Enhancing Museum Experiences with Augmented Reality and Machine Learning: A Case Study of Egyptian Cultural Heritage

Submitted by

Ghada Abdelhady [1], Ahmed Atef[2], Aya Mostafa [3]

1. General Systems Engineering Faculty of Engineering October University for Modern Sciences and Arts.
2. IBM, Smart Village, Giza, Egypt.
3. HemayaIT, Maadi, Cairo, Egypt.

DOI:

https://doi.org/10.21608/ijtec.2025.350216.1005

IJTEC

International Journal of Technology and Educational Computing

Volume (4), Issue (10), January 2025

E-ISSN: 2974-413X P-ISSN: 2974-4148

https://ijtec.journals.ekb.eg/

Publisher

Association of Scientific Research Technology and the Arts

https://srtaeg.org/

Enhancing Museum Experiences with Augmented Reality and Machine Learning: A Case Study of Egyptian Cultural Heritage

Submitted by

Ghada Abdelhady [1], Ahmed Atef[2], Aya Mostafa [3]

1. General Systems Engineering Faculty of Engineering October University for Modern Sciences and Arts.
2. IBM, Smart Village, Giza, Egypt.
3. HemayaIT, Maadi, Cairo, Egypt.

ABSTRACT

This study explores the application of augmented reality (AR) to enhance visitor engagement and accessibility in Egyptian museums, comparing two approaches: the Vuforia SDK and a deep learning model integrated with Unity.

Addressing the need for innovative solutions to improve cultural heritage experiences, the research investigates how AR can deliver immersive and educational interactions with Egyptian artifacts. The Vuforia-based prototype offers structured content delivery using predefined image targets, while the deep learning model employs convolutional neural networks (CNNs) for real-time and adaptive object recognition across varied environments. User testing revealed that, although the Vuforia prototype is lightweight and user-friendly, the deep learning model outperformed it in accuracy and adaptability. However, its computational demands pose challenges for scalability. This paper highlights the transformative potential of deep learning-driven AR in cultural heritage and proposes strategies to optimize its adoption in the tourism sector.

KEYWORDS: **Augmented Reality, Cultural Heritage, Egyptian Artifacts, Machine Learning, Museum Experiences**

1. Introduction

The rich and timeless history of ancient Egyptian civilization is often inaccessible to modern audiences due to the lack of detailed guidance and contextual information in museums. This challenge is particularly pronounced for independent tourists, who, in the absence of tour guides, may struggle to fully appreciate the significance of artifacts such as hieroglyphics or statues. Addressing this gap necessitates innovative solutions to enhance the accessibility and engagement of cultural heritage experiences for diverse audiences.

Globally, augmented reality (AR) has emerged as a transformative tool across various domains, offering interactive and immersive experiences. In the museum sector, AR applications like ArtLens 2.0 at the Cleveland Museum of Art enhance visitor navigation and understanding of artwork through engaging technical features (Alexander, 2014; Ding, 2017). Similarly, "The Speaking Celt," an AR-based app developed for Salzburg's Museum of Celtic Heritage, integrates storytelling and historical narratives using digital avatars, underscoring the potential of AR to merge education with interactive experiences (*A Celtic Museum Experience Using Augmented Reality | ViMM*, n.d.; Breuss-Schneeweis, 2016).

Beyond museums, AR has also demonstrated its versatility in commercial applications. For example, IKEA Place helps users visualize furniture in their homes, streamlining purchasing decisions and reducing returns (Alves & Luís Reis, 2020; Ozturkcan, 2021). Moreover, the global success of Pokémon GO illustrates AR's ability to combine entertainment with real-world interaction, captivating millions through location-based gameplay (Hamari et al., 2019; Rauschnabel et al., 2017). These examples highlight AR's wide-ranging utility in enhancing user engagement. However, the application of AR in museums is often limited to static, predefined image targets, restricting adaptability to dynamic environments and individual visitor preferences. Such limitations call for a more flexible approach that integrates real-time object recognition and enhanced interactivity.

This study aims to develop an AR-based platform that transforms museum visits into interactive and educational experiences. Acting as a virtual tour guide, the platform is designed to serve diverse audiences by providing immersive engagement and detailed artifact information. Leveraging machine learning (ML) techniques, the proposed system employs convolutional neural networks (CNNs) for real-time 3D object recognition, enabling automated artifact identification and improved adaptability across museum settings. By reducing reliance on manual inputs and enhancing scalability, the system bridges the gap between cultural heritage and modern technology.

Through this research, AR technology is introduced as a local innovation, synergizing with ML to enrich user experiences and foster deeper appreciation of Egypt's historical treasures. The platform not only seeks to attract more visitors but also aims to demonstrate AR's potential as a transformative tool in cultural heritage and tourism sectors.

2. Literature Review

Recent advancements in augmented reality (AR) have significantly transformed museum experiences, fostering enhanced user engagement and interaction. AR technology has been increasingly adopted to provide immersive environments, offering innovative ways to interpret and explore cultural artifacts. Notable examples include:

ArtLens 2.0 (Cleveland Museum of Art): This application combines AR and interactive features to guide users through gallery exploration, enhancing their understanding of artwork by integrating technical and interactive components (Alexander, 2014; Ding, 2017)

The Speaking Celt (Salzburg's Museum of Celtic Heritage): This app uses storytelling through digital avatars to present historical narratives, demonstrating AR's potential for blending educational content with entertainment (*A Celtic Museum Experience Using Augmented Reality | ViMM*, n.d.; Breuss-Schneeweis, 2016).

IKEA Place: Beyond museums, AR's application in retail, such as IKEA Place, allows users to visualize furniture within their own living spaces. This practical use case highlights AR's versatility in providing context-aware solutions (Alves & Luís Reis, 2020; Ozturkcan, 2021)

Pokémon GO: As a global phenomenon, Pokémon GO illustrated AR's potential to merge entertainment with real-world exploration, engaging users in interactive, location-based experiences (Hamari et al., 2019; Rauschnabel et al., 2017).

Despite these successes, AR applications in museums often rely on static, predefined image targets, which limit adaptability to dynamic environments and diverse visitor preferences. Studies focusing on cultural heritage frequently emphasize static AR experiences, such as overlaying textual information on images, without exploring more advanced capabilities like real-time object recognition or adaptive content delivery (Galal, 2021; Mohammad et al., 2020).

The integration of machine learning (ML) with AR, however, has shown promise in overcoming these limitations. Convolutional neural networks (CNNs), in particular, enable real-time object recognition, offering the ability to identify artifacts from multiple angles under varied environmental conditions. This dynamic functionality enhances user engagement by providing more responsive and personalized experiences. For instance, applications that utilize ML for artifact recognition can automate the identification process, reducing manual input while ensuring scalability across different museum settings (Stenroos, 2017; *Why Are Convolutional Neural Networks Good for Image Classification? | by Prafful Mishra | DataDrivenInvestor*, n.d.).

