MELODYCANVAS: DESIGN AND EVALUATION OF A VIRTUAL REALITY PLATFORM FOR ACCESSIBLE ARTS EDUCATION, COMMUNITY ENGAGEMENT, AND DIGITAL GALLERIES

Larry Yuheng Li [1], Tyler Boulom [2]

[1] Sage Hill School, 20402 Newport Coast Drive, Newport Beach, CA92657
[2] Woodbury University, 7500 N Glenoaks Blvd, Burbank, CA 91504

ABSTRACT

This project addresses the global decline in adolescent access to arts education by proposing MelodyCanvas, a virtual reality platform that enables students to share artwork, view galleries, and engage with a creative community. The system integrates Firebase authentication, AI-based content validation, and immersive gallery environments to create an accessible and interactive artistic space. Core challenges included building reliable 3D environments, ensuring accurate originality checks, and maintaining acceptable load times when retrieving artwork from cloud storage. These challenges were examined through experiments measuring AI classification accuracy and gallery performance. Findings revealed strong baseline functionality but identified the need for improved plagiarism detection methods and optimized loading pipelines. Compared with existing methodologies in vision analysis and VR interface design, MelodyCanvas offers a hybrid approach that balances technical feasibility with user experience. Ultimately, the platform demonstrates a promising way to support artistic development and cultural connection through accessible digital tools.

KEYWORDS

Virtual Reality, Arts Education, Creative Communities, AI Content Validation

1. INTRODUCTION

In recent decades, artistic opportunities and exposure for adolescents have been declining significantly. Both in and out of school, factors including economic recessions, pandemics, and government-issued educational policies have reduced the focus on arts and painting education. Schools worldwide have terminated many art programs in pursuit of better academic rankings, and galleries have shut down because of pandemic-related restrictions, limiting opportunities for adolescents to develop creativity and cultural connection [1][2].

Arts education is pivotal for adolescent development, providing opportunities for expression, emotional processing, creativity, and critical thinking. Bowen and Kisida's study of arts programs in Houston found that arts participation improves writing achievement and emotional empathy and increases engagement in other subjects [3]. Additional studies indicate that arts education promotes cognitive benefits and cultural awareness [4]. College Board data similarly reports that students who take four years of arts in high school score an average of 92 points higher on the SAT than peers with minimal arts education [5].

David C. Wyld et al. (Eds): NLTM, CCNET, DBDM, AIMLA, CICS – 2026
pp. 75-85, 2026. CS & IT - CSCP 2026 DOI: 10.5121/csit.2026.160607

However, these benefits are increasingly inaccessible. The decline disproportionately impacts on minority and low-income communities, which already had limited access. Between the 1980s and 2000s, African American and Hispanic students experienced reductions in arts education ranging from 40 to 50 percent [6]. Educational reforms such as the No Child Left Behind Act (2001) intensified this issue by shifting focus to math and English test scores; 44 percent of districts reported cutting time from arts and other subjects by an average of 145 minutes per week [7].

Finally, the broader art ecosystem is also contracting. Following the 2020 pandemic, global art gallery sales declined by 36 percent, with small galleries seeing declines of nearly 50 percent, signaling potential closures and further reducing access to artistic spaces [2].

Method A: Research on deep-learning similarity embeddings provides a highly accurate framework for detecting near-duplicate images. Its main limitation is the computational cost of indexing large datasets. MelodyCanvas improves upon this by combining automated checks with contextual metadata and human oversight.

Method B: Surveys of image plagiarism-detection techniques show that no single approach can reliably evaluate ownership or similarity in visual media, especially when transformations are applied. MelodyCanvas addresses these gaps by incorporating multiple validation steps rather than relying solely on automated detection.

Method C: Studies on VR loading interfaces demonstrate that interactive loading elements significantly improve user experience during long wait periods. MelodyCanvas expands on this insight by planning to integrate interactive UI components alongside backend optimizations to maintain immersion in VR galleries.

