Journal of Applied Data Sciences
Vol. 6, No. 1, January 2025, pp. 259-271
ISSN 2723-6471
259

Deep Learning Incorporated with Augmented Reality Application for Watch Try-On

Andri Andri[1,*], Tri Basuki Kurniawan[2], Deshinta Arrova Dewi[3], Mashal Kasem Alqudah[4], Musab Kasim Alqudah[5] Mohd Zaki Zakaria[6], Putri Aisha Athira binti Hisham[7]

[1]*Faculty of Science Technology, Universitas Bina Darma, Palembang, Indonesia*

[2]*Postgraduate Program, Universitas Bina Darma, Palembang, Indonesia*

[3]*Faculty of Data Science and Information Technology, INTI International University, Malaysia*

[4]*Institute of Public Administration, Digital Transformation and Information Programs Department, Riyadh, Saudi Arabia*

[5]*College of Science and Computer Engineering, Yanbu, Taibah University, Yanbu, Al-Madinah Al-Munawwarah, Saudi Arabia*

[6,7]*Faculty of Computer and Mathematics Sciences, University Technology Mara, Malaysia*

(Received: September 21, 2024; Revised: October 13, 2024; Accepted: November 11, 2024; Available online: December 28, 2024)

Abstract

In evaluating the dynamic landscape of online shopping, the integration of Augmented Reality (AR) technologies has emerged as a transformative force, redefining the way consumers engage with products in virtual environments. This research project investigates the intersection of deep learning and AR in the context of online shopping, with a particular focus on a Watch Try-On application. The experimentation involves the use of SSD MobileNet's models for real-time object detection aimed at enhancing the user experience during online watch shopping. Training both SSD MobileNet's V1 and V2 models through 50,000 iterations, the results reveal intriguing insights into their performance. SSD MobileNet's V1 demonstrated superior results, boasting a mean average precision (mAP) of 0.9725 and a significant reduction in total loss from 0.774 to 0.5405. However, the longer training time of 7 hours and 42 minutes prompted the selection of SSD MobileNet's V2 for real-time applications due to its faster inference capabilities. Extending beyond traditional online shopping experiences, the research explores the potential of AR technologies to revolutionize product visualization and interaction. The choice of the Vuforia model target for the Watch Try-On application showcases the synergy between deep learning and AR, allowing users to virtually try on watches and visualize them in their real-world environment. The application successfully detects users' hands with high accuracy, creating an immersive and visually enriching experience. In conclusion, this project contributes to the ongoing discourse on the fusion of deep learning and AR for online shopping. The exploration of SSD MobileNet's models, coupled with the integration of AR technologies, underscores the potential to elevate the online shopping experience by providing users with dynamic, interactive, and personalized ways to engage with products.

Keywords: Augmented Reality (AR), Deep Learning, SSD MobileNet, Watch Try-On Application, Process Innovation

1. Introduction

The fashion industry has been rapidly adopting technology to enhance their customers' shopping experiences. Augmented Reality (AR) has emerged as a game-changing tool in the fashion industry, allowing customers to virtually try on clothing and accessories. Worldwide retailers are contributing to making the future of the shopping experience by implanting Virtual Reality and Augmented Reality into the e-commerce packages, name a few, Amazon's VR kiosk is making consumers visit cities full of products with the help of Oculus Rift. Alibaba's Buy technology allows consumers to combine offline and online shopping experiences and provide a brick-and-mortar store shopping experience; eBay's Virtual Reality department store creates a next level of shopping experience for consumers, Ikea's 3D showroom provides a new way for visualization and exploration of products. In this research [1], we will explore the potential of AR for the watch industry, specifically the concept of an augmented reality watch try-on.

***Corresponding author: Andri Andri (andri@binadarma.ac.id)**
DOI: https://doi.org/10.47738/jads.v6i1.529

One of the biggest challenges in the watch industry is the inability to try on watches before purchase. Customers prefer to see how the watch looks and feels on their wrist before making a purchase decision. This has resulted in a lower conversion rate for online watch purchases. The year 2020 was extremely difficult for luxury retail overall, and certainly, the watch industry has not been spared. Figures from the Federation of the Swiss Watch Industry (FH) show that Swiss watch exports for the year totaled just under 16.9 billion Swiss francs, a 21.8% drop from 2019. The 13.7 million units exported during the same period represent a 33.3% decline for the same period. However, with the advancements in AR technology, it is now possible to create a virtual try-on experience for watches [2].

This research focuses on enhancing the user experience of online shopping for watches through augmented reality (AR) watch try-ons. The aim is to provide customers with a realistic and interactive way to try virtual watches before making a purchase decision. The software tools utilized for this purpose are Vuforia and Unity.

The dataset used in this research consists of watch models developed in Unity. These models will be created specifically for this study, ensuring an accurate representation of various watch designs and details. The augmented reality experience will be facilitated by the Vuforia software, which enables the detection and tracking of the user's environment using markers or image recognition.

Augmented reality watch try-ons offer significant advantages for online shoppers. It allows customers to visualize and evaluate how different watches would look and fit on their wrists in real time without physically trying them on. This immersive experience can enhance customer satisfaction, reduce return rates, and ultimately improve the online shopping experience for watches. The problem inside the data set is carefully described in the problem statement, which can then lead to inquiries about the project. Fulfilling the goals of this research will allow the research questions to be answered. The scope of this study contributes to its particular significance.

