iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: IV Month of publication: April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79533

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Augmented Reality Virtual Try-On Web Application

Vericharla Ratnasri[1], Komatla Naga Lakshmi[2], Atche Krishnaveni[3], Dandasi Dhanalakshmi[4]
[1,2,3,4]*Department of Computer Science and Engineering, Bapatla Women's Engineering College, Affiliated to Acharya Nagarjuna University, Approved by AICTE – New Delhi, Andhra Pradesh, India*

Abstract: This paper presents an Augmented Reality (AR) based Virtual Try-On Web Application that allows users to visualize products such as clothing and accessories in real time using their device camera. The system leverages computer vision and machine learning techniques to detect human body landmarks and overlay virtual objects accurately on the user. The proposed solution enhances the online shopping experience by reducing uncertainty, improving customer satisfaction, and minimizing product returns. The application is developed using modern web technologies integrated with AR frameworks, enabling real-time rendering, object alignment, and interactive visualization. Experimental results demonstrate that the system provides efficient performance, accurate alignment, and improved usability, making it suitable for integration into modern e-commerce platforms.
Keywords: Augmented Reality, Virtual Try-On, Computer Vision, Web Application, E-commerce, Real-time Rendering, Machine Learning.

I. INTRODUCTION

The rapid growth of e-commerce platforms has significantly transformed the global retail industry, enabling consumers to purchase products anytime and anywhere [1]. With the increasing penetration of smartphones and internet accessibility, online shopping has become a preferred choice due to its convenience, wide product variety, and competitive pricing [2]. Despite these advantages, one of the major challenges faced by e-commerce platforms is the inability of customers to physically interact with or try products before making a purchase decision [3]. This limitation often results in uncertainty, lack of confidence, and higher product return rates, especially in categories such as fashion and accessories [4].

To address these challenges, Augmented Reality (AR) has emerged as a promising technology that enhances user interaction by overlaying virtual elements onto the real-world environment [5]. AR-based Virtual Try-On systems enable users to visualize how products such as clothing, eyewear, jewelry, and cosmetics would appear on them in real time using a camera-enabled device [6]. This immersive experience bridges the gap between physical and online shopping, thereby improving customer satisfaction and reducing return rates [7].

In recent years, advancements in computer vision and machine learning have significantly improved the accuracy and efficiency of AR applications [8]. Techniques such as facial landmark detection, pose estimation, and object tracking allow precise alignment of virtual objects with the user's body or face [9]. Frameworks like MediaPipe and OpenPose have made it easier to implement real-time detection of key points, enabling robust and scalable AR systems [10].

Furthermore, the evolution of web technologies such as WebXR and Three.js has enabled the development of browser-based AR applications without requiring specialized hardware or dedicated mobile applications [11]. These technologies allow developers to create lightweight, cross-platform solutions that can run directly in web browsers, making AR more accessible to a wider audience [12]. This advancement is particularly beneficial for e-commerce platforms, as it eliminates the need for additional installations and enhances user engagement. The proposed work focuses on designing and developing a web-based Augmented Reality Virtual Try-On system that is efficient, scalable, and user-friendly. The system integrates real-time video processing, feature detection, and AR rendering to provide an interactive and seamless experience. By enabling users to visualize products virtually, the system aims to improve decision-making, enhance customer satisfaction, and reduce product return rates. Additionally, the solution emphasizes accessibility across devices and browsers, making it suitable for integration into modern e-commerce ecosystems.

II. SYSTEM DESIGN

The proposed Augmented Reality Virtual Try-On Web Application is designed to provide a seamless, real-time, and interactive user experience. The system architecture follows a modular approach, where each component performs a specific function to ensure efficiency, scalability, and accuracy. The overall workflow involves capturing user input, processing visual data, detecting features, rendering virtual objects, and displaying augmented output.

A. *Input Data Representation*

The system processes real-time video frames captured from the user's device. Each frame is represented as a set of pixel values and extracted features:

$$Fi = (Xi, Yi)i = 1,2,3,\ldots, N \quad (1)$$

where F_i represents the detected landmark points in the frame, and (X_i, Y_i) denote the 2D coordinates of facial or body key points. These landmarks include eyes, nose, ears, shoulders, etc., which are essential for accurate placement of virtual objects.

