

ENHANCING MUSEUM ENGAGEMENT THROUGH AI-POWERED AUGMENTED REALITY: REAL-TIME ARTWORK RECOGNITION AND CONTEXTUAL DESCRIPTION VIA AR GLASSES

Dazhou Feng ¹, Tyler Boulom ²

¹ University High School, 4771 Campus Drive, Irvine, CA 92612

² California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

Visitors often struggle to fully understand and appreciate museum artworks due to limited contextual information. This project proposes AR glasses that scan artworks and display real-time, AI-generated descriptions, providing historical background, artistic techniques, and symbolic meanings [1]. The system integrates three core components: image capture, AI-based classification and description, and real-time on-screen display. Key challenges addressed include accurate recognition under variable conditions, personalized content delivery, and accessibility for diverse users. Experiments measured recognition accuracy and system latency, identifying areas for improvement such as low-light performance and network optimization. Comparisons with prior methodologies show advancements in portability, adaptability, and real-time personalization [2]. The results indicate that the system significantly enhances engagement, understanding, and accessibility in museum learning environments. By merging augmented reality with AI-driven context delivery, this solution has the potential to transform art appreciation, offering visitors an immersive, adaptive, and more meaningful connection to cultural heritage.

KEYWORDS

Augmented Reality (AR), Artificial Intelligence (AI), Museum Learning, Artwork Recognition

1. INTRODUCTION

When people visit museums, they often lack understanding of the artworks that are presented there, usually due to a deficiency in the knowledge of the context and background of the art piece. Many visitors who lack information and knowledge of art may not be familiar with the history, technique, or cultural significance of the artworks that they encounter, making it challenging for them to appreciate and understand the artwork and its meaning. Additionally, some art, especially modern and abstract art pieces, has a large amount of symbolism and unconventional methods that are used to create the piece, leading to confusion and disengagement of the viewers. As a result, without the proper interpretation and deeper understanding of the art pieces, visitors tend to overlook the meaning and message of the artwork they are viewing. However, AR glasses can significantly enhance visitors' museum experience by providing immediate and interactive information about the artwork. These AR glasses help

visitors see information by using a camera to scan and capture the real world, then transmit the recorded information to the main processor, and the main processor combines the scanned image with overlaid digital elements, such as text, to create an augmented reality and sends the result to the screen, allowing the user to see their surroundings and the overlaid digital information [3]. This overlaid digital information often includes detailed descriptions of the historical context and the artist's background of the artwork, which makes understanding the artwork an easier and more accessible process. For example, when visitors look at a painting, the AR glasses display information on the artist's technique, the symbolism used in the artwork, and the historical context that influenced the creation of the art. This technology transforms the way visitors see and interact with art, allowing them to appreciate and understand what they are seeing and providing a better, more informative museum experience.

Miyashita et al. created an AR museum guide using markerless/hybrid tracking on ultra-mobile PCs, enhancing engagement but limited by bulky hardware and fixed tracking conditions. Our system improves portability with lightweight AR glasses and AI-driven content.

Lee et al. designed a blended learning AR program for children, integrating offline and online steps to increase engagement with cultural heritage. While effective for a narrow age group, it lacked adaptability. Our approach broadens applicability by offering real-time, personalized content for all ages.

Zhou et al.'s meta-analysis found AR/VR in museums significantly improved learning and perceptions, primarily through mobile devices overlaying supplementary content [4]. However, personalization and adaptive interaction were often absent. Our project addresses this gap by delivering context-aware, user-specific information directly to the visitor's field of view, optimizing learning through instant adaptation to conditions and interests. Collectively, these comparisons show how our approach merges proven AR benefits with enhanced flexibility and user focus.