2. Augmented Reality Research

With a predicted CAGR of 4.8% from 2021 to 2026, the global wristwatch market is expected to increase from USD 62.8 billion in 2021 to USD 78.2 billion by 2026 [3]. One of the most popular fashion items now worn by both adults and children is wristwatches. The younger generation's expanding demand for fashion accessories, along with rising disposable income levels in developed and emerging countries, have been the main drivers of the expansion of the world watch market. The biggest watch market trends for 2021 have been recent technological advancements, including the creation of smartwatches by businesses like Apple and Samsung with features like fitness trackers, heart rate monitors, and smartphone connectivity. The examination of the global watch market reveals that more businesses are starting to enter the smartwatch sector, which will benefit the wristwatch market throughout the projected period.

2.1. Augmented Reality

Augmented reality (AR) is an interactive experience that combines the real world and computer-generated content. The content can be presented in a variety of ways, including visually, aurally, haptically, somatosensorily, and olfactorily [4]. AR can be known as a system that combines the real and virtual worlds with real-time interaction and precise 3D registration of actual and virtual items [3]. This experience is so completely integrated with the real world that it appears to be a realistic component of the setting. In contrast to virtual reality, which replaces the user's real-world environment with a simulated one, augmented reality modifies one's continuous perspective of a real-world environment. The main benefit of augmented reality is how elements of the digital world are integrated into how people experience the actual world, not just as a display of data but also through the incorporation of immersive experiences that are viewed as organic elements of a setting. The Virtual Fixtures system created by the U.S. Air Force's Armstrong Laboratory in 1992 was one of the first practical AR systems to offer users immersive mixed-reality experiences in the early 1990s [5]. Commercial augmented reality applications first appeared in the gaming and entertainment industries.

2.2. Marker-based AR

Marker-based augmented reality, also known as image recognition augmented reality, relies on a fiducial marker, such as a QR code, to start the interactive experience. The visual effects are activated when a customer uses their smartphone camera to scan the marker. To view the digital image in 3D on their screen, they can then move their mobile device around the stationary marker. The main drawback of marker-based augmented reality is that it can only be used with mobile devices, such as smartphones or tablets, and that users may need to download a special program (such as Google Play Services for AR for Android devices. Users can view furniture at home before purchasing it with the IKEA Place app. It scans the user's environment with a smartphone or tablet camera to find flat surfaces like a table or floor. Using digital image processing techniques, markers carrying environmental information can be detected in a frame of the camera. Many of the practical systems use two-dimensional planar markers, which are in a manner very similar to the bar code on the product in supermarkets [6].

2.3. Markerless AR

Markerless augmented reality (AR) does not rely on external markers like a QR code or image. Instead, it tracks the user's environment and locates the virtual content using location-based data from devices like GPS and accelerometers. The program can then overlay the virtual content by the spatial relationships and orientation of the objects and surfaces in the user's field of view. Due to its simplicity and flexibility, markerless AR is typically more difficult and expensive to set up, but it is also the most used alternative in online gaming and commerce. Three further types of AR are also possible without markers: projection-based AR, superimposition-based AR, and location-based AR. One example of markerless augmented reality is virtual shoe placement in a scene using markerless AR; users can try on virtual shoes and see how they look and fit in their real-life environment without the need for physical markers. In recent years, markerless augmented reality has emerged as a distinct category in the field of reality augmentation [7].

2.4. Projection-based AR

Projectors are used to project light onto actual surfaces in projection-based augmented reality. Data, which can be anything from colors and text to enhanced 3D models, is included in projected light. In addition to projecting light onto the surface, the projector is also driven by a sensor that detects human-surface interaction and the human touch of the projected light. Four Kinect sensors positioned on the ceiling at the center of each room wall serve as the infrastructure-based sensors used by these unusual projectors to track their position. Additionally, by modeling the surroundings using these sensors, geometric awareness is also provided [8].

2.5. Location-based AR

A version of markerless AR called location-based AR uses geographic information to deliver digital images to precise areas. For example, Pokémon Go uses location-based AR, which is a popular sort of AR for gaming. Location-based AR might be used by brands to gamify the shopping experience by enticing customers to interact with their goods. You could, for instance, design a digital treasure hunt that encourages customers to explore your business and earn incentives. Most often, location-based AR is utilized for AR location browsers, which assist users in finding interesting places nearby. By reading information from the mobile device's GPS, digital compass, and accelerometer and estimating where the user is looking, this approach determines the user's location and orientation. It then adds relevant information about the items that the camera can see on the screen [9].

2.6. Vuforia as an AR Software development kit (SDK)

Vuforia Engine is a software development kit (SDK) for creating Augmented Reality apps. With the SDK, you add advanced computer vision functionality to your application, allowing it to recognize images, objects, and spaces with intuitive options to configure your app to interact with the real world. Vuforia's ability to register images enables developers to position and orient a virtual object, such as 2D or 3D objects or other types of media in the space, related to real-world images or video when these are viewed through the camera of the handheld devices. The virtual object can then follow the position and orientation of the real image in real-time, allowing the viewer's perspective to match that of the target in the real world. In this way, the virtual object mimics the appearance of a different real-world object. Different target types, both 2D and 3D, including multi-target configurations, markerless image targets, and frame markers, are supported by the Vuforia SDK.

