

AR-Based Interactive Learning System for Engineering Labs Using 3D Motherboard Visualization

Authors and Affiliations

Aditya R. Jamunpane^{1*}, Arnav Wasnik², Aniket Thaokar³

Bhushan Kawale⁴, Aditya Meshram⁵, Rashmi Janbandhu⁶

¹G.H Raison College of Engineering and Management, Nagpur

^{1*}Corresponding Author

Aditya R. Jamunpane

G.H Raison College of Engineering and Management, Nagpur Maharashtra, India, Pin – 4400016

Abstract

Computer hardware education often faces challenges due to limited laboratory resources, restricted access to physical hardware components, and the difficulty of understanding complex motherboard architectures through conventional teaching methods. Traditional learning approaches primarily rely on textbooks, static diagrams, and limited practical sessions, which may not effectively support the development of component identification skills and spatial understanding of hardware structures. To address these limitations, this research presents an interactive WebAR-based learning platform that enables immersive exploration of computer hardware through real-time three-dimensional visualization and Augmented Reality technologies. The proposed system integrates both marker-based and markerless AR modes, allowing learners to interact with virtual hardware models directly through standard web browsers without requiring additional software installation. The platform supports multiple hardware environments, including computer motherboards, Arduino Uno boards, Raspberry Pi systems, and drone flight controllers within a unified educational framework.

The system incorporates optimized 3D rendering, raycasting-based component selection, dynamic object interaction, component isolation, and contextual information delivery to enhance user engagement and learning effectiveness. By enabling active exploration of hardware structures, the platform improves conceptual understanding, component recognition, and spatial visualization skills. Furthermore, its browser-based architecture provides a scalable, cost-effective, and accessible alternative to traditional hardware laboratories. The proposed WebAR learning environment demonstrates the potential of modern immersive technologies to bridge the gap between theoretical instruction and practical hardware education, thereby supporting more interactive, flexible, and technology-driven engineering learning experiences.

1. Introduction

The rapid advancement of digital technologies has significantly transformed teaching and learning methodologies across engineering and technical education. Modern educational practices increasingly emphasize interactive, learner-centered, and experiential approaches that encourage active engagement with complex concepts. However, computer hardware education continues to face several challenges, particularly in laboratory-based environments where students are expected to understand the structure, functionality, and interconnections of physical hardware components. Traditional instructional methods primarily rely on textbooks, presentation slides, static diagrams, and limited laboratory demonstrations, which often provide insufficient support for developing a comprehensive understanding of hardware architectures and component relationships [1], [2].

Computer motherboards, Arduino microcontrollers, Raspberry Pi systems, and embedded controller boards consist of numerous interconnected components arranged within compact and complex structures. Understanding the placement and functionality of processors, memory modules, expansion slots, power connectors, chipsets, and communication interfaces requires strong spatial visualization skills. However, conventional two-dimensional learning resources often fail to effectively represent depth, component hierarchy, and real-world assembly procedures. As a result, students frequently encounter difficulties in identifying hardware components, understanding their interactions, and applying theoretical knowledge during practical laboratory sessions [3], [4].

In addition to educational challenges, engineering institutions frequently face financial and operational constraints in maintaining hardware laboratories. Physical equipment is often expensive, limited in quantity, and susceptible to damage during repeated practical use. Furthermore, laboratory access is generally restricted to scheduled sessions, reducing opportunities for self-paced learning, remote education, and independent exploration. These limitations can negatively affect learner engagement, practical skill development, and overall learning outcomes [5], [6].

Recent advances in Augmented Reality (AR) have introduced new possibilities for enhancing engineering education through immersive and interactive learning experiences. AR technology enables the integration of virtual content with real-world environments, allowing learners to visualize and interact with complex systems in three dimensions. Previous studies have reported that AR-based educational environments can improve learner engagement, knowledge retention, spatial understanding, and practical skill acquisition when compared with conventional teaching methods [7]–[10].

Simultaneously, the emergence of WebAR technologies has expanded the accessibility of immersive learning environments. Modern web technologies and browser-based rendering frameworks enable educational applications to operate directly within standard web browsers without requiring dedicated software installation. Technologies such as Three.js and WebGL facilitate the development of lightweight, scalable, and cross-platform visualization systems that support interactive learning experiences across desktop computers, tablets, and mobile devices [16], [17]. Consequently, WebAR has emerged as a practical solution for delivering immersive educational content while reducing deployment complexity and improving accessibility.

Despite these advancements, existing educational AR solutions exhibit several limitations. Many systems primarily focus on visualization and component identification while offering limited support for active interaction, component-level manipulation, and exploratory learning activities. Some platforms rely exclusively on marker-based tracking approaches, reducing usability in situations where markers are unavailable or impractical. Furthermore, most existing solutions are designed for a single hardware platform and do not provide a unified environment for exploring multiple hardware ecosystems within the same learning framework [14], [15].

Another significant limitation is the lack of comprehensive WebAR-based educational systems that integrate both marker-based and markerless AR technologies while supporting multiple hardware platforms within a browser-accessible environment. This limitation restricts opportunities for learners to develop a broader understanding of hardware architectures through immersive exploration, real-time interaction, and contextual learning support [14], [15].