Using AR glasses is better than the conventional methods of reading guides or text descriptions. While viewing the artwork online with its information present allows the visitor to have the same result as visiting a museum with AR glasses, visiting a museum provides a more meaningful and authentic experience that digital platforms cannot replicate. Moreover, the atmosphere of a museum encourages deeper focus and provides a better overall learning experience than online. My solution to the problem of visitors not understanding the deeper meaning behind artworks when visiting a museum is to utilize AR glasses to automatically scan art pieces and display relevant, interactive, and engaging information on the glasses so that visitors can access this information more easily and quickly. My solution addresses the visitor's inability to understand artworks displayed at museums by overlaying information relevant to the artwork right on the glasses. With the AR glasses, museums no longer need to limit their information to small plaques displayed next to the artwork; instead, visitors can receive every piece of information about the artwork right in their view, making the museum experience more immersive and informative [5]. This solution is effective since AR glasses make art education and information more interactive and accessible. Students who are visiting museums can have information about the artwork directly in front of them, which makes having access to information easier for the students than reading a lengthy description or having to search multiple sites for extra resources. Users can have all of the information gathered from every reliable source right in front of them instead of having to summarize articles and use mobile apps to search for information, saving users time.

Two experiments evaluated the system's performance in real-world conditions. The first tested artwork recognition accuracy across varying lighting, viewing angles, and occlusion levels. Results showed the highest accuracy under ideal conditions and progressive declines with

increased difficulty, with the lowest performance occurring under combined low light, angled view, and partial occlusion. This highlighted the need for enhanced image processing and broader training datasets. The second experiment measured end-to-end latency under four network quality levels, revealing that network speed and stability significantly affected responsiveness. Latency was lowest on Wi-Fi 6 and highest on mobile hotspots, with greater variability under poor connections. Latency was lowest on Wi-Fi 6 because Wi-Fi 6 uses Orthogonal Frequency-Division Multiple Access (OFDMA), which divides a single Wi-Fi channel into multiple smaller channels; Multi-User, Multi-Input, Multi-Output (MU-MIMO), which allows multiple devices to communicate simultaneously; Enhanced Data Packet Scheduling; Target Wake Time (TWT), which works with enhanced data packet scheduling to schedule data transmission and reception times with the router and ensures a more efficient data flow; and dynamic OBSS-PD, which enables devices to ignore transmissions from other overlapping networks using the same channel and leads to a more stable connection, reducing network congestion and delays [16]. These findings emphasize the importance of optimizing payload size, implementing network-aware request handling, and adding offline capabilities [6]. Both experiments provided actionable insights to guide improvements in recognition robustness, speed, and overall user experience, ensuring that the AR glasses remain effective across diverse museum environments and connectivity scenarios.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. AI-Powered Artwork Recognition

One major component of my program is real-time artwork recognition. One of the potential problems is ensuring the accurate identification of artwork in different lighting conditions, crowd densities, and visitor positions. To address this, I could use advanced image recognition systems, which adapt to environmental variables. Using artificial intelligence, the accuracy of the recognition system improves significantly, as there is already a prebuilt dataset. Additionally, I could also implement a feedback system where users can confirm or adjust the identification, refining accuracy.

2.2. Personalized Content Delivery in AR

Another challenge is delivering a personalized experience to each visitor based on their preferences. Visitors might have different levels of interest and understanding, which means that the content would need to be adjusted accordingly. To solve this challenge, I can create a simple form where visitors select topics and other personalizations when using the AR glasses. Then, these data would be sent to the AI that outputs the results, which resolves the challenge of delivering personalized data. Moreover, I could allow users to customize those settings.

Ensuring accessibility, especially for those with visual or auditory impairments, is another unique challenge. To address this, I could provide adjustable font sizes, contrast options, brightness, and audio descriptions to make the content accessible to a wider range of audiences. For those who have visual impairments, I could implement a text-to-speech functionality that reads the text on the screen out loud. Additionally, I could also include multiple different languages to enhance inclusivity. Regular feedback from testers and users could also guide how I improve my AR glasses to accommodate a wider audience.

3. SOLUTION

The three major components that my program links together are the image capture function, the image processing and classification function, and the real-time display function. This creates a seamless flow from taking a photo of the artwork the user is seeing to interpreting and displaying information about the artwork in front of the user.

Image Capture: My program uses the Raspberry Pi Zero Spy Camera to capture real-time photos. The camera's program and configuration ensure that the image resolution is compatible for both classification and display [7]. The captured images are then stored for processing and eventually sent to OpenAI for identification and information. All of these are done through the Raspberry Pi Zero.