2.7. Unity Engine

A cross-platform game engine known as Unity was created by UnityTechnologies and debuted as an exclusive game engine (Mac OS X) at the Worldwide Apple Inc. Conference in June 2005. More than 20 platforms were made possible by the engine's extension in 2018. This game engine can be used to create games that combine simulations with augmented reality, virtual reality, two- and three-dimensional environments, and more. In the twenty-first century, industries other than video gaming, including film, architecture, automotive, construction, and engineering, have used this game engine. Unity, a multi-platform game engine, is commercially available and is used for 2d and 3D video game production accompanied by visualizations and non-game interactive simulations. Additionally, because of its simplicity, adaptability, efficiency, and low power consumption, Unity is one of the most widely used and accessible game engines in the present day. The Unity Editor has a variety of features that allow for quick iteration and editing during development cycles, including smart previews and real-time playback. Additionally, Unity is accessible on Mac, Linux, and Windows, and it comes with a robust set of developer tools for implementing high-performance gameplay and game logic, in addition to a range of artist-friendly tools for designing and building immersive game environments [10].

3. Methodology

The research challenge will be approached systematically using established research techniques. The process can be seen as a scientific investigation into how research is conducted, examining the various approaches commonly employed by researchers to analyze their research problems and the rationale behind them. The framework of the whole of the research methodologies and procedures a researcher selects to carry out a study is known as the research design. The layout enables researchers to focus on developing research techniques applicable to the topic and arrange our investigations for success. It contains objectives, phases, activities, sources, and deliverables that should be worked on specifically for each of the numerous parts.

Knowledge acquisition involves the extraction, classification, and analysis of information from various sources, including human experts, as well as academic databases such as Google Scholar, IEEE Xplore, Scopus, and Web of Science [11]. In this project, the literature analysis technique was employed to identify the research problem and determine its purpose, scope, and significance about the relevant topic. Data labeling is a crucial step in the process of training machine learning models, particularly for supervised learning tasks. It involves annotating or tagging raw data to provide the necessary context for a machine learning algorithm to learn and make accurate predictions. The labeled data serves as the ground truth that the model uses to identify patterns and relationships.

Data preprocessing is a fundamental step in the data science and machine learning pipeline [12]. It involves cleaning and transforming raw data into a format suitable for analysis or training machine learning models. Proper data preprocessing can significantly impact the performance and accuracy of models. In this project, two datasets need to be cleaned, which is the egohand dataset consisting of 4800 that we just resized to the desired size from 416x416 to 320x320. Next, for the 3D hand cad model, Steps like Model Up Vector where the suitable vector was chosen to fit the desired camera angle when tracking and implementing the AR 3D watch model to the track user’s hand.

3.1. System Development

The development phase involves the actual coding or programming of the system based on the design specifications and requirements identified in the earlier stages [13]. The development team translates the system design into executable code guided by different practices from various Lean and Agile methods [14], [15]. Firstly, the design system architecture is established, which outlines the overall structure and components of the system. This includes defining the various modules and their interactions, ensuring seamless integration between different elements of the application.

Next, the development of a 3D watch model takes place using Unity, a popular game development platform. This involves creating a realistic virtual representation of the watches that users will be able to try on. The 3D models are carefully designed to capture the intricate details and aesthetics of each watch, ensuring an authentic experience. To enable marker-based augmented reality, a machine learning technique is developed. This technique uses markers as reference points to accurately position and track the virtual watch models in real time. By leveraging machine learning algorithms, the system can recognize and interpret these markers, enhancing the precision and stability of the AR experience.

Vuforia, a leading AR development platform, is employed to develop the system prototype. Vuforia provides the necessary tools and APIs for marker detection and tracking, simplifying the implementation process [16]. By utilizing Vuforia's features, the system can overlay the virtual watch models onto the user's real environment, allowing them to see how the watches would look and fit on their wrists. Lastly, the interface for the system is developed. The interface includes elements such as buttons, menus, and gesture-based controls, enabling users to navigate through the AR Try-On experience seamlessly. The interface design focuses on simplicity, enabling users to easily select and try different watch models, customize features, and view details.

3.2. Object Detection Framework

The model that was used was SSD MobileNet. This is because the main focus of this project is to be able to detect and visually wear a 3D object to the user's hand on a mobile [17]. MobileNet is known for its efficiency and small model size, making it suitable for deployment on mobile devices with limited computational resources. The combination of SSD and MobileNet allows for real-time object detection on mobile phones, making it suitable for applications such as augmented reality, image recognition, and video analysis. This framework shows the overview of the Vuforia object tracking framework to get a better understanding of how the Vuforia model target generator works.

3.3. System Testing and Evaluation

System testing and evaluation play a crucial role in ensuring the functionality, usability, and overall effectiveness of the AR Watch Try-On system. The testing phase involves conducting rigorous tests to validate the prototype's performance and identify any potential errors or issues that may impact user experience. Functional testing is carried out to verify that all features and functionalities of the AR Watch Try-On system are working as intended [18]. This includes testing the accuracy of marker detection and tracking, ensuring the virtual watch models align correctly with the user's wrist, and validating interactive elements such as buttons and menus [19]. This involves assessing the system's ease of use, intuitiveness, and overall user experience. Test participants are selected to represent the target user group, and they are asked to perform various tasks within the system. Their interactions, feedback, and observations are carefully documented and analyzed to identify any usability issues or areas for improvement. This feedback is invaluable for refining the system's interface, controls, and overall user interaction [20], [21].

