feature extraction algorithms, the system identifies user-drawn inputs and converts them into interactive game elements. The platform provides an intuitive AR interface, enabling seamless user interaction, game physics integration, and real-time rendering. The development schedule spans 10 weeks, covering dataset collection, model training, AR integration, testing, and UI/UX refinement. The project will be executed over a 13-week schedule, covering key phases essential for successful completion. The initial weeks focus on topic selection and an in-depth literature survey to establish a strong foundation. This is followed by system design, where the architecture and framework are planned.

The implementation phase, spanning four weeks, integrates algorithms and key func-tionalities for the AI-based AR system. Once the core system is developed, testing and integration ensure accuracy and performance across different environments. The final phase involves report generation and paper publication, documenting findings, results, and improvements for future work.

4.1 System Architecture

The AI-based AR gaming system is an advanced platform that merges artificial intelli-gence and augmented reality to create an engaging, interactive gaming experience. It allows users to draw sketches on physical or digital surfaces, which are then recognized, processed, and transformed into interactive game elements. The system is designed with multiple layers to ensure seamless execution, real-time responsiveness, and continuous improvements.

The User Interaction Layer serves as the primary interface between the player and the system, enabling users to draw sketches, which are captured through a camera or webcam. This layer also includes input recognition, where the system detects user gestures and interactions, ensuring a dynamic and immersive gaming experience. Once the sketches are captured, they are passed to the AI-AR Processing Layer, which plays a critical role in sketch recognition, game element generation, and augmented reality integration. This layer begins with preprocessing, where noise is removed, and key features of the sketch are extracted. The AI-powered recognition model then analyzes the sketch, determines the intended object or pattern, and overlays interactive AR elements to enhance visualization. The AR system ensures that game elements appear naturally integrated into the real-world environment, making the experience more immersive. Recognized sketches are converted into interactive objects, such as characters, obstacles, or game controls, which users can manipulate in real time.

To ensure long-term efficiency and adaptability, the System Training and Maintenance Layer continuously improves the AI model and gaming functionalities. The system collects newly drawn sketches and incorporates them into its dataset, refining the AI model for more accurate recognition over time. Administrators can update game mechanics, add new gameplay features, and introduce additional sketch-based controls to enhance the gaming experience. This iterative learning approach allows the system to evolve, recognizing a wider variety of sketches with improved accuracy.

The overall workflow of the system follows a structured process. A user begins by drawing a sketch, which is captured through a camera and sent for preprocessing. The AI model analyzes the sketch and applies AR overlays to generate a corresponding game element. The user then interacts with the game, using either touch gestures, voice commands, or external controllers, while real-time feedback ensures an engaging experience. The AI continuously learns from user input, improving recognition accuracy and enhancing game adaptability.

The system also supports multi-game functionality, allowing different game types such as snake game, maze games and action-based interactions to be integrated within a single platform. By leveraging real-time AI inference and AR visualization, the platform enhances user engagement, making gaming more interactive and personalized. Future improvements may include enhanced gesture recognition, voice-controlled interactions, cloud-based AI processing for faster recognition, and cross-platform compatibility to support multiple devices, including AR glasses and mobile interfaces. By combining artificial intelligence, augmented reality, and intuitive sketch-based interaction, the AI-based AR gaming system offers a unique blend of creativity, technology, and immersive entertainment.

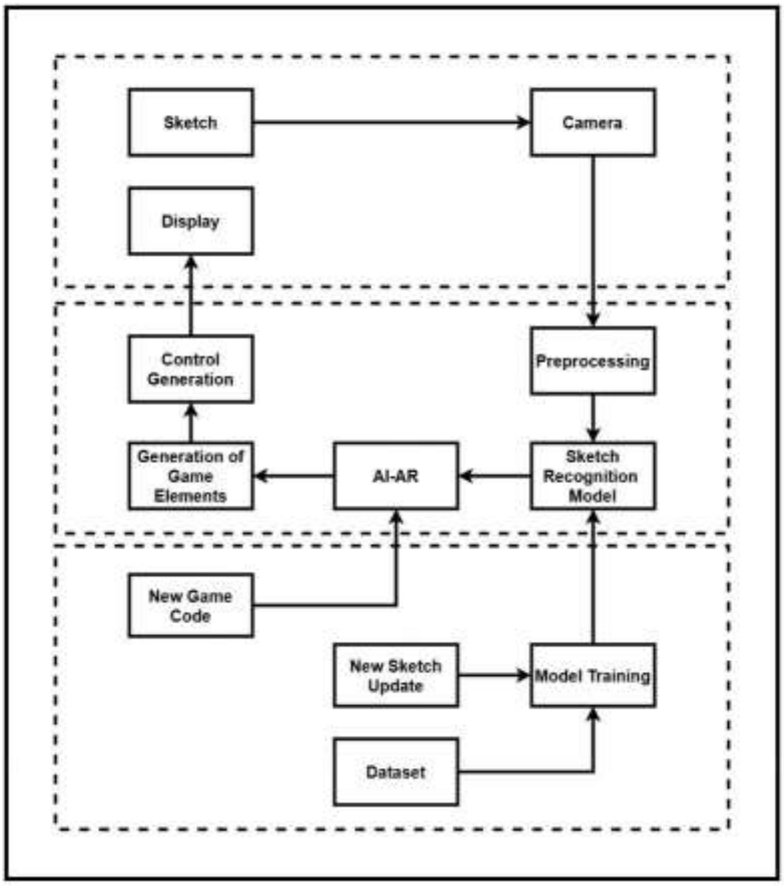

Figure 4.1: Architecture Diagram

4.2 Use Case Diagram

The process begins when the user sketches a shape or object, serving as the primary input for the AI-based AR system. The system captures this sketch using a camera, ensuring that even hand-drawn sketches on physical surfaces can be processed digitally. To improve recognition accuracy and adaptability, the admin continuously updates and modifies the dataset by refining existing sketches and adding variations. These updates help the system recognize diverse sketching styles, angles, and levels of detail, ensuring that even incomplete or roughly drawn inputs can be interpreted correctly. Regular dataset expansion and refinement allow the AI model to evolve, improving recognition capabilities and ensuring a seamless user experience.

Once the dataset is updated, the AI model undergoes training using the new sketches. The training process involves feeding the AI with labeled data, enabling it to learn patterns, identify unique features, and enhance its ability to recognize a wider range of user inputs. When a user submits a sketch, the system processes it through multiple recognition algorithms that analyze its structure, compare it to the trained dataset, and classify the drawn object. This step is crucial for accurate interpretation, as the system must distinguish between similar-looking objects and correctly map the sketch to its intended representation. The efficiency of this classification ensures smooth and responsive interaction between the user and the AR environment.