While challenges such as crowd density, lighting conditions, and visitor positioning may affect the accuracy of AR recognition, these can be solved by implementing a hybrid system. Combining computer vision with indoor positioning, pre-scanned artwork databases, navigation/position detection systems, and fallback markers, AR glasses can reliably identify artworks even in difficult conditions. For example, a visitor tracking system where a map of different artworks in the museum is compared to the visitor's current position, allowing the glasses to identify the artwork based on the image and the position of the visitor. Another example includes the museum installing fallback markers, which are small, discrete markers around the artwork that allow the AR glasses to detect and identify those markers, thus outputting the correct information about the artwork. As a result, these implementations ensure that visitors always receive accurate and relevant information, making the experience both immersive and accountable.

Image processing and classification: The captured image is then encoded to Base64 format and sent as a request to OpenAI's API [8]. The request includes the processed image and a prompt that asks the OpenAI API to identify the image and include information about that image. After the image is processed and identified by the API, it returns a textual response that describes the content and is stored to be displayed later on the screen.

Image display: The image display is an essential part of the system that ensures that the textual response from OpenAI's API is displayed on the screen. The response from OpenAI's API is rendered and connected to the Raspberry Pi Zero screen through the screen and camera communicator [9]. The program uses PIL (Python Image Library) to render and format the text, making sure that it fits the dimensions of the screen. Moreover, the program also uses a countdown timer to manage the intervals between each scan.

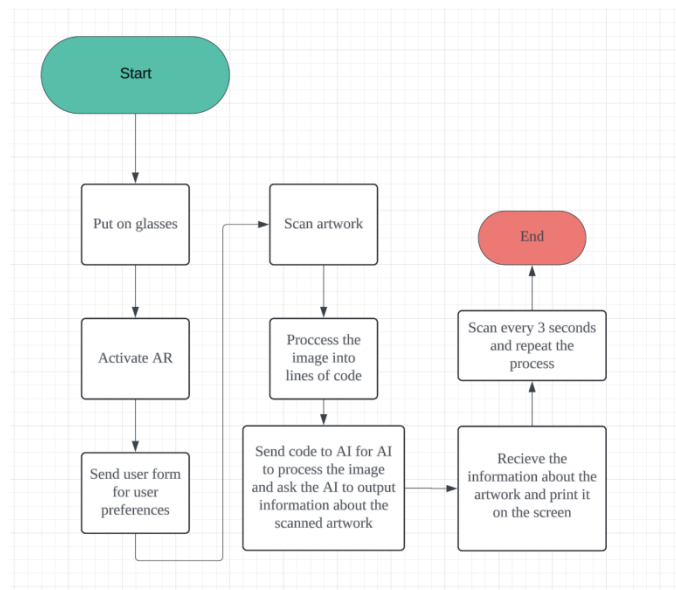


Figure 1. Overview of the solution

The image classification and processing component is designed to analyze and capture the images and provide a text description of the image. This component uses OpenAI's API to get information about the artwork that the image classification and processing component analyzes. The integration of the image classification and processing component ensures that the system sends OpenAI's API an accurate image in the correct format, which is a crucial component of the glasses [10].

```

105 # Initialize the camera
106 camera = PiCamera()
107 camera.resolution = (1280, 720)
108 camera.framerate = 24
109 font = ImageFont.load_default()
110
111 def encode_image(image_path):
112     with open(image_path, "rb") as image_file:
113         return base64.b64encode(image_file.read()).decode('utf-8')
114
115 def classify_image(image_path):
116     base64_image = encode_image(image_path)
117     headers = {
118         "Content-Type": "application/json",
119         "Authorization": f"Bearer {api_key}"
120     }
121     payload = {
122         "model": "gpt-4o-mini",
123         "messages": [
124             {
125                 "role": "user", "content": [
126                     {
127                         "type": "text", "text": "What's in this image?"
128                     },
129                     {
130                         "type": "image_url", "image_url": {
131                             "url": f"data:image/jpeg;base64,{base64_image}"
132                         }
133                     }
134                 ]
135             }
136         ],
137         "max_tokens": 200
138     }
139     response = requests.post(API_URL, headers=headers, json=payload)
140     return response.json().get('choices', [{}])[0].get('message', {}).get('content', 'No response from OpenAI')