B. *Feature Detection and Landmark Extraction*

The Feature Detection Module uses computer vision models to extract key landmarks from the input frame. The mapping from image space to feature space can be defined as:

$$L=f(I) \quad (2)$$

where I represents the input image frame and L represents the set of detected landmarks.

The accuracy of landmark detection directly affects the realism of the virtual try-on system.

C. *Geometric Transformation for Object Alignment*

To correctly place virtual objects on the user, geometric transformations such as scaling, translation, and rotation are applied to ensure proper alignment with detected facial or body landmarks. These transformations allow the virtual object to dynamically adapt to the user's position, orientation, and movement in real time

The scaling transformation is used to adjust the size of the virtual object based on the relative distance between detected landmarks. It is defined as:

$$S = \frac{d_{real}}{d_{detected}} \quad (3)$$

where d_{real} represents the actual dimension of the object and $d_{detected}$ denotes the distance between key detected landmarks (such as the distance between eyes or shoulders). This ensures that the virtual object appears proportionate to the user's physical features.

In addition to scaling, translation is applied to position the virtual object correctly on the user's face or body. The translation vector is expressed as:

$$T=(xt,yt) \quad (4)$$

where (x_t, y_t) represents the shift required along the horizontal and vertical axes to align the virtual object with the detected landmark coordinates. This step ensures that the object is placed at the correct spatial location.

D. *AR Rendering Engine*

The AR Rendering Engine is responsible for superimposing virtual objects onto the detected facial or body landmarks in real time. This module utilizes rendering libraries such as Three.js and WebXR to generate and manipulate 2D/3D models, ensuring realistic integration with the live camera feed. The rendering process takes into account geometric transformations such as scaling, rotation, and translation computed in the previous stage.

The augmented frame is generated by combining the input frame with the rendered virtual object based on the detected landmark positions. This process can be mathematically represented as:

$$F_{out} = R(F_{in}, L, T) \quad (5)$$

where F_{in} represents the input frame, L denotes the set of detected landmarks, T represents the transformation parameters (scaling, rotation, translation), and R is the rendering function that overlays the virtual object onto the real-world frame.

This formulation ensures that the virtual object dynamically adapts to user movements, maintaining alignment and visual consistency throughout the interaction.

E. *Output Display*

The Output Display module is responsible for presenting the augmented frames to the user in real time. The processed frames are continuously rendered and displayed through the web interface, ensuring smooth visualization with minimal latency. Efficient frame handling and rendering optimization techniques are employed to maintain high performance across devices.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

The performance of the output display can be evaluated in terms of frame rate, defined as:

$$FPS = TNframes \qquad (6)$$

where N_{frames} represents the total number of frames rendered and T denotes the total time taken. A higher frame rate indicates smoother user experience and better system responsiveness.

The system is designed to minimize delay between input capture and output rendering, thereby ensuring real-time interaction. This enables users to seamlessly visualize virtual objects with natural movements, enhancing the overall usability and effectiveness of the virtual try-on application.

III. ALGORITHMS AND FLOWCHART

A. Explanation of the Flowchart

The flowchart of the proposed system illustrates the complete workflow of the Augmented Reality Virtual Try-On Web Application. The process begins with capturing real-time video input from the user through a web camera. The captured frames are then passed through preprocessing steps such as noise reduction, resizing, and normalization to enhance image quality.

After feature extraction, the system proceeds with object detection and segmentation, where the selected virtual attire is prepared. The attire image is processed by removing the background and converting it into a transparent format suitable for overlay.

The next stage involves scaling, warping, and alignment of the virtual object. The size of the attire is dynamically adjusted based on the user's body measurements, and geometric transformations are applied to match the user's pose and perspective.

Following this, the AR rendering engine overlays the transformed virtual object onto the user's body. Techniques such as image blending, homography, and alpha compositing are used to ensure a realistic appearance.

Finally, the augmented frames are displayed to the user in real time through the user interface. The system continuously updates frames, ensuring smooth interaction and minimal latency.