Once the system successfully recognizes the sketch, it proceeds to generate an AR overlay, visually enhancing the drawing with interactive elements. This overlay transforms the static sketch into a dynamic digital object that integrates seamlessly with the real-world environment. The system then creates an appropriate game scenario based on the recognized sketch, mapping the input into an engaging gameplay experience. For instance, a simple maze sketch might be converted into a playable puzzle, while a snake-like drawing could initiate a classic snake game. The AI ensures that the generated game elements align with the user's sketch, maintaining consistency and accuracy.

During gameplay, the user interacts with the AR-generated elements dynamically, using gestures, voice commands, or touch-based controls. The system continuously tracks user input and provides real-time feedback, ensuring a smooth, immersive experience. AI- driven adaptive mechanics adjust difficulty levels and interactions based on user behavior, making the gameplay more engaging. The system also allows for modifications during play, such as redrawing objects or repositioning game elements, further enhancing the interactive experience.

Upon completing the game session, the user has the option to save progress, modify the sketch for future interactions, or exit the system. The AI model continues learning from user inputs, refining its recognition accuracy based on real-world interactions. This structured workflow, incorporating AI-driven learning and real-time AR integration, ensures continuous system improvement, enhances user engagement, and maintains high performance in sketch-based interactive gaming.

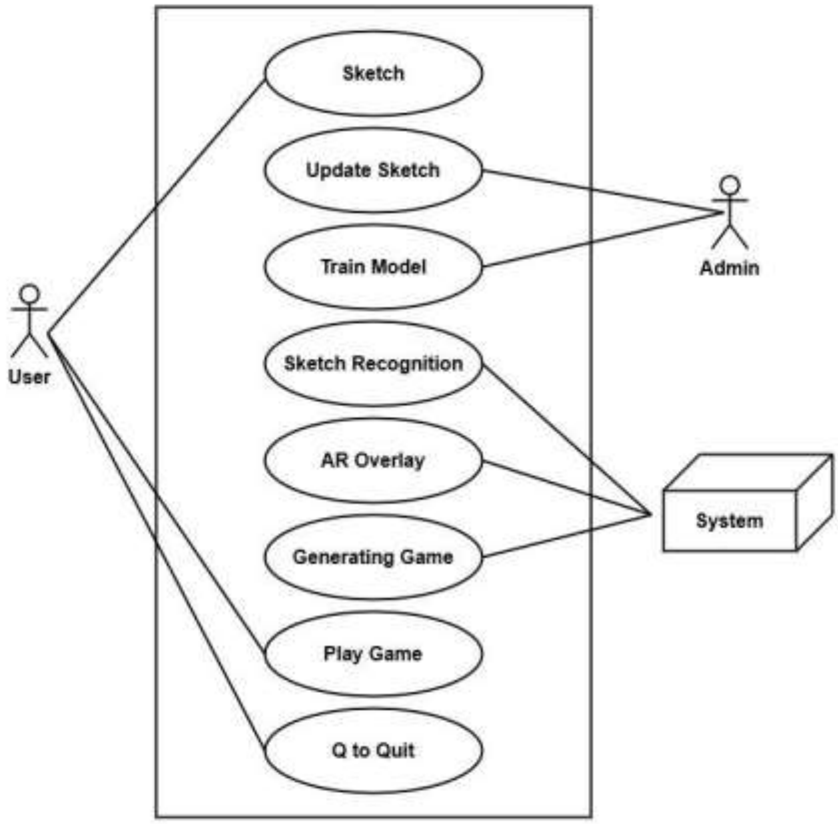

Figure 4.2: Use Case Diagram

4.3 Data Flow Diagram

The data flow diagram for the AI-based Sketch Recognition Gaming System outlines the systematic process within the system, starting with user sketches as input. The system processes these sketches using recognition algorithms and overlays an augmented reality (AR)-based game within the sketch. The Admin oversees updates and maintenance of the system to improve recognition accuracy and gaming interactions. This diagram provides a concise representation of the sequential information flow in the AI-based gaming system.

4.3.1 Level 0 Data Flow Diagram

At Level 0, the Sketch Recognition Gaming System operates as an interactive AI-driven platform where users provide sketches as input. These sketches serve as the foundation for AI-based processing, enabling the system to recognize patterns, shapes, and structures within the sketch. Once a sketch is recognized, the system overlays an Augmented Reality (AR) game within the sketched environment, allowing users to interact with it in real time. The AR overlay enhances the user’s experience by dynamically transforming their sketches into interactive game elements.

The AI continuously refines its recognition accuracy through machine learning-based updates, ensuring that the system can interpret various sketch styles and user-specific drawing patterns. Admin intervention plays a crucial role in this process, as administrators update new sketches, perform system maintenance, and manage dataset enhancements. By regularly refining the AI model, the system becomes more adaptive, improving its ability to identify sketches with higher precision.

The administrator is responsible for optimizing the system by managing model training, fine-tuning datasets, and ensuring seamless integration of new sketches. Through continuous updates and optimizations, the AI model evolves, making the gaming system more intelligent and responsive.

Overall, this structure ensures an adaptive and intelligent gaming experience where AI and human intervention work together. As the system learns from user inputs and administrator interventions, it progressively improves sketch recognition accuracy, AR overlay quality, and game responsiveness.

Figure 4.3: Level 0 Data Flow Diagram

4.3.2 Level 1 Data Flow Diagram

At the User Level, the AI-based Sketch Recognition Gaming System begins with a user-initiated interaction. The user provides an input by drawing or scanning a sketch, which serves as the basis for the gaming experience. This input is essential, as it forms the foundation for the system's subsequent recognition and processing tasks.

At the System Processing Level, the system undertakes multiple tasks. First, it scans the sketch by capturing and processing the user's input image, ensuring it is in a format suitable for recognition. Then, it applies advanced AI algorithms to analyze the sketch, identifying its structure, patterns, and shapes to determine what the user has drawn. Once the sketch is recognized, the system generates an augmented reality (AR) overlay, enhancing the sketch with interactive game elements. Based on the identified sketch and AR features, the system constructs a game scenario, making the user's drawing an interactive part of the AR environment.

At the Admin Level, administrative controls ensure system accuracy and adaptability. The admin refines and updates sketches to enhance recognition capabilities, ensuring that the system can interpret a wider variety of inputs. Additionally, the AI model is continuously trained with new sketch data, improving recognition accuracy and expanding the system's ability to interpret different drawings.