A. Research Gap and Objective

Although previous studies have demonstrated the educational benefits of AR and WebAR technologies, there remains a gap in developing a unified browser-based platform that combines marker-based and markerless AR, supports multiple hardware ecosystems, and provides advanced component-level interaction capabilities. Existing solutions often emphasize visualization rather than interactive exploration and frequently lack scalability across different hardware learning scenarios.

To address these challenges, this research proposes an interactive WebAR-based hardware learning platform that combines marker-based and markerless Augmented Reality with real-time three-dimensional visualization. The developed system enables learners to explore computer motherboards, Arduino Uno boards, Raspberry Pi systems, and drone flight controllers through dynamic interaction, component isolation, object manipulation, and contextual educational guidance. Developed using modern web technologies, including React, Three.js, WebGL, and AR

frameworks, the platform provides a scalable, cost-effective, and accessible virtual laboratory experience that operates directly within standard web browsers.

The primary objective of this research is to design and develop an immersive WebAR learning environment that facilitates hardware exploration through real-time 3D visualization, component-level interaction, and contextual educational content delivery. The proposed platform aims to enhance conceptual understanding, improve spatial learning abilities, reduce dependence on physical laboratory resources, and contribute toward the development of scalable next-generation engineering education systems.

2. Literature Review

The adoption of Augmented Reality (AR) in education has increased significantly over the past decade due to its ability to enhance learner engagement, visualization, and conceptual understanding. AR enables the integration of virtual content with real-world environments, allowing students to interact with digital representations of complex systems in an immersive manner. Previous studies have demonstrated that AR-based learning environments can improve student motivation, participation, and knowledge retention, particularly in engineering and technical education where spatial understanding plays a crucial role [1], [7].

Several researchers have investigated the application of AR in Science, Technology, Engineering, and Mathematics (STEM) education. Studies have reported that interactive AR environments support deeper cognitive processing by enabling learners to visualize three-dimensional structures, explore object relationships, and understand abstract concepts more effectively than traditional teaching approaches. Experimental findings indicate that AR-assisted learning can improve academic performance, problem-solving skills, and long-term retention of technical knowledge [3], [5]–[8].

Recent advances in browser-based technologies have further expanded the accessibility of immersive educational systems. The emergence of WebAR frameworks, WebGL rendering engines, and modern three-dimensional graphics libraries has enabled educational applications to operate directly within standard web browsers without requiring dedicated software installation. This browser-native approach reduces deployment complexity, enhances scalability, and improves accessibility across smartphones, tablets, and desktop computers [16], [17]. As a result, WebAR has emerged as a promising solution for delivering cost-effective and device-independent learning experiences.

In the domain of computer hardware education, researchers have explored virtual laboratories, hardware assembly simulators, and interactive visualization systems to overcome the limitations of traditional laboratory-based learning. These solutions allow learners to explore hardware structures, understand component functionalities, and gain practical knowledge without requiring direct access to physical laboratory equipment. Existing studies have shown that interactive hardware visualization environments can improve component recognition, assembly accuracy, and learner confidence during practical exercises [10], [11], [14].

Despite these advancements, several limitations remain evident in existing educational AR systems. Many platforms primarily focus on visualization and component identification while providing limited opportunities for active interaction and component-level manipulation. Such systems often lack features that allow learners to isolate, inspect, and dynamically interact with individual hardware components. Consequently, the learning experience may remain observational rather than exploratory [12], [14].

Another limitation is the widespread dependence on marker-based tracking mechanisms. Although marker-based AR provides reliable object placement, it restricts usability in scenarios where markers are unavailable, damaged, or difficult to maintain. Markerless AR techniques address these limitations by enabling virtual object placement without predefined visual markers; however, relatively few educational platforms integrate both marker-based and markerless modes within a unified learning environment [10], [15].

Furthermore, most existing solutions focus on a single hardware platform or educational scenario. Systems designed for motherboard learning, microcontroller education, or embedded systems training are typically developed as separate applications. This fragmented approach

limits their applicability across broader engineering curricula and prevents learners from exploring multiple hardware ecosystems within a single educational framework [11], [14].

Comparative Analysis of Existing Studies

Author / Study	Technology	Educational Focus	Limitation
Lu et al.	AR Learning Environment	General Education	Limited hardware interaction
Kumar et al.	AR-Based Hardware Learning	Computer Components	Single-device focus
Silva et al.	Markerless AR	Component Visualization	Lack of integrated learning framework
Rahman et al.	Virtual Hardware Laboratory	Engineering Education	Limited browser accessibility
Chen et al.	AR Assembly Training	Hardware Assembly	Marker-based implementation only
Existing Solutions	AR / WebAR	Hardware Education	Limited interaction and multi-device integration
Proposed System	WebAR + Marker-Based AR + Markerless AR	Interactive Hardware Education	Supports multiple hardware platforms, component isolation, real-time interaction, and browser-based accessibility

Research Gap

The literature indicates that although AR and WebAR technologies have demonstrated significant potential in engineering education, there remains a lack of comprehensive educational platforms that combine browser accessibility, marker-based and markerless AR integration, multi-device hardware support, real-time component interaction, and contextual educational guidance within a single framework. Existing solutions are often limited to visualization-centric learning and do not fully support immersive hardware exploration.