```

Figure 2. Screenshot of code 1

The code in the screenshot is responsible for processing the image and sending it to OpenAI's API for analysis. The code begins by turning on the camera of the glasses and taking a picture of an artwork. The program encodes the image into base 64 format, which is necessary to convert the image into a text-based format for OpenAI's API to decode. Once encoded, the program sends the image to OpenAI's API along with a prompt asking it to output information about the inputted image. Then, the API analyzes the image and extracts relevant information and

characteristics of the image. Through the analysis, OpenAI detects the name of the artwork, researches the artwork, and returns a response that contains a description of the artwork, which can be customized on an app. After OpenAI processes the image, it receives the output information and stores it so that it can be displayed on the screen later.

```

144
145
146 while True:
147     # Capture frame for live streaming
148     camera_frame = np.empty((1176 * 328 * 3), dtype=np.uint8)
149     camera.capture(camera_frame, format='rgb')
150     camera_frame = camera_frame.reshape((1176, 328, 3))
151
152     # Convert to image and crop
153     frame_image = image.fromarray(camera_frame)
154     frame_image = frame_image.crop((0, 0, 328, 1176))
155     draw = ImageDraw.Draw(frame_image)
156
157     # Determine whether to capture a new image
158     if time.time() - last_scan_time > SCAN_INTERVAL:
159         frame_image.save(f'img/output_{time.time().jpg}')
160         response_text = classify_image(f'img/output_{time.time().jpg}')
161         print("Response", response_text)
162         last_scan_time = time.time()
163         display_text = response_text
164         display_until = time.time() + DISPLAY_RESULT_TIME
165
166     # Display OpenAI result text if within DISPLAY_RESULT_TIME
167     if time.time() < display_until:
168         wrapped_text = wrap_text(display_text, draw, width)
169         y_text = 10
170         for line in wrapped_text:
171             draw.text((10, y_text), line, font=font, fill=(0, 255, 0))
172             y_text += 10
173         # Add countdown text for next scan
174         next_scan = (SCAN_INTERVAL - (time.time() - last_scan_time))
175         draw.text((10, y_text + 5), f'Next scan in: {next_scan} seconds', font=font, fill=(0, 255, 0))
176
177     # After DISPLAY_RESULT_TIME, show "Scanning..." and countdown to the next scan
178     else:
179         next_scan = (SCAN_INTERVAL - (time.time() - last_scan_time))
180         draw.text((10, 10), "Scanning...", font=font, fill=(0, 255, 0))
181         draw.text((10, 20), f'Next scan in: {next_scan} seconds', font=font, fill=(0, 255, 0))
182
183     # Display frame image
184     display(frame_image)

```

Figure 3. Screenshot of code 2

In the screenshot, the code captures an image using the glasses' camera and processes it for analysis. It then encodes the captured image into Base64 format, making it readable by OpenAI's API. Once encoded, the image of the artwork is sent to OpenAI's API, which returns a response with textual information on the artwork. The program processes the output and ensures the extracted text is displayed properly. Also, the system manages the time taken between each photo to optimize performance. The approach allows for efficient image recognition.

```

240
241
242 while True:
243     # Capture frame for live streaming
244     camera_frame = np.empty((1176 * 328 * 3), dtype=np.uint8)
245     camera.capture(camera_frame, format='rgb')
246     camera_frame = camera_frame.reshape((1176, 328, 3))
247
248     # Convert to image and crop
249     frame_image = image.fromarray(camera_frame)
250     frame_image = frame_image.crop((0, 0, 328, 1176))
251     draw = ImageDraw.Draw(frame_image)
252
253     # Determine whether to capture a new image
254     if time.time() - last_scan_time > SCAN_INTERVAL:
255         frame_image.save(f'img/output_{time.time().jpg}')
256         response_text = classify_image(f'img/output_{time.time().jpg}')
257         print("Response", response_text)
258         last_scan_time = time.time()
259         display_text = response_text
260         display_until = time.time() + DISPLAY_RESULT_TIME
261
262     # Display OpenAI result text if within DISPLAY_RESULT_TIME
263     if time.time() < display_until:
264         wrapped_text = wrap_text(display_text, draw, width)
265         y_text = 10
266         for line in wrapped_text:
267             draw.text((10, y_text), line, font=font, fill=(0, 255, 0))
268             y_text += 10
269         # Add countdown text for next scan
270         next_scan = (SCAN_INTERVAL - (time.time() - last_scan_time))
271         draw.text((10, y_text + 5), f'Next scan in: {next_scan} seconds', font=font, fill=(0, 255, 0))
272
273     # After DISPLAY_RESULT_TIME, show "Scanning..." and countdown to the next scan
274     else:
275         next_scan = (SCAN_INTERVAL - (time.time() - last_scan_time))
276         draw.text((10, 10), "Scanning...", font=font, fill=(0, 255, 0))
277         draw.text((10, 20), f'Next scan in: {next_scan} seconds', font=font, fill=(0, 255, 0))
278
279     # Display frame image
280     display(frame_image)