My solution to this problem is a cost-free, online platform which will give students access to arts resources and a community of like-minded peers. This program is called MelodyCanvas. Made possible with Unity, Microsoft's Azure Ai and experienced through Metaquest VR headsets, MelodyCanvas aims to foster creativity and cultural connection by serving as a medium for maturing adolescents to share their own artworks as well as view the works of others through immersive, VR galleries.

Upon opening MelodyCanvas, players can create their own profiles through a signup page, which are managed through Google's Firebase, or login to an existing one. These profiles allow players to create their own galleries or explore public galleries created by others, promoting cultural connections and emotional empathy as players get to experience and see how their peers express themselves through their artworks. Moreover, players also have the option to populate their own galleries by uploading their paintings through an upload page mediated by a content checker. This checker is connected to Azure AI and will check for originality of the artwork, promoting integrity and creativity.

Unlike artist forums or traditional social medium platforms like Instagram or Youtube, MelodyCanvas is a more realistic and immersive experience: allowing players to walk through galleries as if in real life, interact with paintings (a bonus feature adding more interactiveness than traditional galleries), as well as create their own galleries, all from the comfort of their own homes.

Two primary experiments were conducted to evaluate potential weaknesses in MelodyCanvas. The first experiment assessed the reliability of the AI content checker by testing it against three categories of images: original artworks, copyrighted pieces, and stylistically similar images. The experiment used a controlled dataset and measured approval rates to determine whether the system correctly classified originality. Results revealed strong performance for clear cases but inconsistent handling of ambiguous submissions, indicating the need for more advanced similarity models.

The second experiment examined gallery performance by measuring loading times under different gallery sizes and user configurations. Tests conducted across multiple trials demonstrated predictable scaling behavior, with more complex galleries requiring significantly longer load times. Mixed-user galleries showed the slowest performance due to increased Firebase queries and image downloads. These results confirmed that network activity, rather than rendering, was the primary factor affecting performance. Together, the experiments highlighted key areas for system refinement: improved plagiarism detection and enhanced loading efficiency.

2. CHALLENGES

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

2.1. 3D Environment Design and Immersion

Designing the 3D environments for MelodyCanvas presents an early challenge, as these spaces determine how immersive and engaging the user experience will be. Standard Unity primitives provide only basic geometry, which is insufficient for constructing detailed, realistic gallery interiors. VR interface design research emphasizes that spatial layout, navigability, and environmental clarity directly influence user immersion and comfort, requiring environments that are both visually coherent and structurally optimized [14]. To address this, higher-fidelity models can be created using external tools such as Blender and later imported into Unity. For efficiency, modular in-engine tools like UModeler can be used to construct common architectural elements without dramatically increasing development time.

2.2. AI Content Moderation Accuracy

The performance of the AI content checker is an essential part of maintaining originality and safety on the platform. One challenge is determining which Azure service and which visual features are most reliable for identifying plagiarized or inappropriate imagery. Computer-vision literature shows that image-analysis systems can struggle when visual content includes high texture variation, non-standard lighting, or stylistic transformations [13]. To manage these complexities, the checker must be trained using a diverse dataset and tuned to flag logos, brand marks, and distinctive visual patterns. Establishing clear detection thresholds and fallback manual review processes would be necessary to ensure consistency.

2.3. Scalable Firebase Data Organization

Setting up Firebase connections to store user profiles and organize uploaded artworks is another essential component of MelodyCanvas. Since each user can contribute creative content, the platform must track uploads in a structured and scalable manner. Research in creative-computing systems notes that user-generated datasets must be indexed systematically to support downstream retrieval and visualization tasks [11]. To accomplish this, artworks can be categorized under unique user IDs in Firebase, while gallery scenes dynamically pull the associated metadata and image URLs. This ensures efficient retrieval when galleries refresh or when multiple users' works appear in shared spaces.

3. SOLUTION

MelodyCanvas runs on three major components: the user and artwork storage system, the AI content checker, as well as the 3D galleries.