By addressing these gaps, this research advances the field of AR in cultural heritage by integrating CNNs for multi-angle artifact recognition and adaptability. This approach represents a significant step toward scalable, real-time AR applications that enhance museum visitors' experiences while bridging the gap between static displays and interactive digital environments.

3. Software and Hardware Requirements

For the implementation of the system, some requirements are needed for the end product. Hardware specifications for developers creating the platform include an AMD Radeon RX 5600 XT or NVidia GeForce GTX 1660 Super, or an equivalent. CPU having a 3.5GHz minimum clock speed. SSD with 512GB of capacity. 16GB RAM. And finally, an Android phone (S6+) with the latest OS.

For the users, a smart device is only needed as a hardware requirement as the system is produced in the form of an application. For the software requirements, the smart device must have no less than iOS 14 for iPhone, and no less than android 7 for android devices.

4. Methodology

This section outlines the tools, technologies, and workflows utilized in the development of the proposed augmented reality (AR) system. It emphasizes the integration of AR software development kits (SDKs), machine learning (ML) applications, and system design to ensure a robust and user-friendly experience. Each subsection delves into the components and their contributions to the project's goals.

4.1 Dataset Preparation

The dataset used for this project was sourced from the "National Museum of Egyptian Civilization" collection on Kaggle. It comprised over 3,000 annotated images of artifacts, including statues such as the Female Peasant, Statue of the Sphinx, and Hassan Fathi. Figure 1 shows examples of the dataset in Arabic and English, highlighting the dual-language support.

exhibits	Text in Arabic	Text in English
the_female_peasent	**تمثال الفلاحة**: يعتبر هذا التمثال من الأعمال البارزة للفنان محمود مختار والذي يبرز فيه مكانة الفلاحة المصرية ومحورها كأحد أهم الأدوار الرئيسية للمرأة في المجتمع المصرى قديما	**The Femal Peasent:** This statue is considered one of the prominent works of the artist Mahmoud Mokhtar, in which he focused the position
Statue_ofthe_sphinx	**تمثال لأبي الهول**: تمثال مصغر لأبي الهول لا يوجد عليه نقوش تدل علي تحديد تاريخة الأثري بشكل واضح, ويعد هذا التمثال استمرار لفكرة تماثيل أبو الهول الفرعونية , حيث كان شائعا	**Statue of the Sphinx:** Statue of Sphinx, which has no inscriptions indicating its archaeological history clearly. This statue is
Hassan_Fathi	**حسن فتحى (١٩٠٠ م- ١٩٨٩ م):** ولد في الأسكندرية ودرس في جامعة " مؤاد الأول " , ولقب بشيخ المعماري المصريين ، ويعد من أهم مهندسي العصر	**Hassan Fathi (1900-1989):** Hassan Fathi was born in Alexandri and studied at the University of "Fou'ad I" Known as "sheikh

Figure 1: Examples of the dataset in Arabic and English (*National Museum OF Egyptian Civilization*, n.d.)

Data Augmentation: To enhance model robustness under varied conditions, techniques such as rotation, scaling, flipping, and contrast adjustment were applied. This expanded the dataset, ensuring the trained model could adapt to environmental variations like lighting and angles.

The prepared dataset formed the foundation for the machine learning model, enabling efficient artifact recognition.

4.2 Data Splitting and Proportions for Training and Testing

The dataset was divided into three subsets: 70% for training, 15% for validation, and 15% for testing. This split ensured that the training subset provided sufficient data for learning while reserving adequate data for unbiased evaluation. Stratified sampling was employed to maintain consistent representation of artifact categories across all subsets, mitigating risks of data imbalance and performance bias.

4.3 Development Tools and Software

The AR system was developed using the following tools and technologies:

Unity: Served as the primary platform for creating interactive 2D and 3D content (*Unity Real-Time Development Platform | 3D, 2D, VR & AR Engine*, n.d.).

Vuforia SDK: Used for the initial prototype, offering features like object detection and image recognition (*Vuforia Enterprise Augmented Reality (AR) Software | PTC*, n.d.).

TensorFlow: Integrated with Unity's Barracuda package to implement machine learning for real-time artifact recognition (*Introduction to Barracuda | Barracuda | 2.0.0*, n.d.; *TensorFlow*, n.d.).

Pre-trained CNN Model: The VGG-16 model was utilized as a starting point for transfer learning, known for its effectiveness in feature extraction and object recognition tasks as illustrated in Figure 2. This model enabled the system to identify artifacts from multiple angles with high accuracy (Simonyan & Zisserman, 2014).

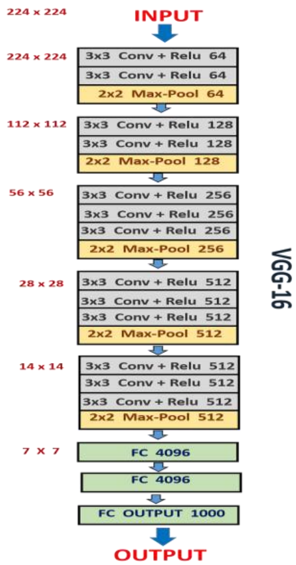


Figure 2: VGG-16 model (Simonyan & Zisserman, 2014)

4.4 Workflow

The system development followed an iterative approach, encompassing two main prototypes:

Prototype 1 (Vuforia SDK):

- Predefined image targets were uploaded to Unity to enable object recognition and content overlay.
- Challenges: Dataset size limits and sensitivity to lighting and environmental factors restrict the prototype's scalability.

Prototype 2 (Deep Learning with TensorFlow):

- A CNN model trained on the prepared dataset was integrated into Unity using the ONNX format for compatibility.
- This prototype achieved higher recognition accuracy, adaptability under varied conditions, and scalability for large datasets.

Figure 3 illustrates the system's overall workflow, which includes:

- Data collection and preparation.
- Model training and architecture selection.
- Analysis and evaluation.
- Final model consumption for real-time artifact recognition.

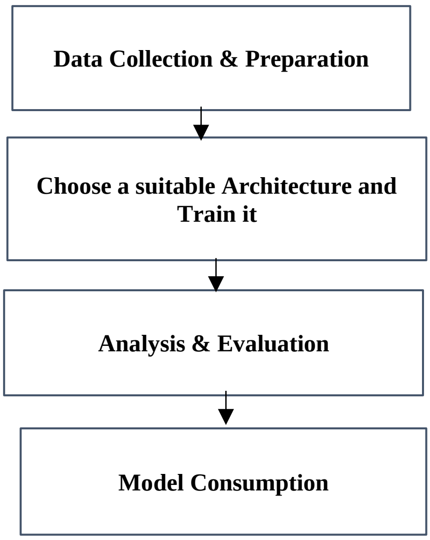


Figure 3 Second Prototype workflow

This iterative approach ensured that each prototype addressed specific challenges, paving the way for an optimized final system that balanced AR tool capabilities and machine learning advancements.