To address these limitations, the present research proposes a WebAR-based interactive hardware learning platform that enables learners to explore computer motherboards, Arduino Uno boards, Raspberry Pi systems, and drone flight controllers through real-time 3D visualization, component-level interaction, object isolation, and contextual information delivery. The proposed approach aims to provide a scalable, cost-effective, and accessible virtual laboratory environment capable of enhancing conceptual understanding and practical hardware learning.

3. Methodology and System Architecture

The proposed WebAR-based hardware learning platform was developed using a structured and iterative methodology to ensure educational effectiveness, technical scalability, and cross-platform accessibility. The development process consisted of requirement analysis, hardware data collection, three-dimensional model creation, web application development, implementation of interactive learning features, user interface design, testing, deployment, evaluation,

and continuous improvement. The overall development workflow of the proposed system is illustrated in Figure 1.

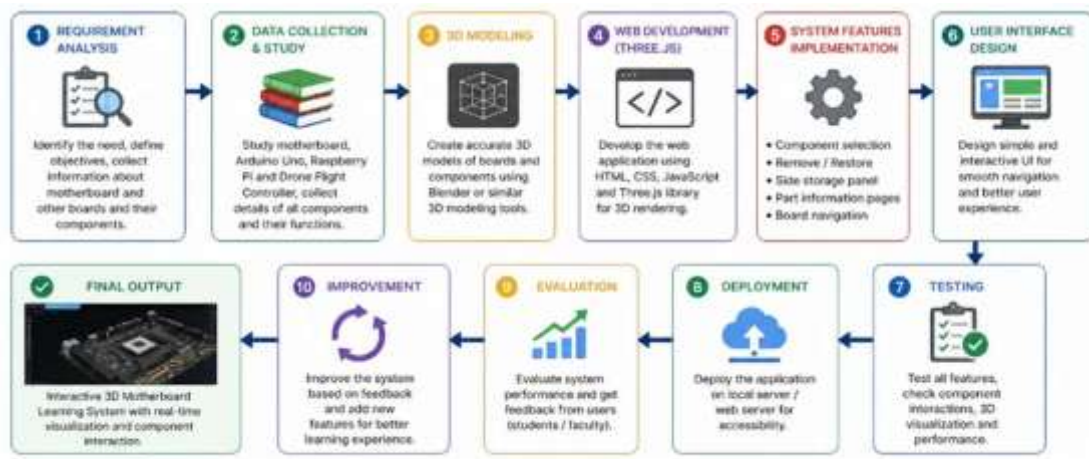


Figure 1: Block diagram of process flow

Figure 1. Development Workflow of the Proposed WebAR-Based Interactive Hardware Learning Platform

Figure 1 presents the complete development workflow adopted for the implementation of the proposed WebAR learning platform. The methodology begins with requirement analysis and hardware data collection, followed by three-dimensional model creation and web application development. Interactive learning functionalities and user interface components are subsequently integrated to enhance user engagement and educational effectiveness. The developed system then undergoes testing, deployment, and evaluation stages, while user feedback is incorporated through a continuous improvement cycle to ensure long-term scalability, usability, and learning effectiveness.

3.1 Requirement Analysis

The development process commenced with a detailed analysis of challenges associated with conventional computer hardware education. Observations from laboratory sessions and discussions with students revealed difficulties in identifying motherboard components, understanding hardware connectivity, and visualizing the spatial arrangement of electronic elements. Similar challenges have been reported in engineering education literature, where limited laboratory access and dependence on traditional teaching resources often restrict conceptual understanding and practical skill development [5], [6].

3.2 Hardware Data Collection and Content Preparation

Following requirement identification, detailed hardware information was collected from technical manuals, hardware documentation, and educational resources. Multiple hardware platforms, including computer motherboards, Arduino Uno boards, Raspberry Pi systems, and drone flight controllers, were studied. Component specifications, structural layouts, functional characteristics, and connectivity relationships were analyzed to ensure that the virtual representations accurately reflected real-world hardware configurations while maintaining educational relevance.

3.3 Three-Dimensional Model Development and Optimization

The next stage focused on the development and optimization of three-dimensional hardware models. Blender was used to create and refine the hardware assets, where mesh structures were optimized through polygon reduction, removal of redundant vertices, and material refinement. These optimization techniques minimized computational overhead while preserving visual quality. The finalized models were exported in GLB format to support efficient transmission, rapid loading, and compatibility with browser-based rendering environments [19].

3.4 Web Application Development

The WebAR platform was developed using React, TypeScript, Three.js, and WebGL technologies [16], [17], [20]. These technologies enabled the creation of an interactive three-dimensional environment capable of rendering

multiple hardware platforms within a unified browser-based workspace. By leveraging modern web technologies, the platform eliminates software installation requirements and provides accessibility across desktop computers, tablets, and smartphones while maintaining consistent rendering performance.

3.5 Interactive Learning Features Implementation

To enhance learner engagement and interactivity, several educational features were incorporated into the system. Users can select hardware components, isolate specific elements for detailed examination, restore removed components to their original positions, and access contextual information related to selected hardware parts. These interactive mechanisms transform passive observation into active exploration, thereby improving understanding of component functionality and hardware relationships.