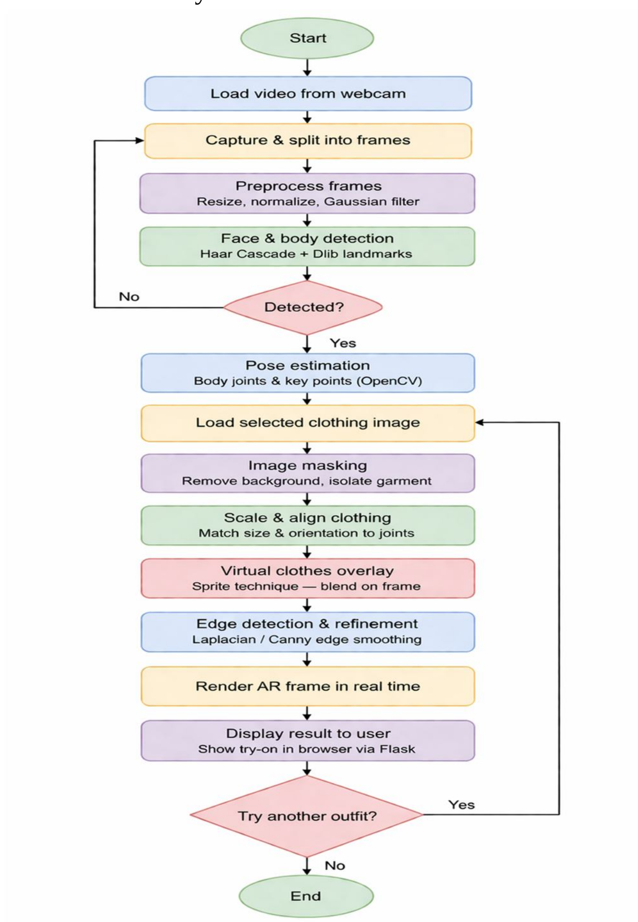

Figure.1. AR-Based Virtual Try-On System Flowchart

B. Pseudocode

1) Algorithm 1 (Image Acquisition)

This algorithm captures real-time video input from the user and performs preprocessing to enhance image quality. The raw frames obtained from the webcam may contain noise, variations in lighting, and unnecessary background information. Therefore, filtering techniques such as Gaussian smoothing and normalization are applied. The processed frames are then prepared for further analysis such as feature detection and pose estimation.

```
Algorithm 1: PreprocessFrames( V )

Input: Video stream V from webcam
Output: Preprocessed frames Fp

Initialize camera device
while video stream V is active do
   F ← CaptureFrame(V)
   G ← ConvertToGrayScale(F)
   Gf ← ApplyGaussianFilter(G)
   Fn ← NormalizeImage(Gf)
   Fp.append(Fn)
end while

return Fp
```

2) Algorithm 2: Face and Body Feature Detection

This algorithm detects the user’s face and body landmarks using trained classifiers and pose estimation techniques. Haar Cascade classifiers are used for face detection, while landmark detection models identify key points such as eyes, nose, shoulders, and torso. These features serve as reference points for aligning virtual objects accurately on the user’s body.

```
Input: Preprocessed frames Fp, classifier C, pose model P
Output: Feature points set K

for each frame f in Fp do
   FaceBox ← DetectFace(f, C)
   if FaceBox ≠ NULL then
      Landmarks ← ExtractLandmarks(f, P)
      Shoulders ← DetectShoulders(Landmarks)
      Torso ← EstimateTorso(Landmarks)
      K.append({FaceBox, Landmarks, Shoulders, Torso})
   end if
end for

return K
```

3) Algorithm 3 (Virtual Attire Scaling and Transformation)

This algorithm adjusts the size, orientation, and position of the selected virtual attire according to the detected body features. Scaling is performed dynamically based on body dimensions, and geometric transformations such as rotation and warping are applied to match the user’s posture and movement.

```
Input: Attire image A, feature points K
Output: Transformed attire At

for each feature set k in K do
   BodyWidth ← ComputeDistance(k.Shoulders)
```

```
    ScaleFactor ← BodyWidth / Width(A)

    As ← Resize(A, ScaleFactor)
    Angle ← ComputeInclination(k.Landmarks)
    Ar ← Rotate(As, Angle)

    Aw ← ApplyWarping(Ar, k.Torso)
    At.append(Aw)
end for
return At
```

4) Algorithm 4 (AR Overlay and Real-Time Rendering)

This algorithm overlays the transformed attire onto the user's frame using augmented reality techniques. Alpha blending and compositing ensure that the virtual object integrates naturally with the real-world image. The final augmented frames are displayed in real time with minimal latency.