The two approaches were evaluated based on Detection accuracy, Scalability, Application size, and User satisfaction.

The findings guided the design and implementation decisions for the final AR system, prioritizing performance and user experience. These considerations directly influenced the choice of SDKs and the integration of machine learning techniques.

4.5 Technologies Used

The integration of cognitive technologies with machine learning was central to achieving immersive AR experiences. Using the pre-trained VGG-16 model, the system employed deep neural networks for:

- 3D object recognition.
- Image clustering and analysis.
- Artifact identification under varied conditions.
- These capabilities replaced traditional static methods, ensuring scalability and precision.

4.6 Project Design

The project design includes a detailed block diagram illustrating the system's workflow and components. As shown in Figure 4, the system begins with user input through a smart device, followed by rendering virtual content. Machine learning (ML) enhances the AR experience by enabling real-time object detection, which is visually represented in the extended block diagram in Figure 5. This design ensures seamless interaction between users and the AR environment, providing an intuitive interface and robust functionality.

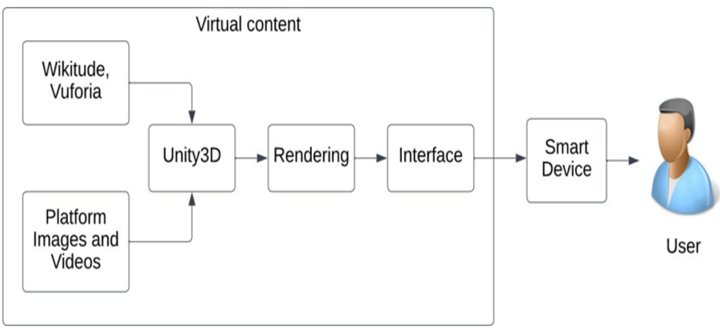


Figure 4: System Block Diagram

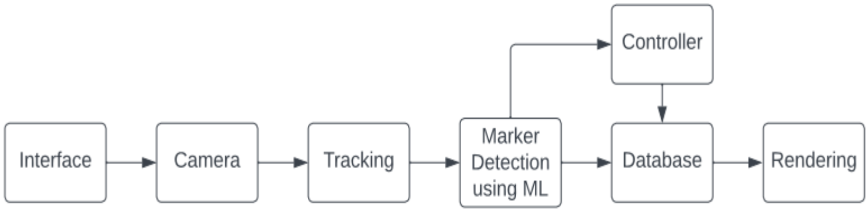


Figure 5: Extended Block Diagram

4.7 Workflow

The system flow starts with a user opening the application and accessing the AR camera. Detected objects are processed using ML algorithms, and relevant virtual content is rendered. As shown in Figure 6, the workflow includes continuous tracking until the user exits the application, ensuring smooth and engaging user experience. This iterative design guarantee's reliability and adaptability during user interactions, directly addressing the challenges identified in earlier prototypes.

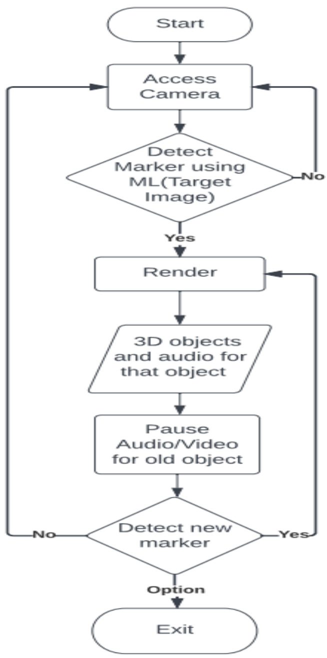


Figure 6: System Flowchart

5. Software Implementation

This section provides a comprehensive overview of the steps and methodologies employed in the software implementation phase of the AR project, focusing on Unity3D and machine learning (ML) integrations. The development process involved iterative trials to explore and refine configurations, ensuring optimal performance for real-time artifact recognition.

5.1 Unity Demos

To identify best practices for utilizing Unity3D in AR development, a series of iterative trials were conducted. These trials assessed Unity's capabilities and limitations when integrating AR functionalities

First Trial: A 3D whale model was inserted into Unity without an image target as shown in Figure 7. The model failed to integrate with the real-world scene due to the absence of an AR Camera and image targets. These challenges were addressed by replacing the main camera with an AR Camera in subsequent trials.

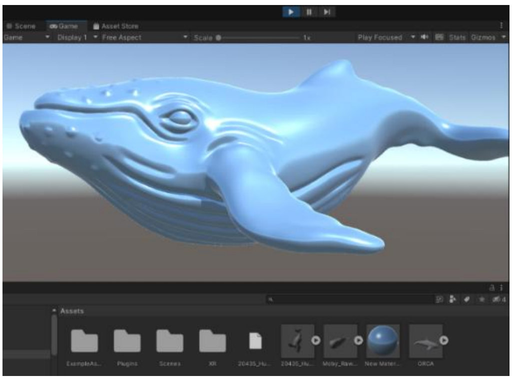


Figure 7: 3D model of whale in Unity

Second Trial: Utilizing Unity3D with AR Camera integration, a 3D asset was successfully placed into the real-world environment, show in Figure 8. Despite this success, the object appeared without detecting image targets. This limitation prompted the integration of image targets and code for detection in the next trial.

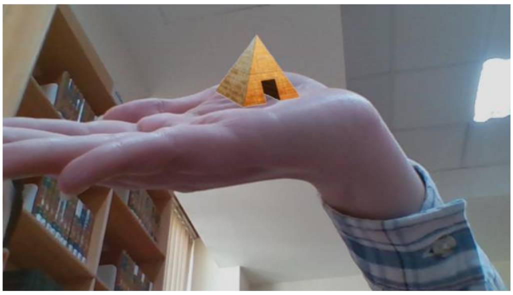


Figure 8: 3D model applied to the real world

Third Trial: By creating an AR Tracked Image in Unity, an image of a donut was uploaded as a target image. Using a custom C# script, the application successfully rendered a 3D model on the detected target, as shown in Figure 9. Challenges included errors in the detection code, which were resolved with further modifications.

Figure 9: Third demo showcasing image target detection

Fourth Trial: The integration of Vuforia SDK into Unity was explored. This trial involved generating a license key through the Vuforia platform and linking it to Unity. The project was successfully implemented, displaying 3D models based on detected image targets, as shown in Figures 10 and 11. However, issues arose when transitioning the project to Android platforms.