3.6 User Interface Design

The user interface was designed with emphasis on usability, accessibility, and intuitive navigation. User interaction patterns were carefully considered to ensure that learners with varying levels of technical expertise could effectively utilize the platform. Responsive design principles were implemented to maintain functionality across different screen sizes and device categories. Navigation controls, information panels, and interaction tools were organized to minimize cognitive load while maximizing educational effectiveness.

3.7 Testing and Validation

Comprehensive testing procedures were conducted throughout development to evaluate system stability, rendering performance, interaction accuracy, and user experience quality. Functional testing verified the correct operation of model-loading mechanisms, interaction modules, and educational content delivery systems. Performance testing was additionally performed to assess rendering responsiveness and compatibility across multiple hardware configurations and browser environments.

3.8 Deployment and Evaluation

Following successful validation, the platform was deployed as a browser-accessible educational application. Learners can access the system directly through standard web browsers without requiring specialized software installations. Evaluation activities were conducted to assess usability, accessibility, and overall learning effectiveness. Feedback collected from users contributed to identifying areas for further refinement and enhancement.

3.9 Continuous Improvement and Methodological Significance

User feedback obtained during evaluation was incorporated into an iterative improvement cycle aimed at enhancing visualization quality, interaction efficiency, and overall user experience. This continuous development approach supports long-term scalability and adaptability as educational requirements evolve.

The proposed methodology establishes a comprehensive framework that integrates educational objectives with modern WebAR technologies. By combining hardware visualization, interactive learning mechanisms, browser-based accessibility, and cross-platform compatibility, the developed system provides a practical and cost-effective alternative to traditional hardware laboratories while supporting future advancements in immersive engineering education.

4. Implementation Details

The proposed WebAR-based hardware learning platform was implemented using a modular architecture that integrates modern web technologies, real-time three-dimensional visualization, interactive learning mechanisms, and educational content delivery. The architecture was designed to establish seamless communication between the user interface, rendering engine, interaction modules, hardware management components, and information delivery systems. This modular design improves scalability, maintainability, and system responsiveness while supporting

immersive hardware learning experiences. The overall architecture of the proposed platform is illustrated in Figure 2.

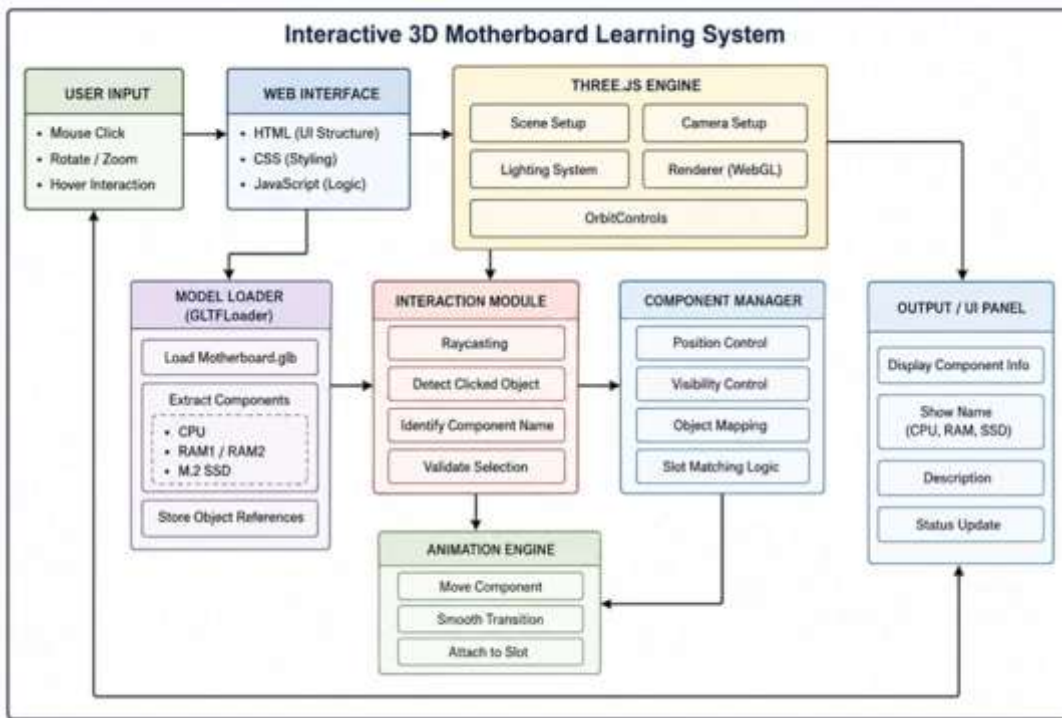


Figure 2. System Architecture of the Proposed Interactive WebAR Hardware Learning Platform

Figure 2 illustrates the architectural framework of the proposed WebAR-based learning platform. The system consists of interconnected modules responsible for user interaction, frontend rendering, hardware model management, object recognition, animation control, and educational information delivery. These modules work collaboratively to provide real-time hardware visualization and interactive learning capabilities within a browser-accessible environment.

4.1 Module Breakdown and Technical Details

4.1.1 User Interaction and Frontend Interface Layer

The user interaction layer serves as the primary communication channel between learners and the virtual learning environment. It captures user actions such as mouse clicks, touch gestures, zoom operations, rotation commands, and component inspection requests. These interactions are continuously monitored and translated into executable instructions that control the behavior of the three-dimensional environment.