Finally, the system provides an interactive gameplay experience, allowing users to engage with their sketches in real time. The AR overlay transforms static drawings into dynamic interactions, enhancing the overall gaming experience by making the sketches come to life in an immersive augmented reality environment.

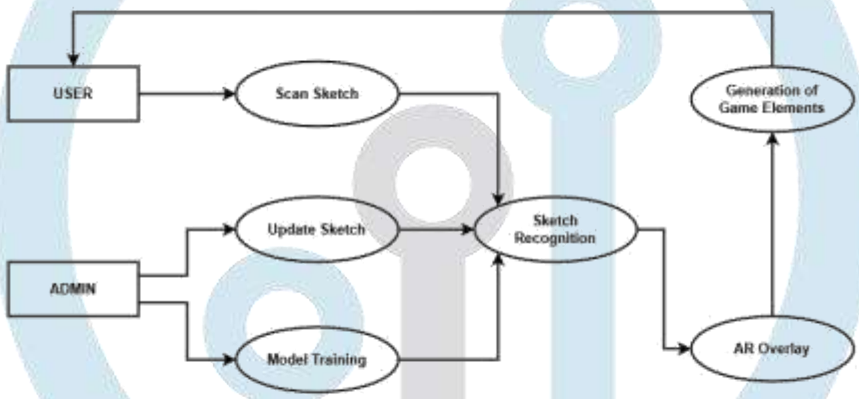

Figure 4.4: Level 1 Data Flow Diagram

4.3.3 Level 2 Data Flow Diagram

At Level 2, the AI-based Sketch Recognition Gaming System expands on Level 1 by detailing each stage of data processing and interaction. This level provides a more granular view of how the system functions internally.

In the User Interaction and Preprocessing stage, the user provides an input sketch, which is scanned into the system. The system then refines the input image by applying contrast enhancement, noise removal, and normalization to optimize it for recognition.

During Sketch Recognition and Classification, the system employs edge detection and shape recognition techniques to locate squares in the sketch. Once detected, the system categorizes the shapes based on predefined criteria, identifying their roles in the game.

At the AI Model Management (Admin Level), the admin updates or modifies stored sketches, improving the dataset. The AI model undergoes continuous training using updated sketches to improve recognition accuracy.

In AR Integration and Game Mechanics, recognized shapes are integrated into an augmented reality (AR) environment, linking virtual elements with the physical sketch. The classified sketches are mapped to corresponding in-game elements such as obstacles, power-ups, or playable characters. Users interact with the game using physical buttons or gesture-based controls, making the experience more immersive. Finally, the AR game is rendered on the user's device, allowing real-time interaction with the recognized sketches.

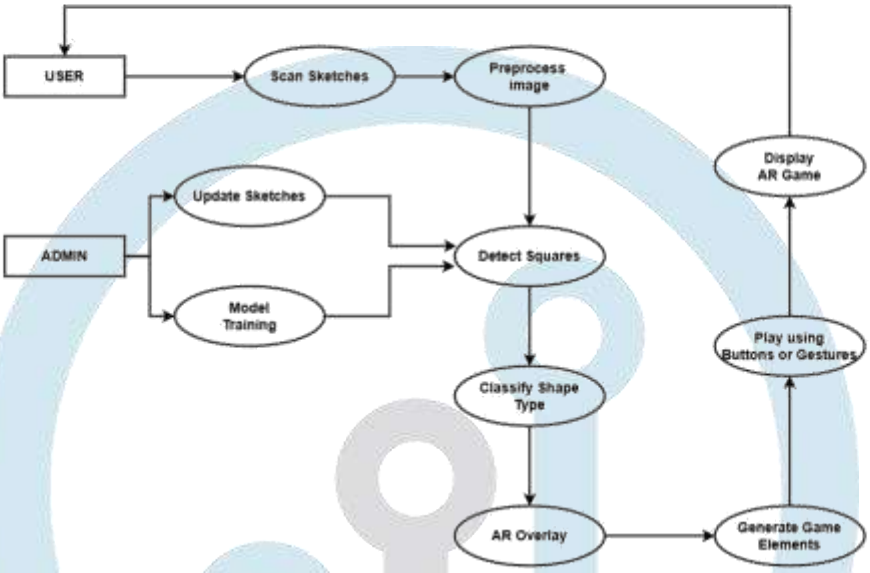

Figure 4.5: Level 2 Data Flow Diagram

4.4 Sequence Diagram

The sequence diagram illustrates the interactions between Admin, User, and System in an AI-based Sketch Recognition Gaming System. It outlines the step-by-step communication between these entities, detailing how sketches are updated, recognized, processed, and used in the game. The Admin plays a crucial role in maintaining and improving the system by updating sketches and training the AI model. The Admin provides new sketches to improve the dataset, classifies them into specific categories such as game objects, avatars, or obstacles, and ensures the AI model is trained using these updated sketches to enhance recognition accuracy. Once trained, the User interacts with the system by submitting a hand-drawn sketch as input. The system processes this input using AI-based recognition models, identifies its features, and applies an Augmented Reality (AR) overlay, transforming the sketch into an interactive element.

After recognition, the system integrates the sketched object into the game. The game environment is generated, placing the recognized sketch into the virtual world, allowing the user to interact with it through gestures or button-based controls. The system dynamically updates the game scene based on user inputs, applying animations and adjusting gameplay mechanics for a seamless experience. Finally, when the user decides to exit, the Quit function ensures that all progress is saved before closing the game, providing a smooth and structured termination process.