Input: Original frames F, transformed attire At

Output: Augmented frames Fa

```
for each frame f in F do
    a ← GetCorrespondingAttire(At)
    a_rgba ← ConvertToRGBA(a)

    Fa_frame ← AlphaBlend(f, a_rgba)
    Fa.append(Fa_frame)

    Display(Fa_frame)
end for
return Fa
```

IV. RESULTS AND DISCUSSION

The proposed AR-based Virtual Try-On system was evaluated across multiple stages of the processing pipeline. Results demonstrate the effectiveness of each component, from initial video acquisition through final augmented output display, under standard operating conditions.

A. *Initial Web Application Interface*

The system begins with the web-based user interface, which acts as the entry point for the virtual try-on application. The interface is designed to be simple and user-friendly, allowing users to easily navigate through the application. It provides options such as starting the camera, selecting products, and interacting with the system. This stage ensures a smooth user experience before initiating the augmented reality process.

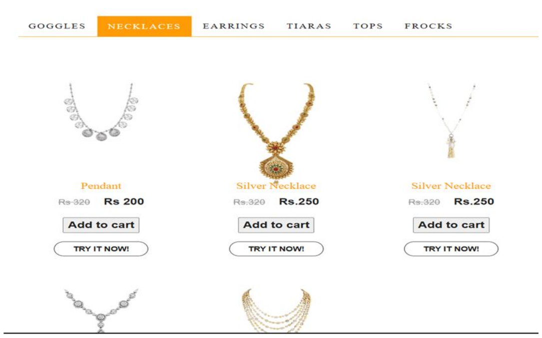

Figure.2. Initial Web Application Interface for Virtual Try-On System.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

B. Product Selection Interface

After accessing the application, users are provided with options to select the desired clothing or accessories. This interface allows users to browse through different categories and choose the item they want to try on. The selected product is then prepared for further processing and overlay.

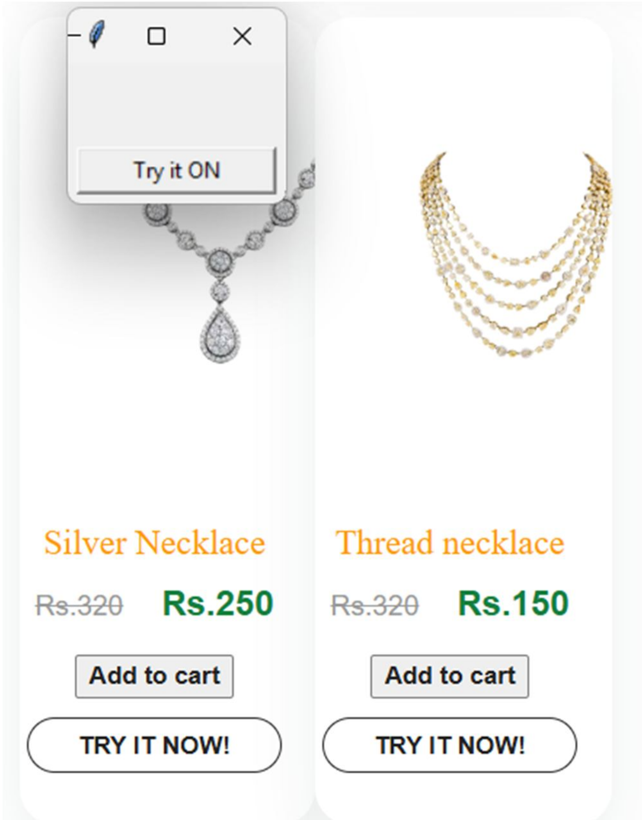

Figure.3. Product Selection Interface for Choosing Virtual Attire.

C. Webcam Input Frame

Once the product is selected, the system activates the webcam to capture real-time video input. The captured frames serve as the raw input for further processing. The system ensures continuous frame acquisition for smooth real-time performance.