During the testing phase, errors and bugs are identified and documented. These errors can range from minor issues, such as visual glitches or incorrect positioning of the virtual watch models, to more critical issues that may impact the system's stability or functionality. The identified errors are prioritized, and the development team works to rectify them through debugging and code optimization. Evaluation of the AR Watch Try-On system also involves assessing the system's performance against predetermined criteria and benchmarks. This may include measuring factors such as system response time, marker detection accuracy, and the smoothness of virtual object rendering [22]. Through quantitative analysis and comparison with predefined performance metrics, the system's overall effectiveness and efficiency can be evaluated [23].

4. Results and Discussion

4.1. Object Detection

The dataset used in this study is a collection of 4800 images with annotations. The dataset consists of 1 class, which is "hand." The collected samples were randomly split into training and test sets with a ratio of 80% and 20%, respectively. We employ commonly used evaluation metrics such as precision, recall, and mAP to evaluate the model performance in the application. We chose two different types of SSD MobileNet models, which are SSD MobileNet V2 FPNLite 320x320 and SSD MobileNet V1 FPN 640x640. This subtopic explains the results of each model together with the discussion on which model is the best.

As the training process progresses, the expectation is that total loss (errors) gets reduced to its possible minimum (about a value of 1 or lower). Based on figure 1, it should be possible to get an idea of when the training process is complete where the total loss does not decrease with further iterations/steps. Training for 50k steps took about seven hours 42 minutes and stopped at a total Loss (errors) value of 0.6324 from 1.223.

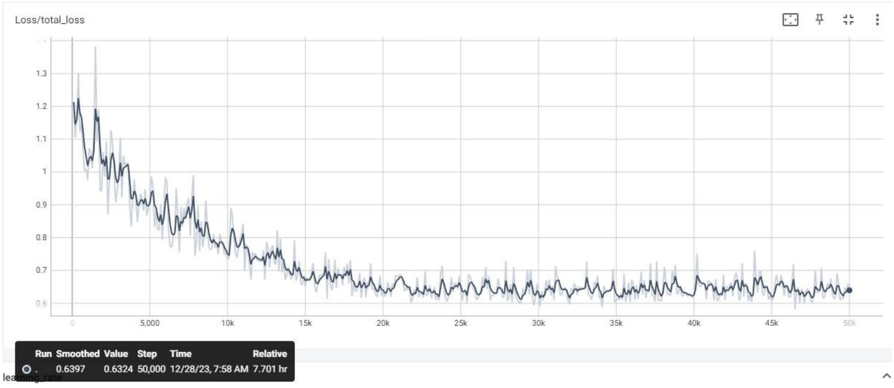

Figure 1. Total Loss SSD MobileNet V1

Based on figure 2, the value of 0.774 at the beginning indicates the initial regularization penalty applied to the model. As the training progresses, this penalty decreases to 0.5405 at the specified 25,000 steps. This reduction suggests that the model has found a balance between fitting the training data and avoiding overfitting, resulting in a lower regularization penalty.

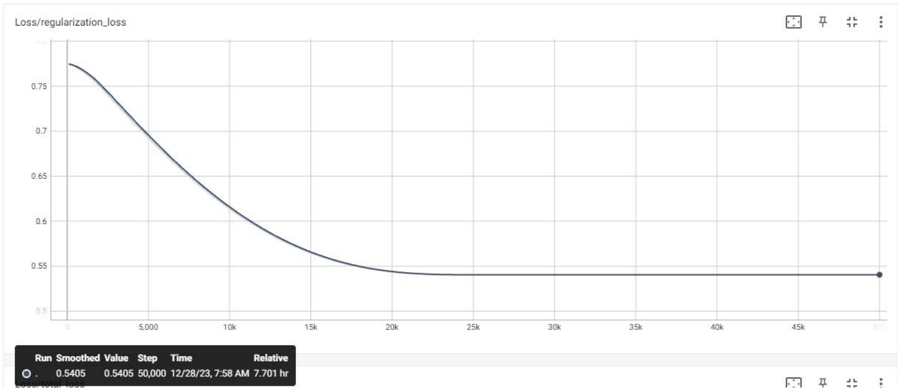

Figure 2. Regularization Loss SSD MobileNet V1

The line graph in figure 3 shows that the mAP increases to around 0.97 over time. This means that the model is getting better at identifying objects over time. The black dot in the middle of the line graph shows the current map, which is 0.9725.