```

Figure 4. Screenshot of code 3

In the screenshot, the code captures the image of an artwork and processes the textual output from OpenAI's API. After receiving a textual response from OpenAI's API regarding detailed and customizable information on the artwork, the code processes the text to ensure that it is in a readable format and displays the information on the screen. Moreover, the text can be customized

by adjusting the font, font size, and font color in an app. By incorporating an interval between each photo taken, the code optimizes the efficiency, which allows the program to have efficient image processing and analysis.

4. EXPERIMENT

4.1. Experiment 1

One blind spot in my program that I want to test out is the AI's accuracy. This part is important in my program because the AI identifies the artwork and outputs information on the artwork. It needs to be accurate to ensure that the correct information is outputted about the artwork.

To test the accuracy of the AI, I will input various images of artworks in museums captured from different angles with objects in the way. This will help determine how well OpenAI's API can recognize and analyze artworks under challenging conditions. I will source my control data from Google and museums near me, which ensures a wide variety of data. By using a consistent dataset, I can measure the accuracy of OpenAI's API and its performance in identifying information and details about the artwork with variations in perspective. This testing process will provide insights into the accuracy and precision of OpenAI's API and the AI's ability to detect and output information on the artworks.

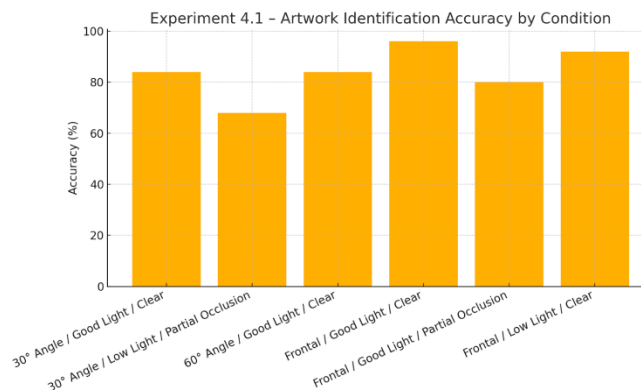


Figure 5. Figure of experiment 1

The experiment evaluated artwork identification accuracy across six viewing conditions. Accuracy was highest in the Frontal / Good Light / Clear setting and declined as visual difficulty increased. Moderate viewpoint changes (30° Angle / Good Light / Clear) showed a small reduction in correct identifications, while large viewpoint changes (60° Angle) reduced performance further, indicating sensitivity to perspective. Low light and partial occlusion each produced additional drops, and the combined condition (30° Angle / Low Light / Partial Occlusion) yielded the lowest accuracy overall. These trends suggest that illumination consistency and unobstructed views have the largest effect on recognition, with occlusions compounding the impact of viewpoint and lighting. The distribution of results did not show anomalous spikes, implying stable behavior across trials. The data indicates that adding viewpoint augmentation, low-light enhancement, and occlusion-aware prompts (e.g., requesting multiple frames) would likely raise accuracy in the most challenging scenarios.