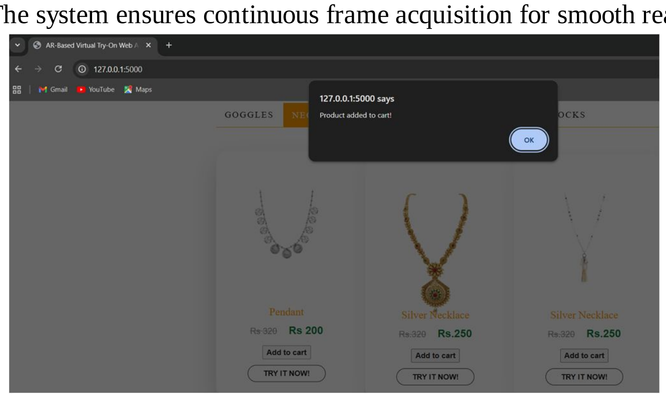

Figure.4. Input Frame Captured from Webcam.

D. Preprocessing and Noise Reduction

The captured frames undergo preprocessing steps such as grayscale conversion, Gaussian filtering, and normalization. These techniques help in reducing noise and improving image quality, which is essential for accurate detection of facial and body features.

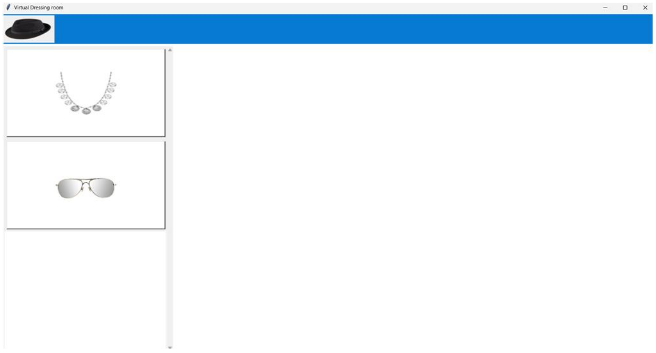

Figure.5. Preprocessed Frame after Noise Reduction.

E. Face and Body Detection

The system applies computer vision techniques such as Haar Cascade classifiers and pose estimation algorithms to detect the face and body landmarks. Key features like eyes, nose, shoulders, and torso are identified, which act as reference points for positioning the virtual attire.

Figure.6. Face Detection Stage of the System

F. Attire Scaling and Transformation

Based on the detected body features, the virtual attire is dynamically resized and aligned. Scaling factors are computed using distances between body landmarks, and transformations such as rotation and warping are applied to match the user's posture and orientation. Figure 6: Scaling and Transformation of Virtual Attire.

G. Augmented Reality Overlay and Final Output

The transformed attire is overlaid onto the user's body using augmented reality techniques such as alpha blending and image compositing. The final output is displayed in real time through the user interface, allowing users to view themselves wearing the selected attire interactively. The system ensures smooth rendering with minimal delay, providing a realistic and immersive experience.

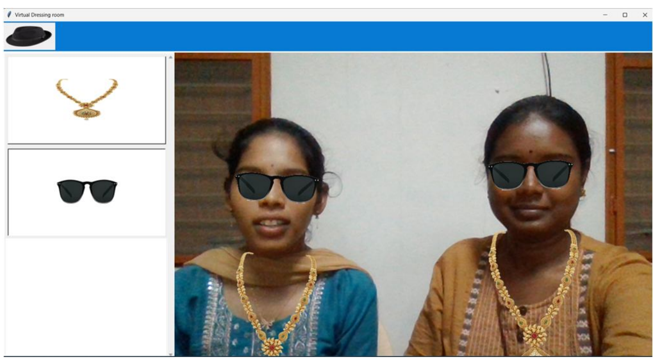

Figure.7. AR Overlay Result – Eyewear Try-On.

Figure.8. AR Overlay Result – Clothing Try-On.