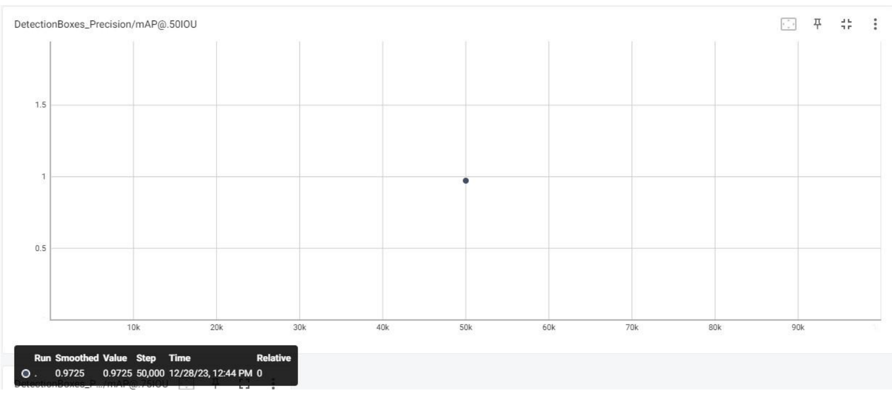

Figure 3. mAP Precision SSD MobileNet V1

Figure 4 illustrates the evaluation results of the SSD MobileNet V1 model on a specific dataset for object detection, focusing on identifying hands. The figure is organized into a 3x3 grid, where each section contains two images displayed side-by-side for comparison. Each detected hand is marked with a green bounding box, accompanied by a label "hand" and a confidence score, indicating the model's certainty in its detection. The confidence scores range from 85% to 100%, depending on factors such as clarity, position, and visibility of the hands in the images.

The model successfully detects hands across various conditions, including different angles, orientations (horizontal, vertical), and even partially occluded hands. Despite minor variations in confidence scores, the majority of detections achieve close to 100% accuracy, highlighting the robustness and reliability of the SSD MobileNet V1 model. This evaluation demonstrates the model's strong performance in accurately identifying hands, regardless of variations in lighting, positioning, or perspective within the images.

Figure 4. Evaluation batch SSD MobileNet V1

Some of the images that were tested on our SSD MobileNet v1. We can conclude that our model predicted the objects in images accurately.

4.2. SSD MobileNet v2

As the training process progresses, the expectation is that total loss (errors) gets reduced to its possible minimum (about a value of 1 or lower). Based on figure 5, it should be possible to get an idea of when the training process is complete where the total loss does not decrease with further iterations/steps. Training for 50k steps took about one hour 33 minutes and stopped at a total Loss (errors) value of 0.2041 from 0.7734.

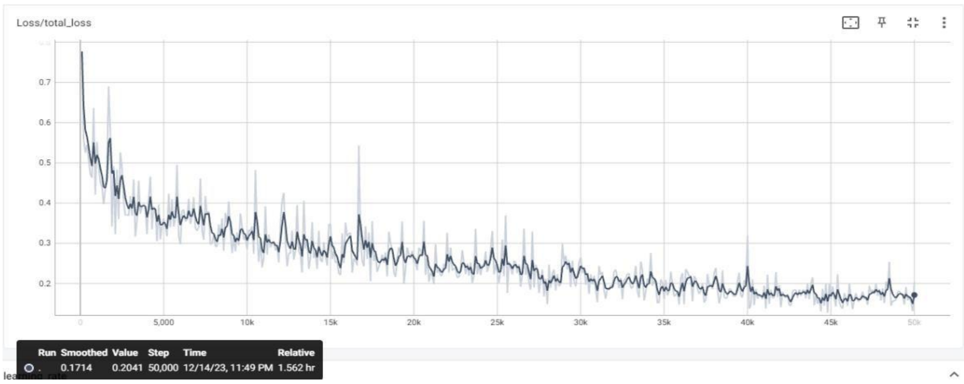

Figure 5. Total Loss SSD MobileNet V2

The value of 0.1538 (see figure 6) at the beginning indicates the initial regularization penalty applied to the model. As the training progresses, this penalty decreases to 0.0691 at the specified 50,000 steps. This reduction suggests that the model has found a balance between fitting the training data and avoiding overfitting, resulting in a lower regularization penalty.

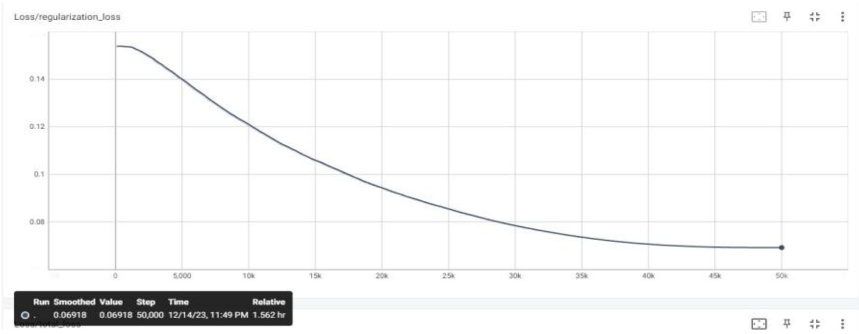

Figure 6. Regularization Loss SSD MobileNet V2

The line graph in figure 7 shows that the mAP increases to around 0.96 over time. This means that the model is getting better at identifying objects over time. The black dot in the middle of the line graph shows the current mAP, which is 0.9688.

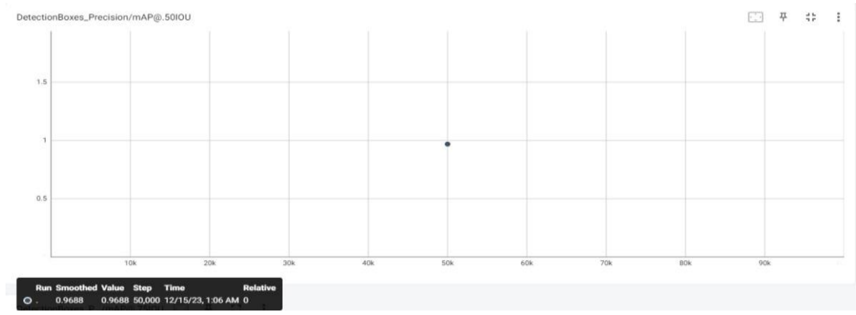

Figure 7. mAP Precision SSD MobileNet V2

Some of the images that were tested on our SSD MobileNet v2. We can conclude that our model predicted the objects in images accurately. However, the last two pictures show an overlapping detection on the detected object (see figure 8).