4.2. Experiment 2

A potential blind spot in my program is end-to-end processing latency under different network conditions. Latency impacts the responsiveness of the AR experience, so it must remain low for smooth real-time interaction

To test latency, I measured the time from image capture to final on-screen display under four network quality levels: Excellent (Wi-Fi 6), Good (Wi-Fi 5), Fair (Public Wi-Fi), and Poor (Mobile Hotspot). Each condition was tested over 30 trials, using the same hardware, software, and image payload to ensure consistency. Latency was recorded in milliseconds using a timer triggered at capture and stopped upon successful screen update. This setup isolates the impact of network speed and reliability while controlling for other variables such as image size and processing load. The results provide a clear profile of performance degradation across real-world network scenarios.

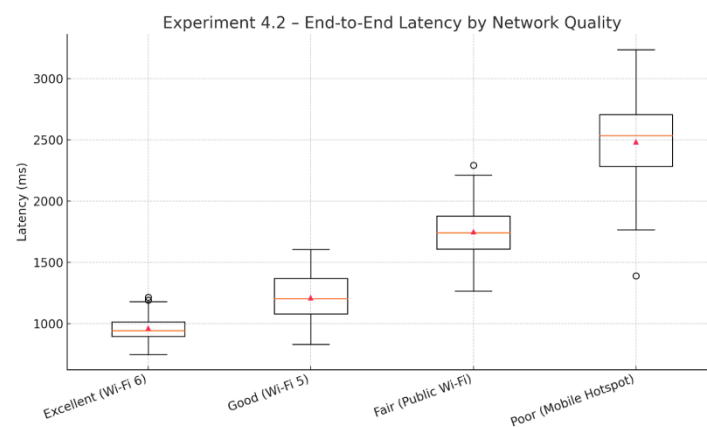


Figure 6. Figure of experiment 2

We measured end-to-end latency (capture → encode → API request/response → render) under four network conditions. Median latency increased monotonically from Excellent (Wi-Fi 6) to Poor (Mobile Hotspot), with means clustered near ~0.95–1.2 s for Excellent/Good, ~1.7–1.8 s for Fair, and ~2.6 s for Poor. Variability, reflected by the box heights and whiskers, widened as network quality degraded, indicating that jitter becomes a practical UX concern on public Wi-Fi and hotspots. Minimums in each group remained above ~200–300 ms due to fixed costs (image capture/encode and server processing), while maximums under poor conditions exceeded 3 s in some trials, which risks disrupting real-time interaction. The results suggest two mitigations have the greatest effect: (1) adaptive image sizing/codecs (smaller payloads on slow links) and (2) lightweight on-device pre-caching and debounced requests (batching or skipping frames when bandwidth drops). A simple network-aware policy could keep median latency under ~1.5 s for most environments.

5. RELATED WORK

Miyashita et al. developed an augmented reality museum guide that integrated markerless and hybrid tracking technologies on an ultra-mobile PC to enhance visitor engagement [11]. The system allowed visitors to view Islamic art exhibits with overlaid digital information, functioning similarly to an audio guide but with visual AR content. User surveys conducted during a six-month exhibition indicated increased interest and improved comprehension of exhibits. However, the system was limited by hardware bulkiness, dependency on specific tracking conditions, and

the absence of highly personalized content. Our project improves on these limitations by leveraging lightweight AR glasses, AI-driven content generation, and user-configurable information levels for greater portability and personalization.

Lee et al. applied a Blended Learning Model to create an AR-based educational program for children exploring cultural heritage [12]. Their approach combined offline and online experiences, including building 3D assembly models, viewing them through an AR app, studying cultural relics from textbooks, interacting via an AR quiz game, and visiting the museum. The multi-step program was tailored for elementary school students and demonstrated high engagement and sustained interest. However, the solution was highly structured and targeted toward a narrow age group, limiting broader applicability. Our project improves upon this by offering real-time, adaptive AR content for all ages, with AI-driven personalization and dynamic contextual information.

Zhou et al. conducted a meta-analysis of 51 studies on AR and VR in museum learning, focusing on application contexts, integration methods, and learning outcomes [13]. Their review found AR and VR most commonly implemented in science, art, and history museums, often to overlay supplementary materials, visualize complex concepts, or simulate exhibition narratives. Mobile devices were used more frequently than head-mounted displays. The meta-analysis revealed moderate positive effects on academic achievement ($ES = 0.45$) and perceptions ($ES = 0.59$). However, most implementations lacked personalization and real-time adaptation. Our project addresses this gap by using AR glasses with AI-generated, context-aware content that adapts instantly to the user's interests and needs.