The frontend interface was developed using React, TypeScript, HTML5, CSS, and JavaScript technologies [20]. A responsive design architecture ensures consistent functionality across desktop computers, tablets, and mobile devices. Navigation menus, control panels, educational popups, and information cards are organized to promote usability while minimizing unnecessary complexity. The interface dynamically updates according to user interactions, ensuring that relevant educational information is presented at appropriate stages of exploration.

Furthermore, responsive design principles enable automatic adaptation to different screen resolutions and aspect ratios, ensuring a consistent learning experience across multiple device categories.

4.1.2 Three.js Rendering Engine

The Three.js rendering engine serves as the visualization core of the proposed platform and manages all real-time three-dimensional graphics operations [16], [17]. This module is responsible for scene creation, camera configuration, lighting management, rendering execution, and interactive object visualization.

A dedicated WebGL scene is initialized to function as a virtual workspace for displaying hardware models. Perspective camera configurations are employed to provide realistic depth perception and natural viewing conditions.

Multiple lighting sources are strategically positioned within the scene to improve visibility and highlight structural details of hardware components.

OrbitControls are integrated to enable intuitive camera manipulation. Users can freely rotate, zoom, and inspect hardware models from different viewing angles, facilitating detailed examination of motherboard architectures and electronic components. Continuous rendering updates ensure smooth visual performance throughout the learning process.

4.1.3 Model Loader Infrastructure

The model loading subsystem manages the acquisition and preparation of three-dimensional assets within the virtual environment. Hardware models created and optimized in Blender [19] are exported in GLB format and subsequently loaded using the GLTFLoader framework provided by Three.js [16].

During the loading process, the system parses model hierarchies and identifies individual component structures such as processors, memory modules, storage interfaces, connectors, and expansion ports. Each component is assigned a unique reference identifier and stored within an internal object management structure, enabling rapid access during user interaction.

Asynchronous loading techniques are implemented to prevent interface blocking and improve overall responsiveness. This approach ensures efficient asset delivery while maintaining smooth user experiences, even when handling complex hardware models.

4.1.4 Interaction Module and Object Recognition Logic

The interaction module enables learners to communicate directly with virtual hardware components. This subsystem employs raycasting techniques to determine which object is selected within the three-dimensional environment.

Whenever a user performs a click or touch action, a virtual ray is projected from the camera toward the scene. Intersection calculations identify the targeted component and retrieve its associated metadata. The system then validates the selected object and determines the appropriate response according to the interaction context.

This mechanism supports accurate component identification, object highlighting, information retrieval, and educational content activation. Through real-time interaction processing, learners can actively explore motherboard architectures and gain detailed insights into individual hardware elements.

4.1.5 Component Management and Animation System

The component management module supervises object positioning, visibility control, hardware mapping operations, and assembly relationships. This subsystem maintains logical connections between motherboard structures and removable hardware components.

When a learner selects a component for detailed inspection, the system can isolate the selected object, temporarily remove it from the primary assembly, and display it independently for closer examination. Smooth transition animations are generated through interpolation techniques that gradually update object coordinates and orientations.

The animation engine additionally supports component restoration procedures, allowing hardware elements to return to their original positions without abrupt visual changes. These animated interactions improve user engagement while preserving spatial understanding of component placement and relationships.

4.1.6 Output and Educational Information Panel

The output interface functions as the educational knowledge delivery layer of the platform. When a hardware component is selected, contextual information is retrieved and presented within dynamically generated information panels.

The displayed content includes component names, technical descriptions, operational functions, hardware specifications, and educational explanations associated with the selected element. Information updates occur instantly

based on interaction events, ensuring that learners receive immediate feedback during exploration activities.

This contextual learning approach strengthens component recognition skills and enables learners to associate visual structures with their corresponding functionalities. Consequently, the platform transforms hardware observation into an interactive educational experience that promotes deeper understanding and practical knowledge acquisition.

4.2 System Workflow and Execution Flow

The operational workflow of the proposed platform follows a sequential process that connects user interaction with real-time visualization and educational content delivery.

1. The user accesses the application through a standard web browser.
2. The rendering environment and required resources are initialized.
3. A hardware model, such as a motherboard, Arduino Uno board, Raspberry Pi system, or drone flight controller, is selected.
4. The corresponding GLB asset is loaded into the Three.js environment.
5. Camera settings, lighting sources, and interaction controls are configured.
6. User interactions such as clicking, touching, rotating, and zooming are continuously monitored.
7. The raycasting engine identifies selected components and retrieves associated metadata.
8. Educational information is displayed through the output panel.
9. Components may be isolated, inspected, or restored as required by the learner.

This workflow creates a continuous cycle of exploration, interaction, and knowledge acquisition within a unified learning environment.

4.3 Marker-Based and Markerless AR Implementation

To improve accessibility and learning flexibility, the platform incorporates both marker-based and markerless Augmented Reality functionalities.

The marker-based mode utilizes predefined visual targets that are recognized through computer vision algorithms. When a valid marker is detected, the system aligns and overlays the selected three-dimensional hardware model onto the physical environment, enabling learners to observe components from multiple perspectives.