Figure 8. Evaluation batch SSD MobileNet V1

4.3. Augmented Reality

This topic will discuss the 3D watch models that were designed in unity, the Vuforia model target, and the experiment that was done to evaluate our model target. The 3D watch models were designed and built using Unity (see figure 9) to be implemented with the Vuforia deep learning model target. There are three types of watch styles, which are field watch style, digital watch style, and chronograph watch style. These watch models will be implemented in our Deep learning incorporated with the AR application Watch Try-On.

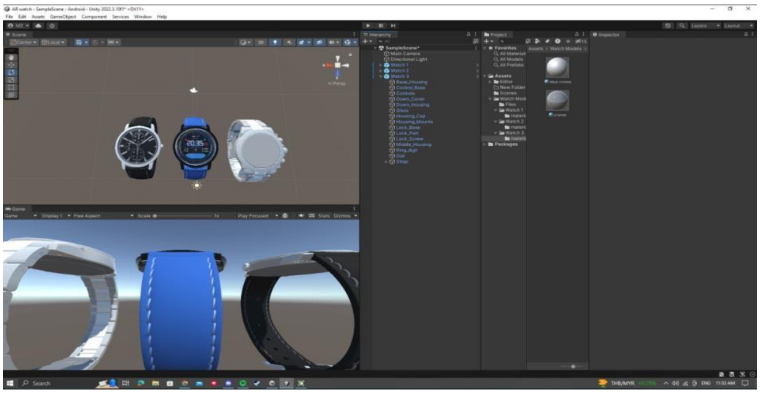

Figure 9. 3D watch model in Unity Hub

4.4. Vuforia Model Target

This subtopic discussed the Vuforia model target and the training result of our trained model target. There were two 3D hand models that were trained using the Vuforia model target generator. Both 3D hand models were successfully trained and showed promising results (see figure 10 and figure 11). The finished trained model target will be exported into Unity to be integrated with the 3D watch models.

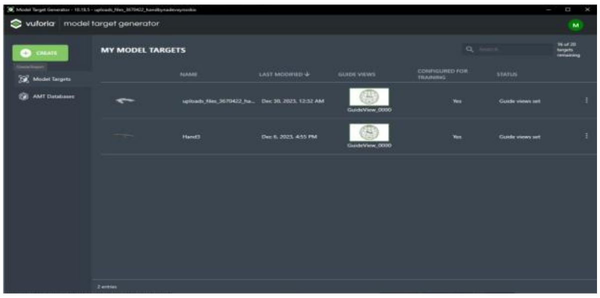

Figure 10. Vuforia model target generator UI

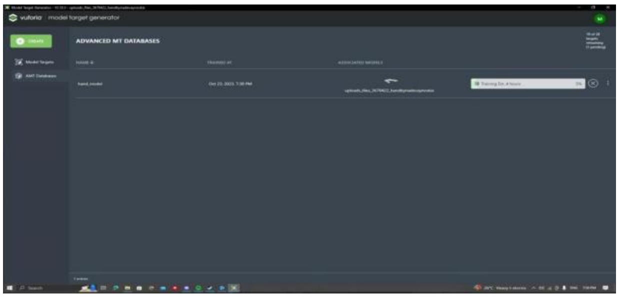

Figure 11. Vuforia model target generator (training)

Detecting a hand in real-time is a good way to test the accuracy and precision of AR-based object detection. A mobile device with three different luminous intensities of 25 lux, 150 lux, and 350 lux. In addition, we included various distances between the mobile camera and the user's hand in our evaluation to understand how good our scanning process was when collecting data points from the user's hand. This evaluation enables us to determine the optimal room lighting configuration for good AR-based object detection. We also included a 3d watch model in the test AR scenario (figure 12 and figure 13) during the evaluation to indicate whether the application keeps detecting and tracking the user's hand.

Figure 12. The screenshot of the watch try-on application successfully detects and renders the 3D watch model to the wearer’s hand.

Figure 13. AR watch try-on application training scenario.

Table 1 presents the results of the AR Watch Try-On Experiment, which evaluates the detection status of the virtual watch under varying luminous intensities, distances, and focus conditions. The experiment examines how light levels (25, 50, and 150), distances (10 cm and 20 cm), and camera focus (Yes/No) affect the application's ability to detect the wearer’s hand and render the virtual watch.

Table 1. AR Watch Try-On Experiment

AR Watch Try-On Experiment			
Luminous intensities	**Distance (cm)**	**Focus**	**Detection Status**
25	10	No	No
25	20	No	No
25	10	Yes	Yes
25	20	Yes	No
50	10	No	Yes
50	20	No	No
50	10	Yes	Yes
50	20	Yes	No
150	10	No	Yes
150	20	No	No
150	10	Yes	Yes
150	20	Yes	Yes

The results indicate that detection is more successful when the camera focus is enabled, especially at shorter distances (10 cm) across all luminous intensities. For example, at 25 and 50 luminous intensities, detection succeeds when focus is “Yes” and the distance is 10 cm. However, detection fails at 20 cm under similar conditions. At higher luminous intensity (150), detection improves further, showing success even without focus at a 10 cm distance, but detection remains inconsistent at 20 cm unless focus is enabled.