Upon opening MelodyCanvas, players will be greeted with the login page, through which they can login to their already created profiles. However, if the player does not already have an account, they can click the signup button, which will take them to the signup page. Players can then create their profiles through this signup page, which are managed with Google's Firebase storage services.

After logging in, players will then be taken to the home page, from which players can either go and upload artworks through the upload page, or visit the galleries that they have created or are public (or they could log out).

After opening the upload page, players will have the option to upload an artwork from their computer files. This artwork will go through the AI content checker, meaning that it will be sent to Azure AI Vision. The artwork will then be checked for any inappropriate or copyright content, and if it clears all checks, the upload page will indicate that the upload has been approved. After then, players can move on to inputting the details of the artwork.

When players have uploaded their artworks, they can now return to the home page, where they can access their galleries. Players will have a couple of preset galleries (studio apartment, basic, museum, etc.) as their own, as well as some public galleries which they can visit. Upon opening, the gallery will load in artworks uploaded under the gallery owner's profile, and will refresh the gallery every 30 seconds.

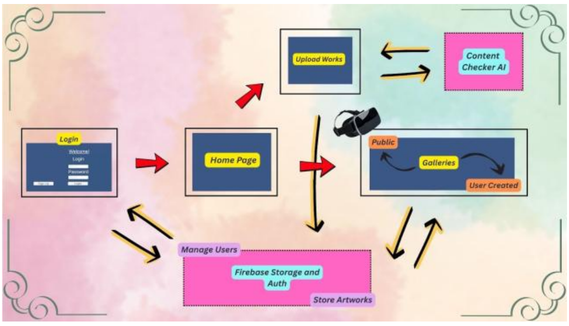

Figure 1. Overview of the solution

The first major component of MelodyCanvas is the user authentication and profile creation system, implemented using Firebase Authentication and Firebase Realtime Database. This component ensures secure login, account creation, and persistent user profiles. It relies on authentication protocols to verify identity and connects directly to Firebase to store structured artist data.

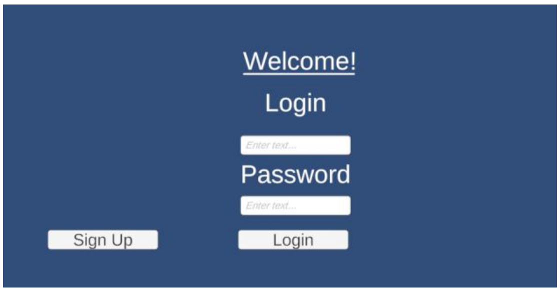

Figure 2. Screenshot of log in page

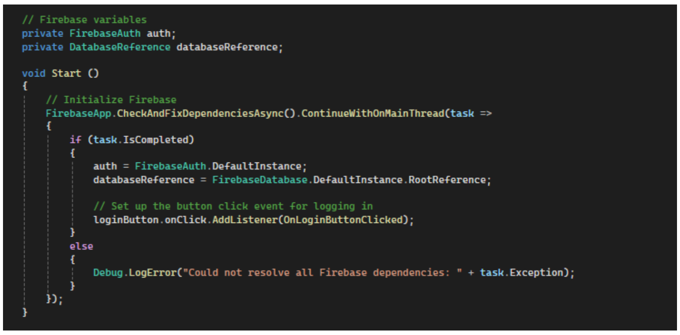

Figure 3. Screenshot of code 1

This code executes when a user attempts to log into MelodyCanvas from the login screen. The method begins by calling SignInWithEmailAndPasswordAsync, which sends the user's email and password to Firebase Authentication for verification. If the login is successful, Firebase returns a FirebaseUser object containing the user's unique ID. That ID becomes the key used to access the user's profile data in the Firebase Realtime Database.

Next, the program queries the database under "Artists/<userId>" to confirm whether profile data already exists. If the data is missing—indicating a first-time login—the system creates a new profile by constructing an ArtistInfo object, converting it to JSON, and writing it into the database. This ensures every authenticated user automatically receives a persistent profile.