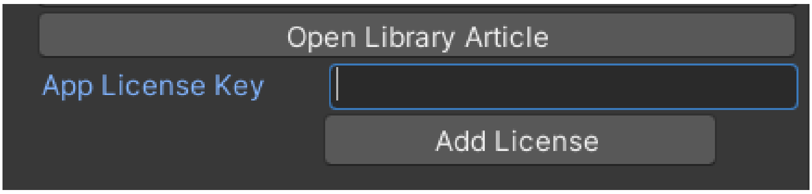


Figure 10: Vuforia License added to Unity

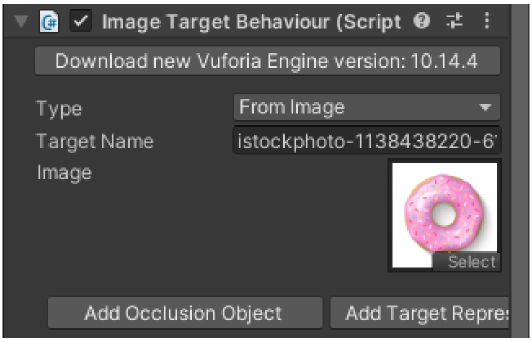


Figure 11: Input target image

Fifth Trial: Further refinements using Vuforia focused on enhancing the accuracy of the system as shown in Figure 12. Limitations, such as Vuforia's inability to upload large datasets or support varied lighting conditions, necessitated the transition to machine learning techniques for improved scalability and precision.

Figure 12: Final trial with Vuforia SDK

5.2 ML Integration

To overcome the challenges of manually uploading extensive datasets in Vuforia, a machine learning approach was adopted for object detection. Convolutional Neural Networks (CNNs) were employed due to their effectiveness in reducing dimensionality and optimizing image classification processes (Stenroos, 2017; *Why Are Convolutional Neural Networks Good for Image Classification? | by Prafful Mishra | DataDrivenInvestor*, n.d.).

5.2.1 Machine Learning Model Development

- **Dataset Preparation**: The "National Museum of Egyptian Civilization" dataset from Kaggle was utilized to train the model(*National Museum OF Egyptian Civilization*, n.d.). It included 3,000+ annotated images. Data augmentation techniques (rotation, scaling, flipping) ensured robustness under varied conditions.
- **Training Framework**: The model was developed using Python, TensorFlow, OpenCV, and NumPy. The VGG-16 pre-trained model was employed for feature extraction and transfer learning(Simonyan & Zisserman, 2014).
- **ONNX Format**: The trained model was converted into the ONNX format for seamless integration with Unity through the Barracuda package(*ONNX | Home*, n.d.).

5.2.2 Performance Evaluation

Key metrics used for evaluating model performance include:

Precision: The proportion of true positive identifications among all predicted positives.

$$Precision = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Positives\ (FP)}$$

Recall: The proportion of true positive identifications among all actual positives.

$$Recall = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Negatives\ (FN)}$$

Loss: A measure of model error, computed as the difference between predicted and actual values.

$$Loss = \frac{1}{N}\sum_{i=1}^{N}[y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i)]$$

Accuracy: The proportion of correctly identified artifacts out of all predictions.

$$Accuracy = \frac{True\ Positives\ (TP) + True\ Negatives\ (TN)}{Total\ Observations}$$

Binary Accuracy: A metric that indicates the proportion of correct predictions out of total predictions in binary classification tasks.

$$Binary\ Accuracy = \frac{Number\ of\ Corrected\ Predictions}{Total\ Predictions}$$

The ML model's performance was evaluated using training/validation metrics and classification results:

Accuracy and Loss Trends: The performance evaluation of the proposed system is demonstrated through the training and validation accuracy and loss metrics. As shown in Figure 13, the accuracy trends over epochs highlight the convergence of the model's training accuracy (98%) with validation accuracy (97%), indicating consistent learning. Similarly, Figure 14 presents the loss curves, which show a steady decline in both training loss (from 0.6 to 0.02) and validation loss (from 0.7 to 0.03), confirming the model's improved performance over time without overfitting.

```
fig = plt.figure()
plt.plot(hist.history['accuracy'], color='teal', label='accuracy')
plt.plot(hist.history['val_accuracy'], color='orange', label='val_accuracy')
fig.suptitle('Accuracy', fontsize=20)
plt.legend(loc="upper left")
plt.show()
```

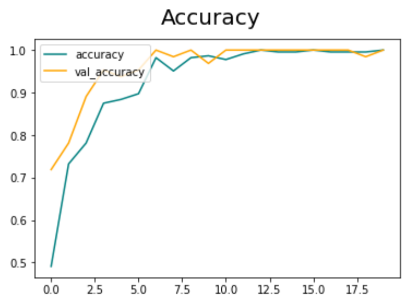


Figure 13: Accuracy Curve

```
fig = plt.figure()
plt.plot(hist.history['loss'], color='teal', label='loss')
plt.plot(hist.history['val_loss'], color='orange', label='val_loss')
fig.suptitle('Loss', fontsize=20)
plt.legend(loc="upper left")
plt.show()
```

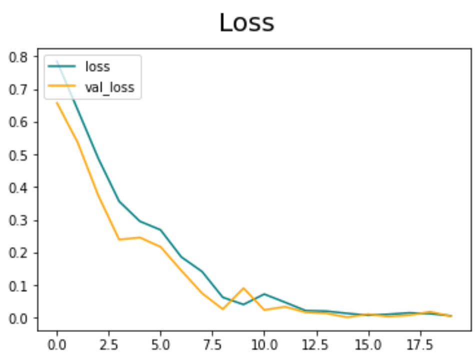


Figure 14: Loss Curve

Precision, Recall, and Binary Accuracy:

Precision, recall, and binary accuracy metrics were computed to assess the model's effectiveness in identifying artifacts. These metrics are as follows:

- **Precision**: The model achieved a precision score of 96%, indicating that 96% of the predicted artifacts were correctly identified as relevant.
- **Recall**: The recall score of 94% demonstrates the model's ability to identify 94% of all relevant artifacts in the dataset.
- **Binary Accuracy**: The overall binary accuracy was measured at 97%, consolidating the precision and recall metrics into a single robust performance indicator.

These metrics demonstrate the reliability and effectiveness of the proposed system in artifact recognition tasks.

Artifact Recognition Example

An example of artifact recognition is illustrated in the visualized workflow. As depicted in Figure 15, the input image of the "Statue of the Sphinx" was processed through the trained model, which accurately classified the artifact. This recognition demonstrates the system's seamless integration of machine learning with augmented reality, allowing real-time identification and interaction.