Overall, the table highlights that camera focus and shorter distances play critical roles in ensuring successful detection of the hand, while higher luminous intensity can enhance performance, particularly at close range. Detection becomes unreliable at longer distances (20 cm), especially when focus is not applied. This suggests that the AR application performs optimally under well-lit conditions, with close range and proper focus being key factors for accurate detection.

4.5. Watch Try-On Applications

The watch Try-On application focuses on these three features, which are try-on, color options, and price information. Implement a try-on feature that superimposes the selected watch onto a user's wrist using AR. This feature enhances the user experience by helping them visualize how the watch will look in different situations. Figure 14 illustrates the try-on feature.

Figure 14. Watch the try-on application interface.

Next are color options and price information illustrates in figure 15. Display the available color options for each watch model. Allow users to switch between colors with a simple tap or click. For the price information. The display price for each watch model and color variant. Transparency about pricing is crucial for users making purchasing decisions.

Figure 15. Watch the try-on application color and price interface.

Based on the performance of the training set of both SSD MobileNet's models, where each ran through 50k iterations with the same number of images for training data, there were differences, especially in the gap of the total loss, regularization loss, and mAP values. Among both the models, SSD MobileNet's V1 and SSD MobileNet's V2, SSD MobileNet's V1 had the highest result in mAP of 0.9725 and a total loss of 0.5405 from 0.774. In addition to that, the training time for this model did take quite a long time, which was 7 hours and 42 when compared to MobileNet's V2, which was 1 hour and 33 minutes with the result in mAP of 0.9688 and a total loss of 0.0691 from 0. 1538.

Consequently, even though SSD MobileNet's V1 was better than SSD MobileNet's V2 in the testing results, we implemented SSD MobileNet's V2 in our webcam due to its faster inference. SSD MobileNet's V1 was lagging and had a slower inference when tested in real-time. For the Watch Try-On application, we choose the Vuforia model target because of its ability to integrate with the 3D watch models' AR technologies. Neither SSD Mobile Net's models have the ability to be incorporated with AR technology for our application.

Finally, we tested the webcam and phone camera with SSD MobileNet's V2 model, and for the Watch Try-On application, the Vuforia model Target was chosen. We were satisfied with the results as all of them were able to detect the user's hand accurately with high confidence, and for the application, the 3D watch manages to visually be worn to the user's hand in real-time. To summarize, we accomplished this project by choosing the best model, and it produced the best results.

ISSN 2723-6471

5. Conclusion

The goal of this project is to identify the user’s hand and visually wear a 3D watch model to the user’s hand by integrating an object detection model with AR technologies. To do this, two versions of SSD MobileNet were used to identify the user’s hand and then detected it using bounding boxes. In addition, the scope of this research is limited to the application of the algorithm to the detection of the user’s hand in real time with different luminosities and ranges between the detected object and the camera. To carry out the task of processing the data and training the models, we utilized Python, Tensorflow, Jupyter, Vuforia model target generator, and Unity.

This project has its strengths and limitations based on the model that was implemented in this project, which was discussed in earlier chapters. Furthermore, the strengths and limitations of this project will be explained and discussed in detail in this topic. The strengths and limitations of the models and techniques used in this research study suggest several recommendations for future work. Firstly, integrating SSD MobileNet models with AR technologies could create a hybrid model for improved tracking. Additionally, increasing the frame rate would enhance real-time object detection capabilities. Furthermore, the application could be expanded to visually simulate wearing various 3D hand accessories, such as bracelets, bangles, and rings. Finally, by adding detection and tracking for the human body, the application could also allow users to virtually try on 3D shirts and dresses.

6. Declarations

6.1. Author Contributions

Conceptualization: A.A., T.B.K., D.A.D., M.Z.Z., and P.A.A.H.; Methodology: T.B.K.; Software: A.A.; Validation: A.A., T.B.K., and P.A.A.H.; Formal Analysis: A.A., T.B.K., and P.A.A.H.; Investigation: A.A.; Resources: T.B.K.; Data Curation: T.B.K.; Writing – Original Draft Preparation: A.A., T.B.K., and P.A.A.H.; Writing – Review and Editing: T.B.K., A.A., and P.A.A.H.; Visualization: A.A. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Z. Zhang, D. Weng, H. Jiang, Y. Liu, and Y. Wang, "Inverse Augmented Reality: A Virtual Agent’s Perspective," in *IEEE Xplore*, vol. 2018, no. 1, pp. 1–5, 2018.

[2] M. Grampp, D. Rohr, M. Buti, and K. Egerth, *The Deloitte Swiss Watch Industry Study 2020*, Deloitte, vol. 2020, no. 1, pp. 1–30, 2020.

[3] T. Hariguna and A. Ruangkanjanases, "Assessing the impact of artificial intelligence on customer performance: A quantitative study using partial least squares methodology," *Data Sci. Manag.*, vol. 7, no. 3, pp. 155–163, Sep. 2024.