The markerless mode eliminates dependence on printed markers and enables direct placement of virtual hardware models within the surrounding environment. Using device camera inputs and spatial tracking mechanisms, learners can position, inspect, and interact with virtual objects in real time.

The integration of both AR modes increases platform flexibility and ensures broader accessibility across different learning environments, educational scenarios, and hardware accessibility conditions. The browser-based AR implementation was supported using modern WebAR technologies and AR frameworks [18].

4.4 Performance Optimization Techniques

Several optimization strategies were implemented to maintain smooth rendering performance across desktop and mobile platforms. Hardware models were optimized in Blender [19] through polygon reduction, mesh refinement, and material optimization techniques to minimize computational overhead while preserving visual accuracy.

Compressed GLB formats were adopted to reduce loading times and improve asset delivery efficiency. Asynchronous

loading mechanisms prevent interface blocking during model initialization, ensuring responsive user interactions. Efficient memory management techniques were incorporated to dispose of unused resources and prevent performance degradation during extended learning sessions. These optimizations were integrated within the Three.js rendering pipeline [16].

Furthermore, responsive rendering configurations dynamically adapt visualization parameters according to device capabilities and screen dimensions. These optimization measures contribute to stable system performance, reduced resource consumption, and a consistent user experience across a wide range of hardware environments.

5. Results and Discussion

The developed WebAR-based hardware learning platform was evaluated through functional testing, visualization assessment, interaction analysis, and cross-device compatibility testing to examine its effectiveness as an educational tool for computer hardware learning. The platform was successfully deployed within a browser-accessible environment and tested across desktop and mobile devices to evaluate usability, responsiveness, rendering performance, and interactive learning capabilities.

The evaluation results demonstrate that the proposed system effectively integrates three-dimensional visualization, component-level interaction, and Augmented Reality technologies within a unified educational framework. The browser-based architecture enabled smooth access to learning content without requiring dedicated software installation, thereby improving accessibility and ease of use.

5.1 Home Page Interface

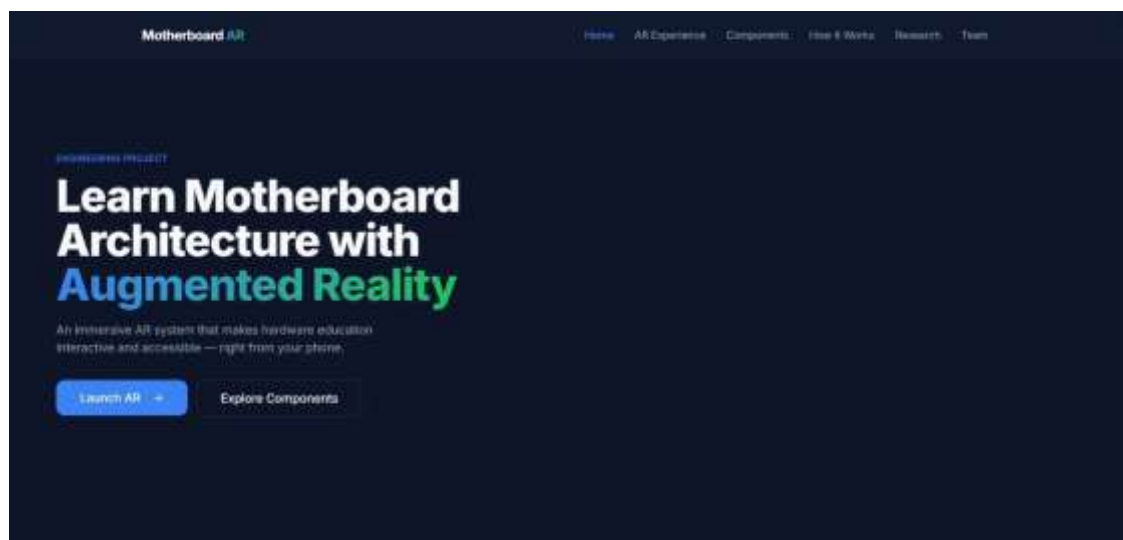


Figure 3. Home Page of the Proposed WebAR Hardware Learning Platform

Figure 3 illustrates the home page interface of the developed platform. The home page serves as the primary entry point to the learning environment and provides users with access to hardware visualization modules, AR experiences, and educational resources. The interface was designed using responsive web design principles to ensure accessibility across multiple devices.

The navigation structure allows learners to quickly access available functionalities without requiring prior technical expertise. Functional testing confirmed that users could efficiently navigate through the system and identify relevant learning modules with minimal guidance. The clean layout and intuitive design contribute to an improved user experience and support self-paced learning activities.