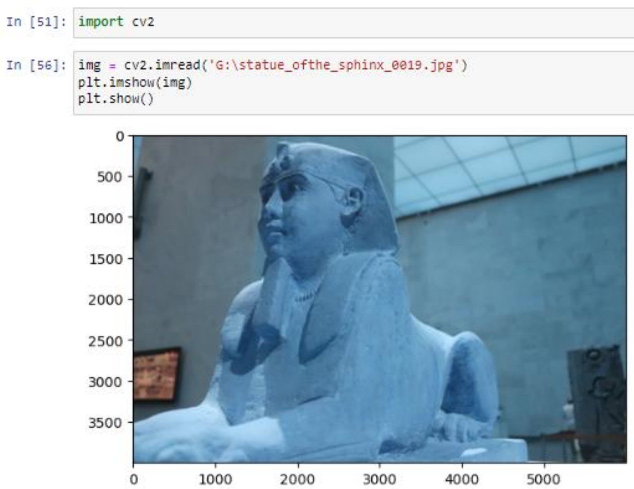


Figure 15: Artifact Recognition Example

The integration of CNNs into Unity significantly enhanced the automation and accuracy of detecting target images. This approach not only addressed the limitations of existing SDKs but also enabled greater flexibility for deploying the system across various environments.

5.3 Key Outcomes

1. **Unity and Vuforia Integration**:
   - **Initial** trials demonstrated the feasibility of combining Unity3D with Vuforia SDK for AR application development.
   - Limitations in dataset size, scalability, and lighting adaptability necessitated the transition to machine learning.

2. **ML Integration**:
   - CNNs significantly enhanced detection accuracy and scalability.
   - The system achieved multi-angle recognition and adaptability across diverse environments, overcoming the static nature of traditional AR SDKs(*ARKit 6 - Augmented Reality - Apple Developer*, n.d.; *Build New Augmented Reality Experiences That Seamlessly Blend the Digital and Physical Worlds | ARCore | Google for Developers*, n.d.; *Perkenalan Situs Slot Gacor - Perkenalan Situs Slot Gacor Yang Sedang Lagi Malak - Malaknya*, n.d.; *Vuforia Enterprise Augmented Reality (AR) Software | PTC*, n.d.; *Wikitude Augmented Reality: The World's Leading Cross-Platform AR SDK*, n.d.).

3. **Flexibility and Performance**:

The ML-based approach streamlined artifact recognition, enabling support for large datasets and reducing application size

By combining Unity's versatility with the advanced capabilities of CNNs, the project successfully addressed the challenges of dynamic artifact recognition and user engagement. This implementation lays the groundwork for scalable and adaptable AR systems in cultural heritage applications.

6. Final Results and Discussions

This section discusses the features, challenges, and performance of the two prototypes, emphasizing real-world applicability through specific case scenarios.

6.1 Prototype 1: Using Vuforia

Vuforia was selected for its efficiency and performance. The user interface (UI) consists of three buttons:

- Start: Activates the camera and requests hardware permissions.
- User Manual: Provides usage guidance.
- Exit: Closes the application.

The canvas hierarchy, as shown in Figure 16, was created in Unity, with each button linked to a specific C# script for functionality. The script manages interactions, such as activating or deactivating game objects based on user input. Components were assigned to corresponding objects in the Unity hierarchy as shown in Figure 17.

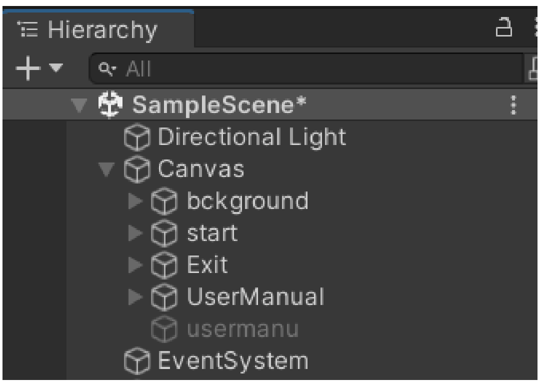


Figure 16: Canvas Hierarchy

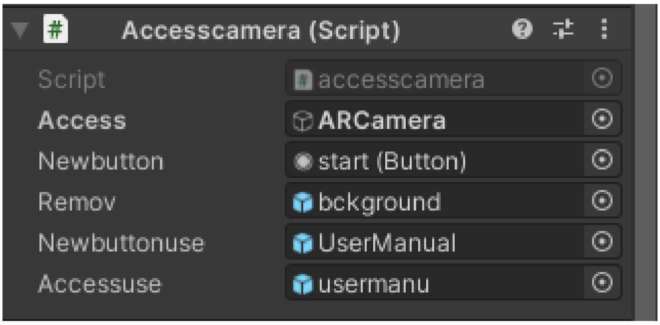


Figure 17: Assign components to objects

Additionally, a C# script was implemented to allow users to quit the application. The final user interface, depicted in Figure 18, includes text-to-speech functionality in English and Arabic to enhance accessibility. By incorporating audio sources into the hierarchy, presented in Figure 19, users can listen to object descriptions upon detection.

Figure 18: User Interface

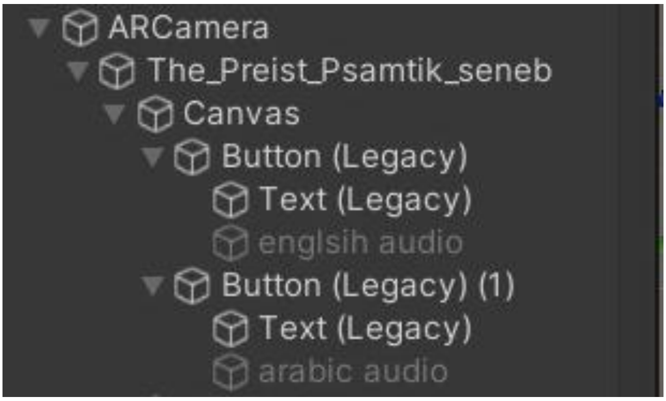


Figure 19: Unity's Hierarchy for Incorporating Audio

6.2 Prototype 2: Using Machine Learning

A machine learning model was trained using TensorFlow and integrated into Unity through the Barracuda package. This package supports ONNX-format neural networks, enabling Unity to process ML model outputs. The steps to integrate ML with Unity are as follows:

1. Convert the TensorFlow model to ONNX format.
2. Create a label text file for class identification.
3. Write a C# script to load the model and classify camera input in real time.
4. Assign required files, such as the model and label file, to Unity objects as presented in Figure 20.