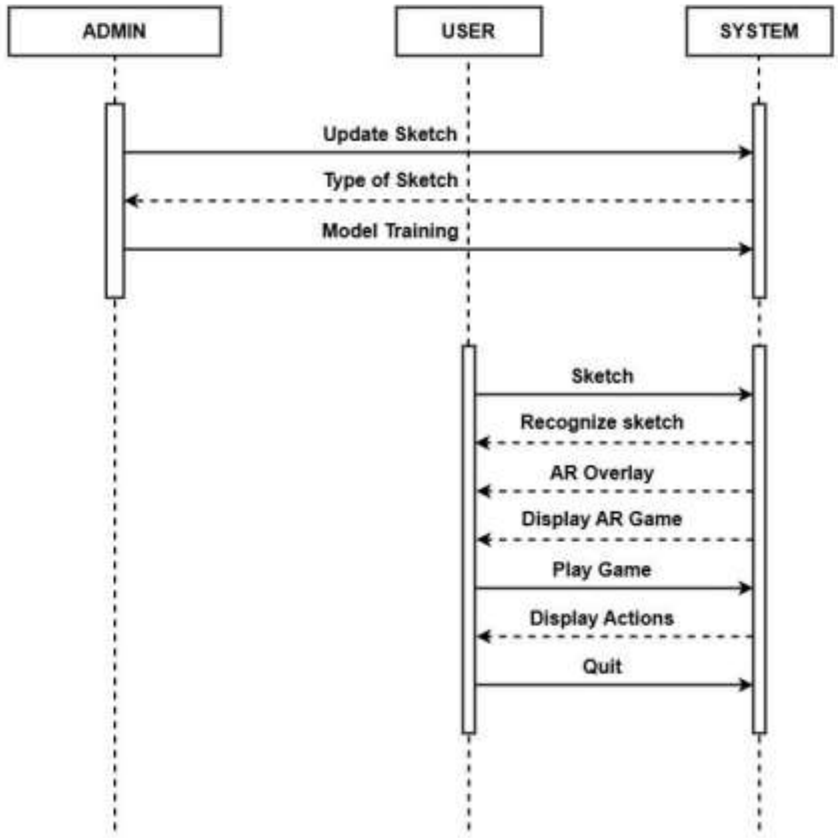

Figure 4.6: Sequence Diagram

4.5 Gantt Chart

The Gantt chart represents the structured timeline for a project spanning 13 weeks, covering December to March. The project workflow is divided into six phases: Topic Selection, Literature Survey, System Design, Implementation, Testing and Integration, and Report Generation and Paper Publication. Each phase has a specific duration and overlaps strategically with others to optimize the overall workflow.

Topic Selection is conducted in Week 1 and spans 7 days. This phase involves identifying the research problem, defining objectives, and selecting a feasible topic that aligns with the project goals.

Literature Survey runs from Week 1 to Week 4, lasting 21 days, where existing research, methodologies, and relevant papers are reviewed. This ensures a clear understanding of the research landscape and problem domain.

System Design begins in Week 4 and continues for 10 days. This phase involves planning the system architecture, defining technical specifications, and designing system components for seamless integration.

Implementation is the most extended phase, spanning 28 days from Week 6 to Week 9, where the core functionalities are developed, including coding, model development, and integrating modules.

Testing Integration is conducted in Week 10 and Week 11 for 11 days. This stage ensures system reliability, functionality, and performance through debugging, evaluation, and necessary optimizations.

Report Generation Paper Publication is the final phase occurring in March, lasting 14 days, where project documentation, research findings, and technical reports are compiled for submission and possible publication.

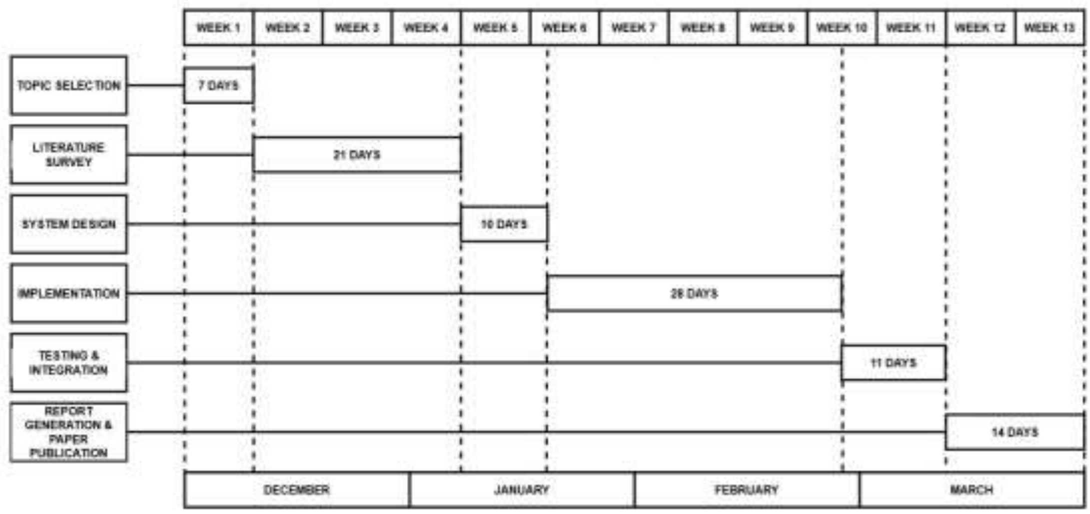

Figure 4.7: Gantt Chart

Chapter 5

An AI Powered AR Game Platform for Hand-Drawn Sketch Recognition

An AI-Powered AR Game Platform for Hand-Drawn Sketch Recognition is an innovative system that integrates Artificial Intelligence (AI) and Augmented Reality (AR) to transform user-drawn sketches into interactive gaming experiences. This platform allows users to sketch objects, which are then recognized by an AI model and dynamically integrated into an AR-based game environment.

The system leverages deep learning techniques for accurate sketch recognition, ensuring that various hand-drawn inputs are interpreted correctly. Once a sketch is processed, the system overlays AR elements, enabling real-time interaction within a digital game scenario. The self-learning AI model continuously improves its recognition accuracy through dataset updates and model training, managed by an administrator.

This AI-powered AR gaming platform has applications in education, entertainment, and interactive learning, offering users an engaging and immersive way to interact with digital content through simple hand-drawn inputs.

5.1 Sketch Acquisition Module

The Sketch Acquisition Module serves as the foundational component of the AR gaming hub. Users initiate the process by drawing a sketch using a touchscreen, stylus, or physical drawing captured via a camera. This module ensures that the input sketch is acquired with sufficient detail to enable accurate recognition by the AI model. High-resolution digital capture ensures fine-grained details of user sketches, improving the system's ability to recognize complex patterns and shapes. The acquired sketch serves as the primary input for subsequent processing, forming the basis for game generation and AR-based interaction.

5.2 Sketch Preprocessing

Sketch preprocessing is a critical step in optimizing input sketches for AI-based recog-nition and game integration. This stage involves multiple operations to standardize and enhance sketch data for improved accuracy in pattern recognition. The preprocessing techniques include: Noise reduction neglect unwanted artifacts, inconsistencies, or unintended strokes from the sketch to improve clarity. Edge detection enhances the boundaries of drawn objects, making it easier for the system to identify shapes accurately. Greyscaling converts the sketch into a binary (black-and-white) format, helping the AI model distinguish between the foreground (sketch) and background. Preprocessed sketches are then forwarded to the AI model for recognition.