5.2 AR Mode Selection Interface

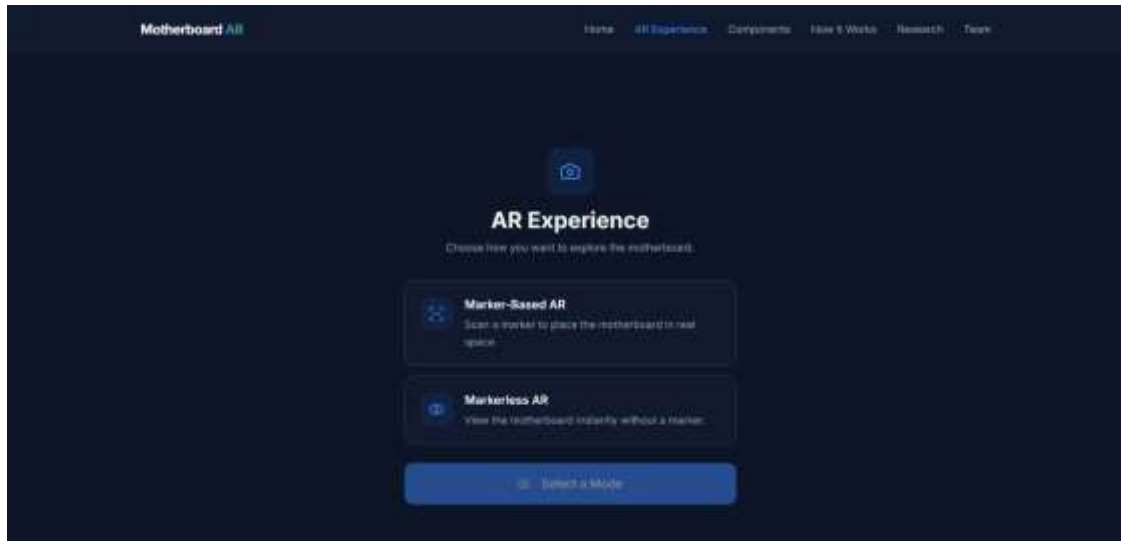


Figure 4. AR Mode Selection Interface Supporting Marker-Based and Markerless Learning Modes

Figure 4 presents the AR mode selection interface that enables learners to choose between marker-based and markerless learning experiences. The integration of both modes provides flexibility by accommodating different learning scenarios and device capabilities.

During system evaluation, both AR modes operated successfully and enabled accurate visualization of virtual hardware models within real-world environments. The availability of multiple interaction modes enhanced accessibility and ensured that learners could engage with educational content regardless of marker availability or environmental constraints. The mode selection process was completed seamlessly and demonstrated the practical integration of dual AR technologies within a single educational platform.

5.3 Interactive Motherboard Visualization Environment



Figure 5. Interactive 3D Motherboard Visualization and Component Exploration Environment

Figure 5 demonstrates the interactive motherboard visualization environment, which represents one of the core functionalities of the proposed platform. The optimized three-dimensional motherboard model was rendered successfully within the browser environment, allowing learners to inspect hardware structures from multiple viewing angles.

Interactive features such as zooming, rotation, component selection, and free camera navigation enabled detailed exploration of individual hardware components and their spatial relationships. When a component was selected,

contextual educational information was displayed in real time, allowing learners to understand its functionality, purpose, and role within the overall hardware architecture.

The visualization environment maintained stable rendering performance throughout testing sessions. Hardware elements including processors, memory modules, storage interfaces, connectors, and expansion slots remained clearly visible and accessible during interaction. The ability to freely explore component arrangements contributed significantly to improving spatial understanding and hardware recognition skills.

5.4 Discussion

The evaluation results indicate that the proposed WebAR platform successfully addresses several limitations associated with traditional computer hardware education. Unlike conventional instructional resources that rely heavily on static diagrams and limited laboratory demonstrations, the developed system enables learners to actively explore hardware structures through direct interaction and immersive visualization.

The integration of marker-based and markerless AR technologies enhances learning flexibility and supports diverse educational scenarios. Furthermore, browser-based accessibility eliminates software installation barriers and enables learning activities across desktop and mobile devices.

Another important outcome is the reduction of dependence on physical laboratory resources. Through realistic virtual representations and interactive educational guidance, learners can familiarize themselves with hardware components before participating in practical laboratory activities. This approach has the potential to improve preparedness, increase learner confidence, and reduce the likelihood of hardware misuse during hands-on training sessions.

Overall, the results demonstrate that the proposed WebAR learning platform provides an effective, scalable, and cost-efficient solution for computer hardware education. The combination of immersive visualization, interactive exploration, and browser-based accessibility contributes to a learning experience that is both engaging and educationally meaningful.

6. Conclusion and Future Work Conclusion

This research presented a WebAR-based Interactive Hardware Learning Platform designed to enhance computer hardware education through immersive visualization and interactive learning experiences. The proposed platform integrates browser-based three-dimensional visualization, component-level interaction, and both marker-based and markerless Augmented Reality technologies to create an accessible and engaging educational environment.

The developed system enables learners to explore motherboard architectures and related hardware platforms, including Arduino Uno boards, Raspberry Pi systems, and drone flight controllers, without requiring access to physical laboratory equipment. Through features such as component selection, isolation, real-time information retrieval, and interactive visualization, learners can develop a deeper understanding of hardware structures, component functionality, and spatial relationships.

A key advantage of the proposed platform is its browser-based architecture, which eliminates software installation requirements and supports accessibility across desktop and mobile devices. The integration of modern technologies such as React, Three.js, WebGL, and WebAR frameworks demonstrates the feasibility of delivering immersive educational experiences through standard web browsers while maintaining scalability and performance.