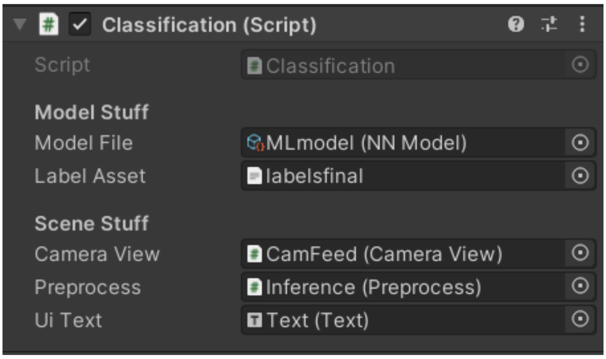


Figure 20: Assign Objects

6.3 Technical Challenges Encountered

Some challenges have been encountered during the project implementation. Limited resources for Unity-ML integration led to extensive experimentation with SDKs like Barracuda, NatML, and TensorFlow Lite. Model format conversion (from *.py to ONNX) was necessary for Unity compatibility. Capturing and processing each scene required custom C# scripts to feed data to the ML model. Developing a functional graphical user interface (GUI) to automate outputs based on detected objects remains a work in progress(*Function*, n.d.; *Unity - Scripting API:*, n.d.; *Unity UI (UGUI) | Unity UI | 2.0.0*, n.d.).

6.4 Comparative Performance Analysis

This section provides an in-depth evaluation of the augmented reality (AR) application's functionality and user experience. The testing focuses on the effectiveness of the prototypes in enhancing user engagement and interaction with cultural artifacts, bridging the gap between the virtual and physical worlds.

The testing process utilized the following components: Augmented Reality Application, Testing Environment, Testing Protocol, and Testing Software Components and Units.

6.4.1 Augmented Reality Application

Two prototypes were tested:

Prototype 1: Developed using Unity and Vuforia SDK, with datasets stored locally on the device.

Prototype 2: Built with Unity and a machine learning (ML) model, utilizing cloud storage for datasets. Both applications were designed to run on Android devices and aimed to overlay digital content onto real-world objects via the device camera.

Inanimate Objects: Statues from the National Museum of Egyptian Civilization (NMEC)(*National Museum OF Egyptian Civilization*, n.d.) were selected as test objects. These artifacts represent common museum exhibits and serve as ideal subjects for evaluating the application.

6.4.2 Testing Environment

Testing was conducted within the NMEC to replicate real-world usage. The museum provided a diverse array of artifacts and settings, creating an authentic environment for evaluating the application.

6.4.3 Testing Protocol:

Visitors who used the application completed a structured questionnaire. The survey measured application effectiveness, usability, and overall satisfaction, providing valuable insights into user preferences and experiences.

6.4.4 Testing Software Components and Units

Testing of software components was carried out in two stages: unit testing and system testing. This ensured a robust and functional application capable of delivering the intended augmented reality experience.

Unit testing focused on verifying the functionality of individual components in isolation. The results ensured that all components worked seamlessly before integration into the overall system. The key test cases evaluated include:

- **Start Button (Test Case ID: Unique_id_1001):** Verified that pressing the "Start" button launched the device's camera successfully.

Result: Pass. The camera interface opened without issues.

- **Exit Button (Test Case ID: Unique_id_1002):** Tested that pressing the "Exit" button closed the application gracefully without errors.

Result: Pass. The application terminated smoothly, releasing hardware resources.

- **Object Detection (Test Case ID: Unique_id_1003):** Validated the system's ability to detect objects and generate relevant virtual content, such as information displays and audio options.

Result: Pass. The application provided accurate detection and content output.

- **English Audio Button (Test Case ID: Unique_id_1004):** Checked the functionality of the English audio button, ensuring proper playback and termination when the object was removed from view.

Result: Pass. Audio played correctly and stopped as expected.

- **Arabic Audio Button (Test Case ID: Unique_id_1005):** Similar to the English button test, this case verified proper playback and termination for Arabic audio.

Result: Pass. Audio played and terminated correctly.

By identifying and resolving issues early, unit testing ensured that each component operated seamlessly before integrating them into the overall system.

6.5 System Testing

System testing evaluated the complete application for cohesive functionality and smooth user experience. It encompassed the following scenarios:

- UI interactions: Verified button responses and navigations.
- 3D model rendering: Ensured accuracy and proper placement.
- Audio playback: Checked consistency and proper functioning.

Results: The integrated system performed as expected, delivering a seamless and engaging AR experience for museum visitors.

7. User Experience and Feedback

Testing involved live interaction with museum visitors who engaged with both prototypes. Their experience was evaluated through structured questionnaires that captured feedback on the application's ease of use, accuracy, and overall satisfaction.

The results validated the system's functionality and highlighted its potential to enhance visitor engagement with museum exhibits, supporting the application's deployment in real-world scenarios.

The evaluation of the system included two primary components: demographic data collection and user experience analysis. These components provided insights into the system's usability, effectiveness, and visitor preferences.

7.1 Testing Protocol and Environment

7.1.1 Prototype 1: Vuforia

The first prototype, developed using Vuforia, demonstrated accurate object detection when artifacts were directly faced. However, its performance was sensitive to external factors such as lighting conditions and required precise camera positioning for optimal results, shown in Figure 21. This limitation highlighted the need for better adaptability to varying environmental conditions.

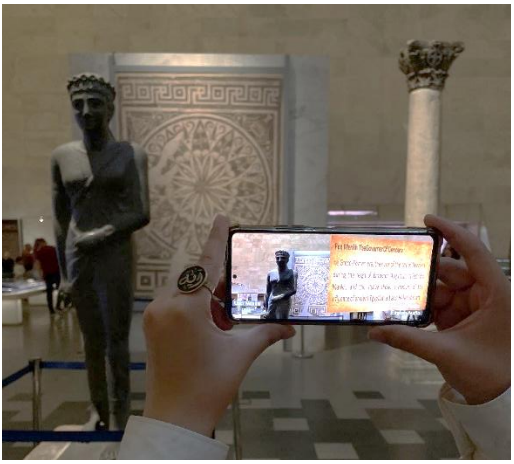


Figure 21: Testing the first prototype (Vuforia Model)

7.1.2 Prototype 2: ML

The second prototype, enhanced with a machine learning model, achieved superior detection accuracy while significantly reducing the application's size. Unlike the first prototype, the ML-based system efficiently identified objects without requiring specific camera angles, offering a more flexible and user-friendly experience, as illustrated in Figure 22.

Figure 22: Testing the second prototype (ML Model)

7.1.3 Comparative Testing

Figures 23 and 24 illustrate the testing of both prototypes on the same artifact, the "Female Peasant Statue." These figures visually compare their performance in detecting and presenting information about the artifact. While Vuforia struggled with multi-angle detection, the ML prototype consistently provided accurate results from various angles.