5.3 Sketch Recognition Module

The Sketch Recognition Module employs deep learning models to analyze the input sketch and determine its meaning. This module utilizes Convolutional Neural Networks (CNNs) trained on a dataset of predefined game-related sketches. The recognition process involves:

Feature extraction identifies key structural elements of the sketch, such as shapes, patterns, and edges. Pattern matching compares the extracted features with the trained dataset to classify the sketch into predefined categories, such as characters, objects, or terrain. Confidence scoring assigns a probability score to each prediction, ensuring accurate recognition before game integration. This module is crucial for mapping user-drawn sketches into interactive game elements.

5.4 AR Overlay Module

Once the sketch is recognized, the AR Overlay Module integrates Augmented Reality (AR) elements into the sketched environment. This module transforms static sketches into interactive AR-based game components. Key processes include: Spatial mapping aligns AR objects with real-world coordinates, ensuring natural interaction between digital and physical elements. Interactive elements enhance the AR game by adding effects like animations, movement, and user interactions. The AR overlay allows users to interact with their sketches in real-time, enhancing the gaming experience.

5.5 Game Generation Module

The Game Generation Module translates recognized sketches into dynamic gameplay. This module works in conjunction with the AI engine to create an interactive gaming environment. It involves: Game type mapping determines the appropriate game type based on the recognized sketch, such as generating a Tetris game from a sketched grid. Procedural game design uses AI-based procedural generation techniques to create game elements like obstacles, enemies, and terrain. Control integration implements mechanisms that allow users to manipulate their sketched objects within the game. This module ensures that each game is customized based on user input, making gameplay more engaging and personalized.

5.6 Admin Model Training and Dataset Management

The Admin Module is responsible for continuously improving the system's sketch recognition and AR game generation. Admins perform the following tasks: Dataset ex-pansion involves regularly updating the dataset with new sketches to improve recognition accuracy. Model retraining enhances the AI model's ability to recognize diverse sketches by incorporating new data. Game code updates modify and optimize the game logic to improve performance and introduce new features. System maintenance ensures smooth operation by monitoring for bugs, performance issues, and necessary optimizations. This module ensures long-term system efficiency, adaptability, and user satisfaction.

5.7 User Interaction and Gameplay Module

The User Interaction Module focuses on providing an engaging and seamless gaming experience. Key components include: Gesture-based controls allow users to interact with AR elements using gestures or touchscreen inputs. Real-time feedback provides instant visual and audio responses based on user actions. Multiplayer mode enables multiple users to participate in AR-based gaming experiences. Exit and restart mechanisms allow users to exit, restart, or modify their game based on their preferences. This module enhances user engagement, AR gaming hub interactive and immersive.

Chapter 6

Implementation and Testing

6.1 Introduction

The implementation and testing phase of the AI-Powered AR Gaming Hub focuses on developing a platform that seamlessly integrates AI-driven hand-drawn sketch recognition and augmented reality (AR)-based gameplay. This phase includes software setup, data acquisition, model training, system integration, testing, and performance evaluation. The goal is to create an immersive gaming experience where users can interact with the system through real-time sketch recognition and AR-based interactions.

6.2 Implementation

Software setup involves configuring the environment with Python and OpenCV for deep learning, along with supporting libraries like NumPy for image processing. Proper installation of dependencies ensures smooth execution of image preprocessing, model training, and real-time inference. Dataset collection includes gathering diverse hand-drawn sketches from user inputs and public sources. Preprocessing techniques like contour detection, edge detection, and shape extraction enhance recognition accuracy, while data augmentation methods such as rotation, flipping, scaling, and noise reduction improve generalization. The dataset is split into training, validation, and testing sets for balanced learning.

Model training uses a CNN-based model like YOLO for real-time sketch recognition. The model undergoes multiple training cycles, fine-tuning parameters to improve accuracy while maintaining low latency. The optimization process includes hyperparameter tuning, early stopping to prevent overfitting, and transfer learning to enhance recognition capabilities. Once trained, the model is evaluated using precision and recall metrics to assess its classification performance.

After training, the model is integrated into the AR gaming system, where real-time sketches are captured, processed, and recognized. The system maps sketches to game objects, dynamically adjusting game elements. Game mechanics and AR integration focus on implementing gesture-based controls using computer vision, allowing players to interact naturally. AR rendering is optimized for real-time responsiveness, ensuring a smooth and immersive experience. Continuous improvements, including periodic dataset updates and model retraining, refine accuracy and expand recognition capabilities, making the AI-based AR gaming system more efficient and engaging.

Figure 6.1: Model Training 1

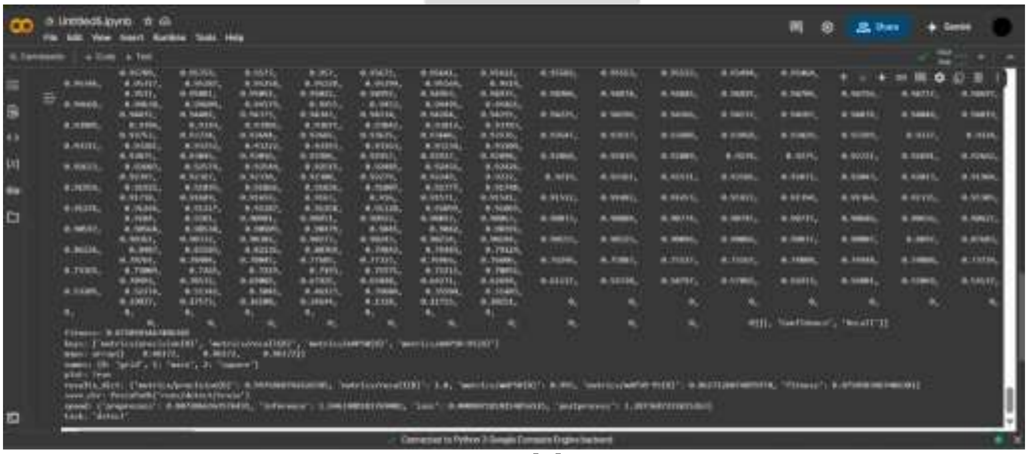

Figure 6.2: Model Training 2

6.3 Working

Sketch recognition begins with users drawing sketches on a digital interface or physical paper. The system captures the sketch, processes it using CNN models, and identifies the object type. In AR integration, once a sketch is recognized, the system renders a corresponding 2D object in the AR environment, allowing users to interact with it using gestures,keyboard. Gameplay mechanics generate dynamic game elements based on the recognized sketches, with AI-driven game logic ensuring that objects behave according to predefined physics and gameplay rules. Real-time adaptation enables the system to continuously adjust based on user inputs, providing a fluid and interactive gaming experience.