The results obtained from functional testing, interaction analysis, and visualization assessment indicate that the platform effectively supports hardware exploration, improves learner engagement, and provides a practical alternative to traditional laboratory-based learning environments. By bridging the gap between theoretical instruction and practical understanding, the proposed system contributes to the development of accessible, cost-effective, and technology-enhanced engineering education.

Future Work

Several opportunities exist for extending the capabilities of the proposed platform. Future development may focus on incorporating additional hardware platforms, including advanced motherboard architectures, networking devices, embedded systems, industrial control hardware, and Internet of Things (IoT) devices, to broaden the educational scope of the system.

The learning experience can be further enhanced through guided assembly simulations, interactive tutorials, assessment modules, progress tracking mechanisms, and performance analytics. These additions would support personalized learning pathways and enable educators to evaluate student progress more effectively.

Another promising direction involves the integration of Artificial Intelligence (AI) technologies to provide adaptive learning support. AI-driven recommendation systems and intelligent virtual tutors could deliver personalized guidance, answer hardware-related queries, and assist learners during practical exploration activities.

Future versions of the platform may also incorporate collaborative multi-user learning environments, Virtual Reality (VR) integration, voice-based interaction, and dedicated mobile application support. Such enhancements would contribute toward the development of a comprehensive digital learning ecosystem capable of supporting next-generation engineering education.

Overall, the proposed platform establishes a strong foundation for immersive hardware learning and demonstrates the educational potential of WebAR technologies. Continued

development and expansion of the system can further strengthen its role as a scalable, accessible, and technologically advanced solution for modern engineering education.

References

- [1] H. Lu and M. E. A. Hashim, "Augmented Reality in Education Applications: A Systematic Review," *International Journal of Modern Education*, 2025.
- [2] A. Kumar and S. Gupta, "Augmented Reality for Computer Hardware Learning: A Case Study," *International Journal of Computer Applications*, vol. 186, no. 4, pp. 22–28, 2024.
- [3] K. Patel and S. Mehta, "Recent Developments in Augmented Reality for STEM Education," *Journal of Science Education and Technology*, vol. 33, no. 1, pp. 55–70, 2024.
- [4] M. Radu, "Evaluating the Usability of Augmented Reality Systems for Education," *Computers & Education*, vol. 200, Art. no. 104115, 2024.
- [5] M. Bacca, S. Baldiris, R. Fabregat, S. Graf, and Kinshuk, "Augmented Reality Applications in Engineering Education," *IEEE Access*, vol. 12, pp. 7890–7905, 2024.
- [6] M. Akçayır and G. Akçayır, "Augmented Reality in Education: Current Trends," *Computers & Education*, vol. 210, pp. 104–120, 2024.
- [7] S. Alalwan and N. Alzahrani, "Augmented Reality in Education: A Systematic Review of Recent Advances," *Education and Information Technologies*, vol. 28, no. 2, pp. 1234–1250, 2023.
- [8] J. Lee and H. Kim, "The Effectiveness of Augmented Reality in Engineering Education: A Meta-Analysis," *IEEE Access*, vol. 11, pp. 45678–45690, 2023.
- [9] L. Wang and M. Chen, "User Experience in Educational Augmented Reality Applications: Recent Trends," *Computers & Education: Artificial Intelligence*, vol. 5, Art. no. 100123, 2023.
- [10] F. Silva and D. Costa, "Markerless Augmented Reality for Interactive Learning of Computer Components," *Computers & Education*, vol. 195, Art. no. 104789, 2023.
- [11] M. Rahman and T. Ahmed, "Augmented Reality-Based Laboratory for Computer Engineering Education," *Education Sciences*, vol. 13, no. 2, pp. 210–220, 2023.

- [12] M. Garzón, M. Pavón, and M. Baldiris, “The Impact of Augmented Reality on Education: A Meta-Review,” *Computers & Education*, vol. 153, pp. 103–110, 2023.
- [13] S. Yuen, G. Yaoyuneyong, and E. Johnson, “Development of an Augmented Reality System for Engineering Laboratory Education,” *International Journal of Engineering Education*, vol. 39, no. 3, pp. 500–510, 2023.
- [14] C.-H. Chen, C.-J. Su, and Y.-C. Chen, “An Augmented Reality System for Learning Computer Hardware Assembly,” *International Journal of Emerging Technologies in Learning*, vol. 18, no. 1, pp. 100–110, 2023.
- [15] S. K. Ong, M. L. Yuan, and A. Y. C. Nee, “Augmented Reality for Computer Hardware Assembly Training,” *Procedia Computer Science*, vol. 207, pp. 123–130, 2023.
- [16] R. Cabello, P. Houx, G. Zaal, and Three.js Contributors, *Three.js: JavaScript 3D Library and Documentation*. Three.js Foundation, 2025.
- [17] Khronos Group, *WebGL Specification and Programming Guide for Interactive 3D Graphics on the Web*, Version 2.0. Khronos WebGL Working Group, 2025.
- [18] H. Kim, *MindAR: Browser-Based Augmented Reality Framework Documentation*. MindAR Project Documentation, 2025.
- [19] Blender Foundation, *Blender 3D Creation Suite: Official Documentation and User Manual*. Amsterdam, The Netherlands: Blender Foundation, 2025.
- [20] Meta Platforms Inc., *React: A JavaScript Library for Building User Interfaces – Official Documentation*. React Development Team, 2025.