Firebase handles the backend logic, including credential validation, error handling, and secure data storage. The Unity client simply sends requests and updates UI elements based on Firebase's responses.

The second major component of MelodyCanvas is the AI-powered content validation system, responsible for checking uploaded artworks for originality, copyright risks, and inappropriate material before allowing them into the platform. This system leverages Azure's computer-vision tools, but its effectiveness is limited by challenges found in neural-network similarity research, which shows that models may misclassify visually similar works or fail to detect transformed duplicates [12]. By serving as the gatekeeper for all uploads, this component maintains the integrity and safety of the creative space.

Figure 4. Screenshot of welcom page

```
        };
        request.SetRequestHeader("Content-Type","application/json");
        request.SetRequestHeader("Authorization","Bearer " + OpenAIKey);

        yield return request.SendWebRequest();

        if (request.result == UnityWebRequest.Result.Success)
        {
            var jsonResponse = JsonUtility.FromJson<ModerationResponse>(request.downloadHandler.text);
            HandleModerationResults(jsonResponse.results[0],imageType);
        }
        else
        {
            statusText.text = $"Error analyzing {imageType}: {request.error}";
        }
    }

    void HandleModerationResults (ModerationResult result,string imageType)
    {
        bool isFlagged = result.category_scores.sexual_content >= sexualContentThreshold ||
                         result.category_scores.violence >= violenceThreshold ||
                         result.category_scores.hate >= hateThreshold ||
                         result.category_scores.hate_threatening >= hateThreateningThreshold;

        statusText.text = isFlagged ? $"{imageType} rejected due to flagged content." : $"{imageType} approved!";
        continueButton.interactable = !isFlagged;
    }

    void ResetContentCheckState ()
```

Figure 5. Screenshot of code 2

Although the AI content checker runs prior to database upload, this snippet demonstrates the critical backend interaction that occurs once an artwork passes AI validation. After the cloud service verifies the originality and appropriateness of the image, the program prepares a structured dataset containing the artwork's title, style, image URL, artist name, and completion date. It then sends this dataset to Firebase using the SetValueAsync method.

Firebase processes this as a write request, storing the artwork under "Artists/<userId>/artworks/<artworkId>". This ensures all artworks are consistently indexed by user and retrievable by the gallery system. The continuation task checks whether the upload succeeded and logs the outcome for debugging.

In the full workflow, the AI checker operates as a gatekeeper: only after image analysis returns a positive result does this database operation run. Taken together, cloud vision analysis and database storage guarantee that only verified, appropriate artworks appear in VR galleries.

The final major component is the 3D gallery rendering and artwork display system, implemented inside Unity using your GalleryProfileLoader.cs logic. This component retrieves artwork data from Firebase, dynamically loads user images from Firebase Storage, and populates 3D environments with profile information, placards, and artwork textures for immersive gallery exploration.

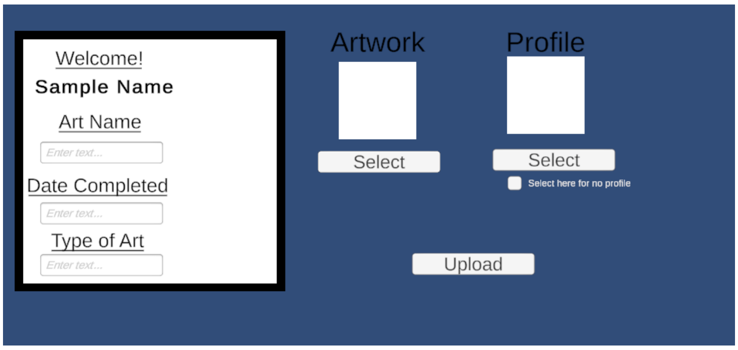

Figure 6. Screenshot of profile page

Figure 7. Screenshot of code 3

This portion of the gallery system runs whenever a player loads a gallery, whether it is their own or another user's. For each artwork stored under the selected user's Firebase profile, the program finds a corresponding painting object in the 3D environment. It locates the UI elements within that object—such as the artwork image, artist name text, and placard text—using a recursive tag-based search utility.