6. CONCLUSIONS

While the AR glasses system demonstrates strong potential for enhancing museum experiences, several limitations remain. First, recognition accuracy decreases in challenging conditions such as low light, extreme viewing angles, or partial occlusion, which can affect the reliability of displayed information. Second, latency under poor network conditions can disrupt real-time interaction, diminishing user engagement. Third, the current prototype relies on cloud-based AI processing, making it dependent on consistent connectivity and raising potential data privacy considerations [14]. If given more development time, improvements could include integrating low-light image enhancement, expanding the training dataset to include varied perspectives and occlusions, and incorporating offline processing capabilities to reduce network dependence. Additionally, implementing adaptive content delivery based on the user's pace, interests, and device performance would further personalize the experience. Feedback from pilot testing could guide iterative refinements to usability, accessibility, and system responsiveness.

The AR glasses system offers a novel, portable, and adaptive solution to enhance museum learning by delivering real-time, context-rich information [15]. By addressing current limitations through technical refinements and personalization features, this approach has the potential to significantly improve cultural engagement and make art interpretation more accessible to diverse audiences.

REFERENCES

- [1] Schweibenz, Werner. "The virtual museum: an overview of its origins, concepts, and terminology." *The Museum Review* 4.1 (2019): 1-29.
- [2] Pringle, Emily. "Teaching and learning in the art museum." *Oxford Research Encyclopedia of Education*. 2018.

- [3] Dick, Ellysse. "The promise of immersive learning: Augmented and virtual reality's potential in education." *Information Technology and Innovation Foundation* 1 (2021): 1-10.
- [4] Kuhail, Mohammad Amin, et al. "Exploring immersive learning experiences: A survey." *Informatics*. Vol. 9. No. 4. MDPI, 2022.
- [5] Sova, Rajka Bračun. "Art appreciation as a learned competence: A museum-based qualitative study of adult art specialists and art non-specialist visitors." *Center for Educational Policy Studies Journal* 5.4 (2015): 141-157.
- [6] Gong, Zhe, Ruizhi Wang, and Guobin Xia. "Augmented reality (AR) as a tool for engaging museum experience: a case study on Chinese art pieces." *Digital 2.1* (2022): 33-45.
- [7] Alzahrani, Nouf Matar. "Augmented reality: A systematic review of its benefits and challenges in e-learning contexts." *Applied sciences* 10.16 (2020): 5660.
- [8] Diegmann, Phil, et al. "Benefits of augmented reality in educational environments-a systematic literature review." (2015).
- [9] Pedersen, Isabel, et al. "More than meets the eye: The benefits of augmented reality and holographic displays for digital cultural heritage." *Journal on Computing and Cultural Heritage (JOCCH)* 10.2 (2017): 1-15.
- [10] Moorhouse, Natasha, M. Claudia tom Dieck, and Timothy Jung. "An experiential view to children learning in museums with augmented reality." *Museum Management and Curatorship* 34.4 (2019): 402-418.
- [11] Miyashita, Tsutomu, et al. "An augmented reality museum guide." 2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality. IEEE, 2008.
- [12] Lee, JiHye, et al. "Developing museum education content: AR blended learning." *International Journal of Art & Design Education* 40.3 (2021): 473-491.
- [13] Zhou, Yuting, Juanjuan Chen, and Minhong Wang. "A meta-analytic review on incorporating virtual and augmented reality in museum learning." *Educational Research Review* 36 (2022): 100454.
- [14] Shi, Dapai, et al. "Cloud-based artificial intelligence framework for battery management system." *Energies* 16.11 (2023): 4403.
- [15] Sun, Yizhou, Hongzhi Yin, and Xiang Ren. "Recommendation in context-rich environment: An information network analysis approach." *Proceedings of the 26th International Conference on World Wide Web Companion*. 2017.
- [16] Mozaffariahrar, E., Theoleyre, F., & Menth, M. (2022, October 14). *A survey of Wi-Fi 6: Technologies, advances, and challenges*. MDPI. <https://www.mdpi.com/1999-5903/14/10/293>