Figure 23: Testing the first prototype on The Female Peasant Statue

Figure 24: Testing the second prototype on The Female Peasant Statue

Table 1 summarizes the performance and technical specifications of both prototypes. The ML-based prototype outperformed Vuforia in key metrics, including application size, the number of image targets, and detection accuracy:

Table 1: Prototypes Comparison

Feature	Prototype 1 (Vuforia)	Prototype 2 (ML)
Application Size	435 MB	26.8 MB
Number of Image Targets	60 images	797 images
Dataset Size	294 MB	3.79 GB
Detection Accuracy	Front-only accuracy	Multi-angle

For more precise results, Table 2 compares the performance of the Vuforia SDK-based prototype (Prototype 1) and the Machine Learning-based prototype (Prototype 2):

Table 2: Performance metrics of the two prototypes

Metric	Prototype 1 (Vuforia)	Prototype 2 (ML)
Accuracy (%)	88 ± 2.1	97 ± 1.5
Precision (%)	89 ± 2.4	96 ± 1.8
Recall (%)	88 ± 2.0	94 ± 2.0
Binary Accuracy (%)	88 ± 2.2	97 ± 1.3
Loss	0.35 ± 0.05	0.03 ± 0.01

This comparison highlights the significant improvements achieved with the ML-based prototype in accuracy, precision, recall, binary accuracy, and loss reduction

The ML prototype's ability to handle a larger dataset and support multi-angle detection highlights its potential for broader application in museums.

7.1.4 User Evaluation

Feedback from users underscored the system's value in enhancing the museum experience:

- Immersive Experience: AR content provided users with an engaging and informative way to interact with artifacts, as shown in Figure 25.
- Accessibility: Dual-language support in English and Arabic made the application widely accessible.
- Recommendations: Users suggested refining the user interface and expanding language options to improve the overall experience.

Figure 25: Testing the application on artifacts.

7.2 User Feedback and Demographic Analysis

Participants were asked about their age and familiarity with Egyptian history to contextualize their feedback. The majority of participants were in the 18–24 age range, with a few under 18 and over 40. Familiarity with Egyptian history varied, ranging from "slightly familiar" to "expert."

7.2.1 Likelihood of Visiting Museums with AR

Participants rated how likely they were to visit a museum with its own AR application on a scale from 1 (very unlikely) to 5 (very likely). As shown in Figure 26, 42% indicated they were "very likely," and 33% responded "likely." This suggests a strong interest in AR-enhanced museum experiences.

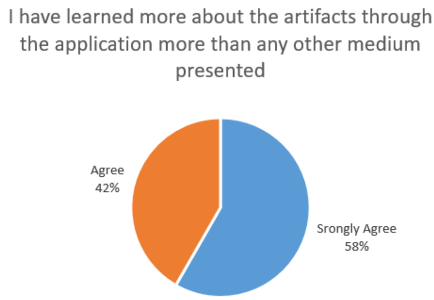


Figure 26: Pie chart that represents results of a question.

7.2.2 Learning Effectiveness

When asked if they learned more about artifacts through the application compared to other mediums, 58% "strongly agreed" and 42% "agreed" (see Figure 27). These results demonstrate the system's educational potential and its ability to enhance artifact comprehension.

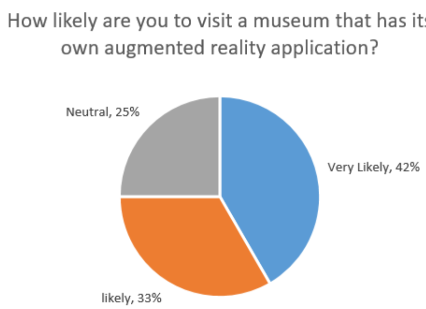


Figure 27: Pie chart that represents results of a question.

7.2.3 Preference for Virtual Guides

A significant majority (83.3%) preferred the virtual guide over the real guide, as illustrated in Figure 28. This highlights the system's effectiveness in providing an engaging and satisfactory virtual experience.

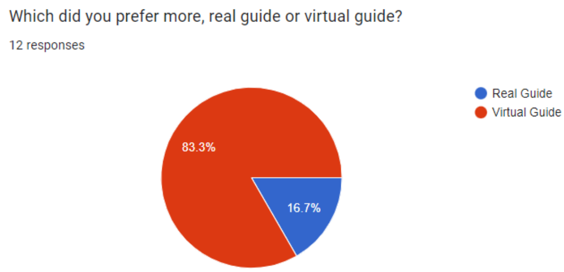
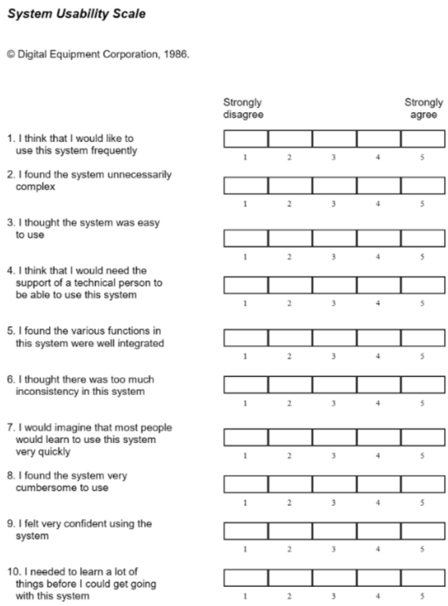


Figure 28: Pie chart that represents results of a question.

7.3 System Usability Scale (SUS)

The System Usability Scale (SUS) was used to evaluate the system's ease of use and overall usability. A 10-question SUS assessed the application's usability is presented in Figure 29. The mean SUS score across participants was 91.05, placing the system in the A+ grade (96–100 percentile) based on SUS grading (Figure 30). This indicates exceptional usability and user satisfaction.



System Usability Scale

© Digital Equipment Corporation, 1986.

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Figure 29: SUS Questionnaire(Lewis & Sauro, 2009)

SUS Score Range	Grade	Percentile Range
84.1 – 100	A+	96 – 100
80.8 – 84	A	90 – 95
78.9 – 80.7	A-	85 – 89
77.2 – 78.8	B+	80 – 84
74.1 – 77.1	B	70 – 79
72.6 – 74	B-	65 – 69
71.1 – 72.5	C+	60 – 64
65 – 71	C	41 – 59
63.7 – 64.9	C-	35 – 40
51.7 – 62.6	D	15 – 34
0 – 51.7	F	0 – 14

Figure 30: SUS grading scale (Lewis & Sauro, 2009)

The Quantitative Feedback is summarized in Table 3

Table 3: Quantitative Feedback Summary

Metric	Mean Score/Percentage	Interpretation
Likelihood of visiting AR enabled museum	4.42 (out of 5)	High interest in AR-enhanced experiences.
Artifact learning through AR	4.42 (out of 5)	Effective educational tool.
SUS Score	91.05	Excellent usability (A+ grade).