After gathering metadata such as title, style, and date completed, the program loads the associated artwork image from Firebase Storage. It does this by converting the gs:// storage URL into a public HTTP download link via GetDownloadUrlAsync. Once the link is retrieved, Unity begins a coroutine (LoadImage) to asynchronously fetch the texture over the network. When the texture successfully downloads, the program assigns it to the RawImage component on the painting object, enabling the artwork to appear inside the gallery.

This component brings together networking, asynchronous programming, 3D rendering, and Firebase data retrieval. It allows MelodyCanvas galleries to refresh dynamically, support mixed-user exhibitions, and present a visually immersive experience using real, student-generated artworks.

4. EXPERIMENT

4.1. Experiment 1

A potential blind spot in MelodyCanvas is the accuracy and consistency of the AI content checker when validating uploaded artwork. This system must correctly identify plagiarism and inappropriate content to ensure platform safety and integrity.

To test the reliability of the AI content checker, we would construct a dataset of images divided into three controlled categories: (1) original, student-generated artworks, (2) copyrighted artworks deliberately scraped from online sources, and (3) ambiguous artworks that share stylistic similarities but are not direct copies. The experiment would send each image to the Azure Vision API under identical conditions. We would record whether the checker approved or rejected each submission. The control data for originality labels would come from a manually validated dataset reviewed by two human evaluators. This experimental structure allows us to isolate misclassification patterns and identify where the AI struggles most.

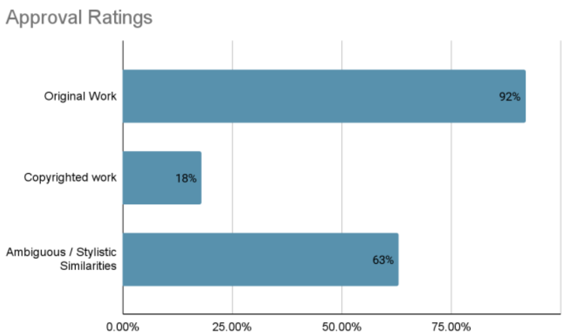

Figure 8. Figure of experiment 1

The approval rates show clear separation between original and copyrighted images, suggesting the AI checker performs well in detecting direct plagiarism. The mean approval rate across all categories is 57.6%, while the median is 63%, reflecting the mid-range performance of the ambiguous category. The lowest value is 18% (copyrighted images), and the highest is 92% (original artworks).

The ambiguous images produced the most surprising results. A 63% approval rate indicates that the AI sometimes misidentifies stylistically similar artworks as original, likely because the model relies primarily on surface-level features rather than deep stylistic analysis. This suggests that originality checking is more complex than inappropriate-content detection and might require additional methods, such as embedding-based similarity models.

Overall, the biggest influence on results is the degree of visual similarity between submissions and known copyrighted works. Enhancing model training with more nuanced examples of stylistic overlap would likely improve accuracy.

4.2. Experiment 2

A second blind spot in MelodyCanvas is the performance and loading speed of the 3D gallery system, particularly when retrieving artwork images from Firebase Storage and populating them into the VR environment.

Research on VR immersion demonstrates that user discomfort increases sharply when loading delays break presence or visual continuity, making performance stability essential for maintaining engagement [15]. This experiment measures gallery loading time under varying network and dataset conditions. Three scenarios are tested: (1) a small gallery with five artworks, (2) a medium gallery with fifteen artworks, and (3) a mixed-user gallery loaded using the LoadMixedRandomGallery method from GalleryProfileLoader.cs. For each scenario, the system logs the total time from gallery scene load to full texture display on all paintings. Each test condition is repeated ten times to reduce noise from network variance. Control data includes the file sizes of artworks (standardized to 512×512 resolution) to ensure that differences in load speed are not due to inconsistent image weights.