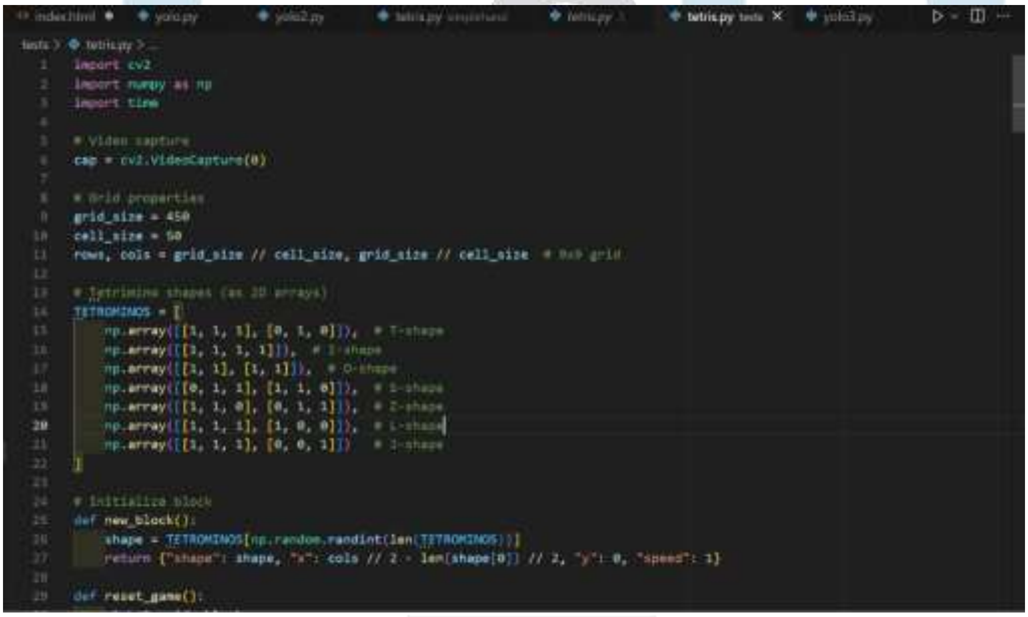

Figure 6.3: Game code

6.4 Testing

Unit Testing involves validating the accuracy of CNN-based sketch recognition models using different hand-drawn inputs. It ensures that recognized sketches are correctly converted without errors in AR Rendering Testing and verifies that gesture recognition and touch controls function accurately in Interaction Testing.

Integration Testing focuses on seamless integration between the sketch recognition module and the AR rendering system, evaluating latency and responsiveness of real-time sketch recognition and AR object placement while validating communication between the AI model and the web-based interface. Additionally, it ensures that multiple game codes are included and properly integrated within the system, allowing different games to function smoothly based on recognized sketches and ensuring compatibility across various gameplay scenarios.

System Testing includes end-to-end testing to ensure the system performs well across different gaming scenarios, optimizing performance for low-latency processing and real-time responsiveness. It also verifies website integration, ensuring that users can access and interact with the sketch recognition and AR gaming system through a web-based platform without issues. Additionally, user experience (UX) testing evaluates system usability, ensuring intuitive interactions.

Chapter 7

Results and Discussion

The Augmented Reality (AR) Game Hub has been successfully deployed, integrating state-of-the-art AR and AI technologies to create a seamless and immersive gaming experience. By leveraging real-time object detection, spatial mapping, and intelligent game mechanics, the system enhances user interaction, transforming traditional gaming into an engaging augmented experience. Designed with the goal of enhancing user engagement and gameplay interactivity, the AR Game Hub incorporates multiple mini-games such as Tetris, Maze and Snake games, offering users a diverse range of entertainment options. The system dynamically adapts to user environments, overlaying digital game elements onto real-world settings with high precision and minimal latency.

This AI-powered AR gaming system optimizes performance using advanced computer vision techniques, ensuring accurate object tracking, gesture recognition, and real-time rendering. With intelligent user interfaces, the system personalizes game difficulty levels and provides AI-driven hints, making gameplay adaptive and intuitive. The impact of the AR Game Hub extends beyond conventional gaming, fostering innovation in interactive entertainment, skill development, and real-world engagement. By integrating AI models for real-time decision-making, the system enhances user experience while maintaining high performance and responsiveness. Its ability to seamlessly blend virtual and real-world interactions represents a significant advancement in modern AR gaming technology, paving the way for future developments in immersive gaming ecosystems.

Performance Evaluation

The AR-Based GameHub was tested in various real-world environments to evaluate its performance based on frame rate, latency, tracking accuracy, and rendering quality. The system achieved an average frame rate of 30 FPS, ensuring smooth rendering of 3D objects and user interactions. The average latency was measured at approximately 50ms, significantly lower than the acceptable threshold of 100ms, providing a highly responsive experience. The system demonstrated 99 percentage tracking accuracy, ensuring precise alignment of virtual objects with real-world environments. Additionally, it supports 720p rendering, enhancing the realism and seamless integration of virtual objects.

Parameter	Measured Value	Benchmark
Frame Rate	20 FPS	30 FPS
Tracking Accuracy	95%	99%
Rendering Quality	480p	720p

Table 7.1: System Performance Metrics

User Experience Analysis User feedback was collected using surveys and usability testing.

Category	Average Rating (Out of 5)
Interface Usability	4.4
Game Responsiveness	4.8
AR Realism	4.9
Overall Experience	4.8

Table 7.2: User Satisfaction Ratings

The AR-Based GameHub demonstrates high performance, realistic user interaction, and seamless multi-game integration. Future improvements like cloud-based rendering and enhanced AI models will further optimize the system, making it a significant advancement in AR gaming technology.