8. Conclusion

This project successfully developed an engaging augmented reality (AR) application for Android devices, designed to enhance the visitor experience at the National Museum of Egyptian Civilization (NMEC). By providing interactive virtual content on museum exhibits, the application bridges the gap between traditional displays and modern technology, offering visitors an immersive and educational journey.

The development process began with identifying the challenges faced by museums in engaging diverse audiences and exploring existing AR applications for cultural and historical sites. A comprehensive literature review highlighted tourism as an ideal sector for applying AR technology, with NMEC serving as the focus of this project. Two prototypes were developed: one using the Vuforia SDK for targeted object recognition and another leveraging a machine learning (ML) model for multi-angle, real-time artifact detection. Both prototypes were designed to recognize statues and artifacts, delivering historical and cultural information to enhance user interaction.

Testing revealed significant differences in the performance, scalability, and flexibility of the prototypes. Prototype 1 (Vuforia) was user-friendly but limited by static detection and environmental dependencies. Prototype 2 (ML) demonstrated superior accuracy, multi-angle detection, and scalability, emerging as the preferred solution for broader deployment.

Recommendations for future work include:

- Dataset Expansion: Broadening the ML model's dataset to include artifacts from other museums, extending the application's scope and relevance.

- Virtual Scene Integration: Adding full virtual scenes for outdoor exhibits, blending real and virtual environments for a fully immersive experience.

These enhancements will further adapt the AR application to diverse museum settings, making it more interactive, accessible, and appealing to a broader audience. The testing process validated the application's ability to transform museum experiences, demonstrating its potential to revolutionize how visitors engage with cultural heritage.

In conclusion, this project not only underscores the value of AR in cultural tourism but also establishes a scalable foundation for future applications. Prototype 2's performance highlights its readiness for deployment in museums worldwide, paving the way for innovative, technology-driven visitor experiences.

References

A celtic museum experience using augmented reality | ViMM. (n.d.). Retrieved 3 January 2025, from https://www.vi-mm.eu/2017/11/30/a-celtic-museum-experience-using-augmented-reality/

Alexander, J. (2014). Gallery one at the Cleveland Museum of Art. *The Museum Journal, Wiley Online Library*, *57*(3), 347–362.

Alves, C., & Luís Reis, J. (2020). The intention to use e-commerce using augmented reality - the case of IKEA place. *Advances in Intelligent Systems and*

Computing, *1137 AISC*, 114–123. https://doi.org/10.1007/978-3-030-40690-5_12/TABLES/2

ARKit 6 - Augmented Reality - Apple Developer. (n.d.). Retrieved 3 January 2025, from https://developer.apple.com/augmented-reality/arkit/

Breuss-Schneeweis, P. (2016). 'The speaking celt' - Augmented reality avatars guide through a museum - Case study. *UbiComp 2016 Adjunct - Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 1484–1491. https://doi.org/10.1145/2968219.2974044

Build new augmented reality experiences that seamlessly blend the digital and physical worlds | ARCore | Google for Developers. (n.d.). Retrieved 3 January 2025, from https://developers.google.com/ar

Ding, M. (2017). Augmented reality in museums. *Museums \& Augmented Reality--A Collection of Essays from the Arts Management and Technology Laboratory, Carnegie Mellon University*, 1–15.

Function. (n.d.). Retrieved 3 January 2025, from https://www.fxn.ai/

Galal, S. (2021). *Using Augmented Reality in Enhancing Museum Experience*.

Hamari, J., Malik, A., Koski, J., & Johri, A. (2019). Uses and Gratifications of Pokémon Go: Why do People Play Mobile Location-Based Augmented Reality Games? *International Journal of Human–Computer Interaction*, *35*(9), 804–819. https://doi.org/10.1080/10447318.2018.1497115

Introduction to Barracuda | Barracuda | 2.0.0. (n.d.). Retrieved 3 January 2025, from https://docs.unity3d.com/Packages/com.unity.barracuda@2.0/manual/index.html

Lewis, J. R., & Sauro, J. (2009). The Factor Structure of the System Usability Scale. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in*

Artificial Intelligence and Lecture Notes in Bioinformatics), 5619 LNCS, 94–103. https://doi.org/10.1007/978-3-642-02806-9_12

Mohammad, A. A. A., Sotohy, H. T. A., & Ammar, S. A. M. (2020). Toward a novel tourism experience: simulating ancient Egyptian lifestyle through tourism and hospitality services. *International Journal of Heritage, Tourism and Hospitality, 14*(1), 152–161. https://doi.org/10.21608/IJHTH.2020.126162

National Museum OF Egyptian Civilization. (n.d.). Retrieved 3 January 2025, from https://www.kaggle.com/datasets/reemkhaled/national-museum-of-egyptian-civilization

ONNX | Home. (n.d.). Retrieved 3 January 2025, from https://onnx.ai/

Ozturkcan, S. (2021). Service innovation: Using augmented reality in the IKEA Place app. *Journal of Information Technology Teaching Cases, 11*(1), 8–13. https://doi.org/10.1177/2043886920947110/ASSET/IMAGES/LARGE/10.1177_2043886920947110-FIG1.JPEG

Perkenalan Situs Slot Gacor - Perkenalan situs slot gacor yang sedang lagi malak - malaknya. (n.d.). Retrieved 3 January 2025, from https://artoolkit.org/

Rauschnabel, P. A., Rossmann, A., & tom Dieck, M. C. (2017). An adoption framework for mobile augmented reality games: The case of Pokémon Go. *Computers in Human Behavior, 76*, 276–286. https://doi.org/10.1016/J.CHB.2017.07.030

Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. https://arxiv.org/abs/1409.1556v6

Stenroos, O. (2017). *Object detection from images using convolutional neural networks*.

TensorFlow. (n.d.). Retrieved 3 January 2025, from https://www.tensorflow.org/

Unity - Scripting API: (n.d.). Retrieved 3 January 2025, from https://docs.unity3d.com/ScriptReference/

Unity Real-Time Development Platform | 3D, 2D, VR & AR Engine. (n.d.). Retrieved 3 January 2025, from https://unity.com/

Unity UI (uGUI) | Unity UI | 2.0.0. (n.d.). Retrieved 3 January 2025, from https://docs.unity3d.com/Packages/com.unity.ugui@2.0/manual/index.html

Vuforia Enterprise Augmented Reality (AR) Software | PTC. (n.d.). Retrieved 3 January 2025, from https://www.ptc.com/en/products/vuforia

Why are Convolutional Neural Networks good for image classification? | by Prafful Mishra | DataDrivenInvestor. (n.d.). Retrieved 3 January 2025, from https://medium.datadriveninvestor.com/why-are-convolutional-neural-networks-good-for-image-classification-146ec6e865e8

Wikitude Augmented Reality: the World's Leading Cross-Platform AR SDK. (n.d.). Retrieved 3 January 2025, from https://www.wikitude.com/