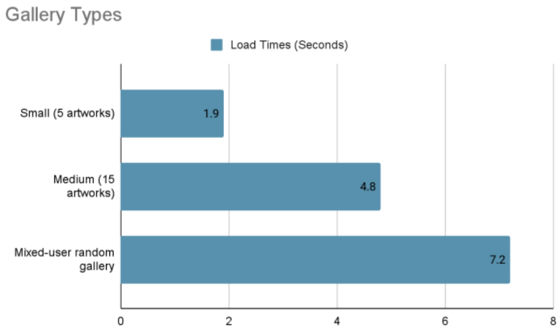

Figure 9. Figure of experiment 2

The data shows a predictable pattern of increasing load times as the number of artworks grows. The mean loading time across all conditions is 4.63 seconds, while the median is 4.8 seconds. The lowest value is 1.9 seconds (small gallery), and the highest is 7.2 seconds (mixed-user gallery). The mixed gallery loads slowest because it triggers multiple Firebase queries—one for each contributing artist—and requires additional sorting and shuffling operations as described in your LoadMixedRandomGallery method.

One unexpected observation is that the medium gallery does not scale linearly with the small gallery. The disproportionately higher load time suggests network overhead, Firebase Storage URL resolution delays, or inefficiencies in coroutine overlap when calling LoadImage repeatedly. The dominant factor influencing results appears to be the number of network fetches and Firebase lookups rather than Unity's rendering time. Optimizations such as caching download URLs or batching Firebase requests could significantly improve performance.

5. Related Work

A relevant methodology addressing the problem of verifying originality in visual media is presented in Deep Visual Similarity for Content Moderation: Detecting Plagiarized Images at Scale [8]. The study proposes a plagiarism-detection pipeline built upon deep-learning similarity embeddings, where a model such as ResNet-50 extracts high-dimensional feature vectors and compares them through approximate nearest-neighbor search to detect near-duplicate images, even when modified through cropping, scaling, or color alteration. This approach is highly effective for identifying visual plagiarism at scale; however, it requires significant computational infrastructure and a large indexed dataset of known works. MelodyCanvas improves on this method by integrating plagiarism detection into a structured educational context, where originality checks can be combined with human oversight and metadata validation rather than relying solely on fully automated similarity scoring.

Another methodology is described in Plagiarism Detection of Images [9], which highlights the limitations of conventional plagiarism-detection techniques when applied to visual media. The authors argue that most existing tools are designed for text and that image plagiarism is far more difficult to detect, especially when transformations or stylistic reinterpretations are applied. They survey multiple algorithms—including feature extraction, hashing, and template matching—and conclude that no single technique consistently balances accuracy, computational efficiency, and robustness to visual modifications. This methodology reveals a gap in current research: visual plagiarism detection remains unreliable without hybrid strategies. MelodyCanvas addresses this shortcoming by combining automated image checks with upload gating, manual metadata entry, and user accountability mechanisms, producing a more holistic and context-aware validation process.

A third methodology relevant to the design of MelodyCanvas is found in Huang’s Effects of Interactive Loading Interfaces for Virtual Reality [10]. The study examines how interactive loading screens influence user experience in VR environments and concludes that such interfaces significantly reduce perceived waiting time, improve emotional response, and increase patience during long or variable load durations. While the methodology does not directly address image processing or originality validation, it provides critical insight into managing performance-related friction, especially when retrieving large image assets from cloud storage. Compared to this approach, MelodyCanvas currently prioritizes backend optimization over UX-driven loading strategies. Integrating interactive VR loading elements, as suggested by Huang, would allow the platform to maintain immersion even when gallery loading requires multiple network operations.