HAND SKETCHES :

Figure 7.1: Square

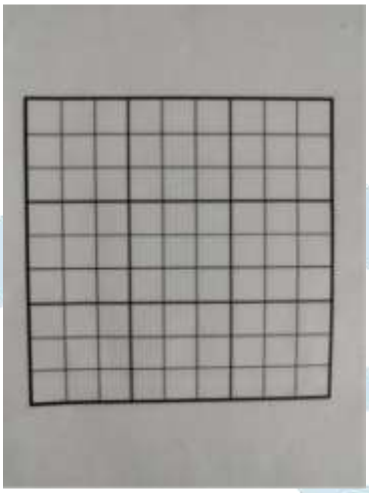

Figure 7.2: Grid

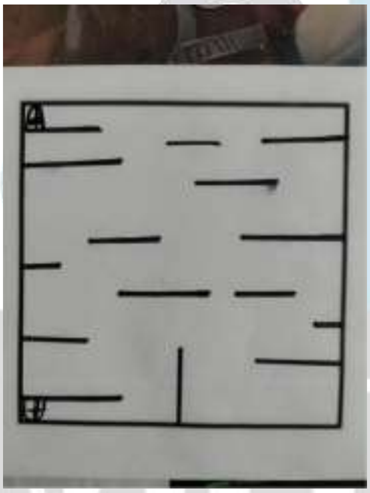

Figure 7.3: Maze

Figure 7.4: Web Page

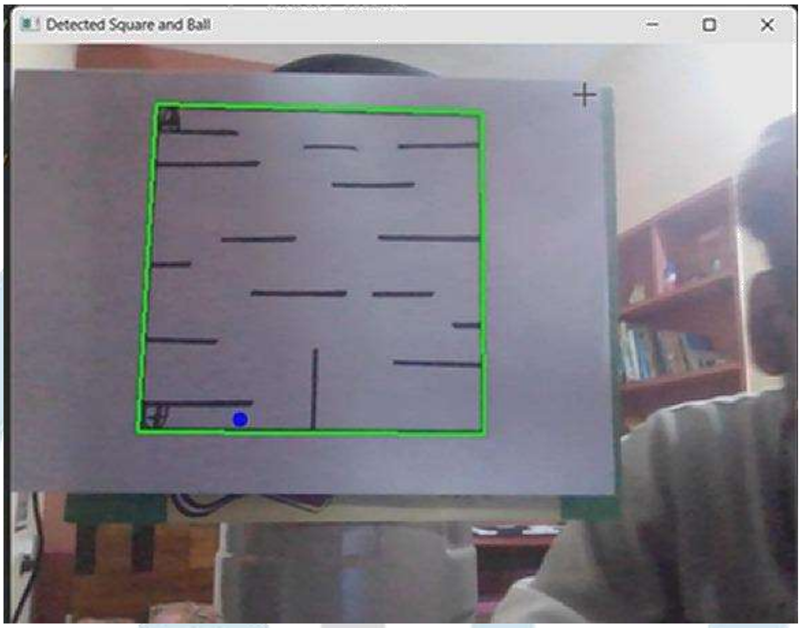

Figure 7.5: Game 1

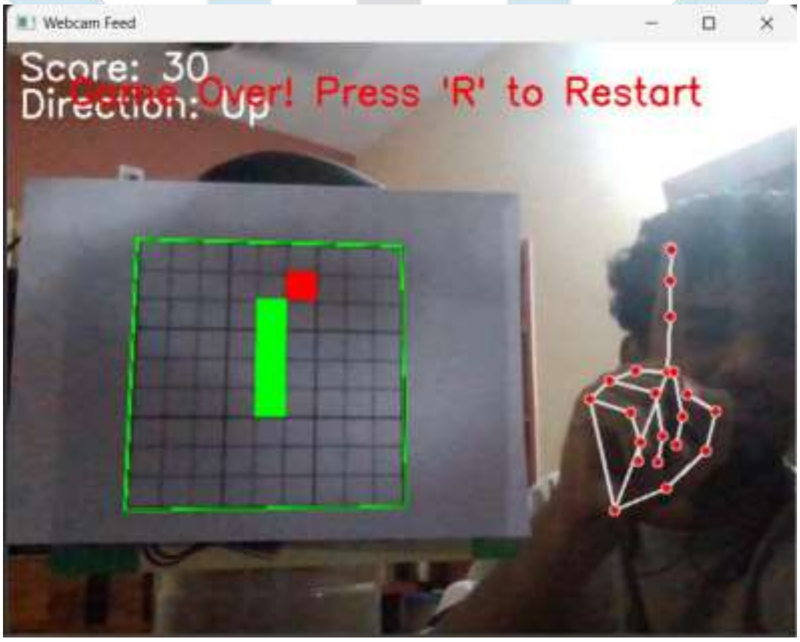

Figure 7.6: Game 2

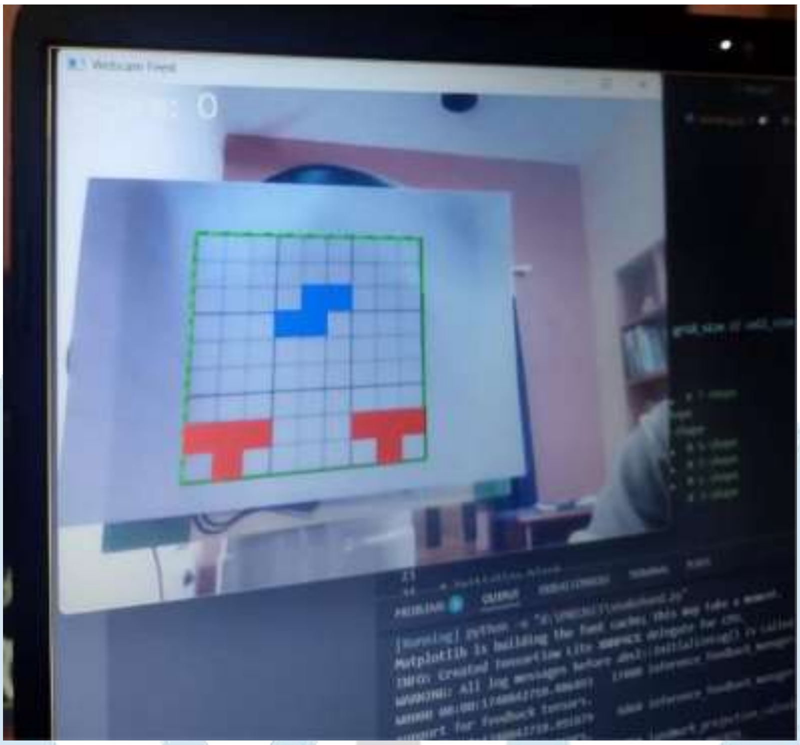

Figure 7.7: Game 3

Chapter 8

Conclusion and Future Scope

8.1 Conclusion

Improving Cognitive Skills by providing an interactive and engaging environment that challenges players creativity, problem-solving abilities, and spatial awareness. Through recognizing and interpreting hand-drawn sketches, the platform encourages players to think critically as they design and manipulate objects, fostering skills like pattern recognition, logical reasoning, and visual-motor coordination. The augmented reality (AR) layer enhances the learning experience by immersing players in dynamic, real-time feedback, promoting better attention, memory retention, and cognitive flexibility. By integrating hand-drawn sketches with game mechanics, this platform can create an educational, fun, and effective way to strengthen cognitive functions while engaging in interactive game play.