These The results demonstrate that the proposed system successfully integrates computer vision and augmented reality techniques to provide a seamless virtual try-on experience. The system performs effectively under normal lighting conditions and maintains real-time responsiveness. The detection and alignment of virtual attire are accurate in most cases, ensuring a realistic appearance.
However, certain limitations were observed, such as reduced accuracy in low lighting conditions and slight misalignment when the user moves rapidly. These challenges can be addressed in future work by incorporating advanced deep learning models and improved tracking algorithms.
Overall, the system achieves its objective of providing an efficient and user-friendly virtual try-on solution, which can be highly beneficial in e-commerce and fashion applications.

V. CONCLUSION

A. *Conclusion*

In this study, an Augmented Reality Based Virtual Try-On Web Application was designed and developed to provide an interactive and real-time virtual dressing experience. The system integrates computer vision and augmented reality techniques to capture user input, process images, and accurately detect facial and body features.
The application successfully performs preprocessing, feature detection, and virtual attire alignment using scaling and transformation techniques. The implementation of AR overlay enables realistic visualization of selected clothing on the user's body. Additionally, the user-friendly web interface allows smooth interaction, from product selection to real-time output display.
The experimental results indicate that the system achieves reliable performance under normal conditions, with accurate detection and minimal latency. The proposed approach demonstrates significant potential in enhancing user experience in online shopping and fashion applications by reducing the need for physical trials.

B. *Future Work*

Despite the effectiveness of the system, several enhancements can be considered for future development. The integration of advanced deep learning techniques, such as Convolutional Neural Networks (CNNs), can significantly improve the accuracy of face and body detection as well as tracking performance. Furthermore, extending the system to support 3D virtual try-on using three-dimensional clothing models can provide a more realistic and immersive user experience compared to the current 2D overlay approach. In addition, improving the robustness of the system under varying lighting conditions and handling occlusions effectively will enhance its reliability in real-world scenarios. The development of a mobile-friendly application can further increase accessibility, allowing users to utilize the system across different devices conveniently. Incorporating an AI-based recommendation system can also add value by suggesting suita ble outfits based on user preferences, trends, and past selections.
Moreover, enabling multi-user support will allow the system to handle multiple users simultaneously, thereby improving scalability. Finally, integrating the application with e-commerce platforms can provide a seamless experience where users can directly purchase the selected products after trying them virtually. These enhancements will make the system more efficient, scalable, and suitable for real-world commercial applications.

REFERENCES

[1] P. Milgram and F. Kishino, "A Taxonomy of Mixed Reality Visual Displays," IEICE Transactions on Information Systems, vol. E77-D, no. 12, pp. 1321–1329, 1994.

[2] R. Azuma, "A Survey of Augmented Reality," Presence: Teleoperators and Virtual Environments, vol. 6, no. 4, pp. 355–385, 1997, doi: 10.1162/pres.1997.6.4.355.
[3] S. S. Kannan, R. Mohan, and P. R. Kumar, "Virtual Trial Room Using Augmented Reality," International Journal of Advanced Computer Technology (IJACT), vol. 9, no. 2, pp. 45–50, 2020.
[4] M. Zollhöfer et al., "State of the Art on Monocular 3D Face Reconstruction, Tracking, and Applications," Computer Graphics Forum, vol. 37, no. 2, pp. 523–550, 2018.
[5] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," in Proc. IEEE CVPR, 2017, pp. 7291–7299.
[6] D. C. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification," in Proc. IEEE CVPR, 2012.
[7] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.
[8] T. Kanade, "Picture Processing System by Computer Complex and Recognition of Human Faces," Kyoto University, 1973.
[9] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic Generation and Detection of Highly Reliable Fiducial Markers," Pattern Recognition, vol. 47, no. 6, pp. 2280–2292, 2014.
[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in Proc. IEEE CVPR, 2016.
[11] A. Sanna and F. Manuri, "A Survey on Applications of Augmented Reality," Advances in Computer Science: an International Journal, vol. 5, no. 1, pp. 18–27, 2016.
[12] Three.js Documentation, "JavaScript 3D Library," Available: https://threejs.org/.
[13] Blender Online Community, "Blender - a 3D Modelling and Rendering Package," Blender Foundation, Available: https://www.blender.org/.
[14] Microsoft Corporation, "Kinect for Windows SDK," Available: https://developer.microsoft.com/en-us/windows/kinect/.