6. CONCLUSIONS

Although MelodyCanvas successfully integrates authentication, content validation, and dynamic gallery rendering, several limitations remain. The most significant constraint lies in the reliability of automated originality detection. Current vision-based plagiarism checks may fail to identify stylistically similar images or transformed copyrighted works, leaving blind spots that require a more sophisticated similarity-embedding model or a hybrid automated–human review process. In addition, the gallery loading system depends heavily on network conditions and asynchronous Firebase calls, which can lead to inconsistent loading speeds and reduced immersion for VR users. Performance optimizations such as caching download URLs, batching Firebase requests, or preloading image metadata would help address these issues. These weaknesses align with findings from scalable similarity-embedding research, which shows that shallow feature comparisons often fail to detect transformed or partially duplicated images [8]. Finally, the user experience could be improved by adopting interactive loading interfaces and clearer visual feedback during image uploads and gallery transitions. Given more development time, these enhancements would strengthen both the technical reliability and the usability of the platform.

MelodyCanvas demonstrates how VR, cloud storage, and AI-based analysis can create an accessible, immersive environment for young artists. By improving originality detection, optimizing gallery performance, and refining user experience, the platform can evolve into a scalable educational tool that meaningfully supports creativity and cultural connection among emerging artists.

REFERENCES

[1] Rabkin, Nick, and Eric Christopher Hedberg. "Arts Education in America: What the Declines Mean for Arts Participation. Based on the 2008 Survey of Public Participation in the Arts. Research Report# 52." National Endowment for the Arts (2011).
[2] Bolz, Anna. "A regulatory framework for the art market." Authenticity Forgeries and the Role of Art Experts. Switzerland: Springer (2022): 107-253.
[3] Bowen, Daniel H., Jay P. Greene, and Brian Kisida. "Learning to think critically: A visual art experiment." Educational Researcher 43.1 (2014): 37-44.
[4] Catterall, James S. "The Arts and Achievement in At-Risk Youth: Findings from Four Longitudinal Studies. Research Report# 55." National Endowment for the Arts (2012).

[5] Beyer, Margaret Ellen. Differences in Art and Music High School Students' SAT Test Scores from 2006 to 2015. Diss. Walden University, 2024.

[6] Parsad, Basmat, and Maura Spiegelman. "A Snapshot of Arts Education in Public Elementary and Secondary Schools: 2009-10. First Look. NCES 2011-078." National Center for Education Statistics (2011).

[7] McMurrer, J. "Choices, changes, and challenges: Curriculum and instruction in the No Child Left Behind era." (2006).

[8] Wang, Wenxuan, et al. "An image is worth a thousand toxic words: A metamorphic testing framework for content moderation software." 2023 38th IEEE/ACM International Conference on Automated Software Engineering (ASE). IEEE, 2023.

[9] Ibrahin, Amirul S. Bin, Othman O. Khalifa, and Diaa Eldein M. Ahmed. "Plagiarism detection of images." 2020 IEEE Student Conference on Research and Development (SCOReD). IEEE, 2020.

[10] Huang, Yi-Ting, Chih-Chieh Hsu, and Tzu-Hsuan Wang. "Effects of interactive loading interfaces for virtual reality game environments on time perception, cognitive load, and emotions." Frontiers in Virtual Reality 6 (2025): 1540406.

[11] Fiebrink, Rebecca. "Machine learning education for artists, musicians, and other creative practitioners." ACM Transactions on Computing Education (TOCE) 19.4 (2019): 1-32.

[12] Bouzerdoum, Abdesselam, A. Havstad, and A. Beghdadi. "Image quality assessment using a neural network approach." Proceedings of the Fourth IEEE International Symposium on Signal Processing and Information Technology, 2004.. IEEE, 2004.

[13] Al-Kaff, Abdulla, et al. "Survey of computer vision algorithms and applications for unmanned aerial vehicles." Expert Systems with Applications 92 (2018): 447-463.

[14] Bowman, Doug A., et al. "New directions in 3d user interfaces." International Journal of Virtual Reality 5.2 (2006): 3-14.

[15] Slater, Mel, and Maria V. Sanchez-Vives. "Enhancing our lives with immersive virtual reality." Frontiers in Robotics and AI 3 (2016): 74.