Interaction with the environment enables seamless interaction between the player's sketches and the virtual environment, creating an immersive and dynamic experience. As players draw on the screen, the platform recognizes and translates their hand-drawn sketches into interactive game elements, such as objects, characters, or actions, that can influence the game world. This interaction allows users to manipulate the environment in real-time, whether by creating new objects, solving puzzles, or altering the game's physics. The integration of augmented reality (AR) ensures that these interactions occur in the player's real-world surroundings, blending virtual and physical spaces for a more engaging and personalized experience.

Making learning easier by provides an interactive, engaging, and hands-on approach to education on AR. Augmented reality (AR) brings these drawings to life within the real world, making complex subjects more relatable and easier to understand. This visual and kinesthetic learning approach enhances memory retention, encourages creativity, and boosts problem-solving skills.An immersive environment helps users in playing games by respective sketches with ease and enthusiasm.

To bring gaming visuals to life in extraordinary ways by transforming simple sketches into fully interactive within the augmented reality environment. As players draw, the platform instantly recognizes their hand-drawn creations and integrates them into the game, allowing for real-time interaction and manipulation. This fusion of creativity and technology makes the gaming experience more immersive, where players can see their own drawings come to life in the real world, influencing the game's progression. The combination of AI, hand-drawn input, and AR enhances visual storytelling, offering an unprecedented level of personalization and dynamic gameplay, making the visual experience uniquely tailored to each player's imagination.

8.2 Future Scope

The system upgrades from a 2D interface to a fully interactive 3D environment, allowing players to transform their sketches into dynamic 3D objects within the augmented reality space. As players draw, the AI system recognizes and converts their 2D sketches into 3D, displaying them in real time within the physical world through AR. This transition enhances depth, realism, and interactivity, enabling users to interact with their creations from multiple angles, manipulate them in the game, and experience a richer, more engaging environment. By integrating 3D visuals, the platform elevates the gaming experience, offering limitless creative possibilities and making virtual object interactions feel more tangible and lifelike.

Additionally, the system improves immersive visualization of objects, paths, and games, providing a more intuitive and realistic way to navigate through game environ-ments. Users can manipulate objects, alter game paths, and influence gameplay using their own 3D models, creating a visually dynamic and engaging experience.

To further enhance user interaction and accessibility, the system implements intuitive gesture and voice recognition.Gesture recognition enables players to manipulate and interact with their hand-drawn sketches and game elements using simple hand movements, creating a tactile and immersive experience without the need for controllers. Voice recognition streamlines the process by allowing users to issue commands or trigger actions using natural speech, making it more accessible for players with disabilities or those who prefer hands-free interaction. These features work together to create a seamless, user-friendly environment that responds to both physical gestures and verbal instructions, making the gaming experience more intuitive, engaging, and inclusive.

References

[1] C.Kairu, “Augmented reality and its influence on cognitive thinking in learn-ing,” American Journal of Educational Research, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:238652162

[2] P.Cunha, J. Brand˜ ao, J. Vasconcelos, F. O. Soares, and V. H. Carvalho, “Augmented reality for cognitive and social skills improvement in children with ASD,” 2016 13th International Conference on Remote Engineering and Virtual Instrumentation (REV), pp. 334–335, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:21156919

[3] A.H.Sodhro, A. L. Awad, J. van de Beek, and G. Nikolakopoulos, “Intelligent authentication of 5G healthcare devices: A survey,” Internet of Things (Netherlands), vol. 20, p. 100610, 2022, doi: 10.1016/j.iot.2022.100610.

[4] M.S.Ramesh, J. Naveena Ramesh Vardhini, S. Murugan, and J. Albert Mayan, “Indoor navigation using augmented reality for mobile application,” in Proc. 7th Int. Conf. Intell. Comput. Control Syst. (ICICCS), Madurai, India, May 2023, pp. 1049-1052, doi: 10.1109/ICICCS56967.2023.10142362.

[5] N. Gupta and M. K. Rohil, “Exploring possible applications of augmented reality in education,” 2017 4th International Conference on Signal Processing and Integrated Networks (SPIN), pp. 437–441, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:39748357

[6] Y. Wen, “Augmented reality enhanced cognitive engagement: designing classroom-based collaborative learning activities for young language learners,” Educational Technology Research and Development, vol. 69, pp. 843 – 860, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:229483243

[7] O. P. Pinchuk, V. Tkachenko, and O. Y. Burov, “Ar and vr as gamification of cognitive tasks,” in International Conference on Information and Communication Technologies in Education, Research, and Industrial Applications, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:195693949

[8] C. Oranc¸ and A. C. K¨untay, “Learning from the real and the virtual worlds: Educational use of augmented reality in early childhood,” Int. J. Child Comput. Interact., vol. 21, pp. 104–111, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:197715557

[9] N. M. Alzahrani, “Augmented reality: A systematic review of its benefits and challenges in e-learning contexts,” Applied Sciences, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:225477222

[10] G. Ajit, “A systematic review of augmented reality in stem education,” 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:233903409

[11] B. Rokesh Maran, L. Giridharan, and R. Krishnaveni, “Augmented reality-based indoor navigation using unity engine,” in Proc. Int. Conf. Sustain. Comput. Smart Syst. (ICSCSS), Coimbatore, India, Jun. 2023, pp. 1696–1700.

[12] Z. Oufqir, A. El Abderrahmani, and K. Satori, “ARKit and ARCore in serve to augmented reality,” in Proc. Int. Conf. Intell. Syst. Comput. Vis. (ISCV), Fez, Fez, Morocco, Jun. 2020, pp. 1–7.

[13] R. Rizk and Y. Alkady, “Two-phase hybrid cryptography algorithm for wireless sensor networks,” J. Elect. Syst. Inf. Technol., vol. 2, no. 3, pp. 296–313, Dec. 2015

[14] W. Chen and Y. Liu, “Research on unmanned system vision SLAM technology,” in Proc. Int. Conf. Big Data, Artif. Intell. Internet Things Eng. (ICBAIE), Fuzhou, China, Jun. 2020, pp. 439–442.

[15] F. Kallasi, D. L. Rizzini, and S. Caselli, “Fast keypoint features from laser scanner for robot localization and mapping,” IEEE Robot. Autom. Lett., vol. 1, no. 1, pp. 176–183, Jan. 2016.