[4] M. Lorenz, S. Knopp, and P. Klimant, "Industrial Augmented Reality: Requirements for an Augmented Reality Maintenance Worker Support System," in *IEEE Xplore*, vol. 2018, no. 1, pp. 1–6, 2018.

[5] D. N. Nasser, "Augmented Reality in Education Learning and Training," in *International Conference on Control and Automation*, vol. 2018, no. 1, pp. 1–7, 2018.

[6] T. S. Barbosa and A. H. Kronbauer, "Panorama of Researches Related to the Application of Virtual Reality in the Health Area in SVR," in *IEEE Xplore*, vol. 2019, no. 1, pp. 1–5, 2019.

[7] S. Chen, B. Hu, Y. Gao, Z. Liao, J. Li, and A. Hao, "Lower Limb Balance Rehabilitation of Post-stroke Patients Using an Evaluating and Training Combined Augmented Reality System," in *IEEE Xplore*, vol. 2020, no. 11, pp. 1–6, Nov. 2020.

[8] G. K. Upadhyay, D. Aggarwal, A. Bansal, and G. Bhola, "Augmented Reality and Machine Learning based Product Identification in Retail using Vuforia and MobileNets," in *IEEE Xplore*, vol. 2020, no. 2, pp. 1–6, Feb. 2020.

[9] G. Wu, M. Qiu, Y. Zhang, and Y. Zheng, "Research and Development of Augmented Reality Children's Puzzle Game Based on Vuforia," in *IEEE Xplore*, vol. 2019, no. 8, pp. 1–5, Aug. 2019.

[10] J. S. I. Rieder, D. H. van Tol, and D. Aschenbrenner, "Effective close-range accuracy comparison of Microsoft HoloLens Generation one and two using Vuforia ImageTargets," in *IEEE Xplore*, vol. 2021, no. 3, pp. 1–6, Mar. 2021.

[11] Z. Zeng, Y. Yao, Z. Liu, and M. Sun, "A deep-learning system is bridging molecule structure and biomedical text with comprehension comparable to human professionals," *Nature Communications*, vol. 13, no. 1, pp. 1–10, 2022.

[12] M. Bilal et al., "Auto-Prep: Efficient and Automated Data Preprocessing Pipeline," *IEEE Access*, vol. 10, no. 1, pp. 107764–107784, 2022.

[13] J. White, S. Hays, F. Q., S.-S. J., and S. D. C., "ChatGPT Prompt Patterns for Improving Code Quality, Refactoring, Requirements Elicitation, and Software Design," in *Generative AI for Effective Software Development*, P. Nguyen-Duc, F. K. Anh, and A. Abrahamsson, Eds., Springer Nature Switzerland, vol. 2024, no. 1, pp. 71–108, 2024.

[14] M. K. Alqudah, R. Razali, M. K. Alqudah, M. N. Al Dalaien, H. M. Alabool, and H. A. Alkhazaleh, "A grounded theory of selecting lean and agile practices for software development," Journal of Software: Evolution and Process, vol. 36, no. 4, p. e2539, 2024.

[15] R. Razali, M. K. Alqudah, and D. A. P. Zainal, "The discovery of grounded theory practices for software engineering research," Electronic Journal of Business Research Methods, vol. 18, no. 2, pp. 153–165, 2020.

[16] P. T. Ho, J. A. Albajez, J. Santolaria, and J. A. Yagüe-Fabra, "Study of Augmented Reality Based Manufacturing for Further Integration of Quality Control 4.0: A Systematic Literature Review," *Applied Sciences*, vol. 12, no. 4, pp. 1–18, 2022.

[17] R. bin Islam, S. Akhter, F. Iqbal, Md. S. U. Rahman, and R. Khan, "Deep learning based object detection and surrounding environment description for visually impaired people," *Heliyon*, vol. 9, no. 6, pp. e16924, 2023.

[18] Y. Liu et al., "Comparing VR- and AR-Based Try-On Systems Using Personalized Avatars," *Electronics*, vol. 9, no. 11, pp. 1–15, 2020.

[19] K. M. Jeon, C. Chun, H. Kim, and M. J. Lee, "User-Aware Audio Marker Using Low Frequency Ultrasonic Object Detection and Communication for Augmented Reality," *Applied Sciences*, vol. 9, no. 10, pp. 1-9, 2019.

[20] R. Liu and Y. Wang, "Enhancing Enterprise Value Creation Through Intelligent Digital Transformation of the Value Chain: A Deep Learning and Edge Computing Approach," *Journal of the Knowledge Economy*, vol. 2024, no. 1, pp. 1–19, 2024.

[21] M. Karimi, Z. Zakariyaeinejad, A. Sadeghi-Niaraki, and A. Hosseininaveh, "A new method for automatic and accurate coded target recognition in oblique images to improve augmented reality precision," *Transactions in GIS*, vol. 26, no. 1, pp. 1509–1530, 2022.

[22] F. J. Romero Ramírez, R. Muñoz-Salinas, and R. Carnicer, "Speeded up detection of squared fiducial markers," *Image Vis. Comput.*, vol. 76, pp. 38–47, 2018.

[23] D. Z. Rodríguez, O. D. Okey, S. S. Maidin, E. U. Udo, and J. H. Kleinschmidt, "Attentive transformer deep learning algorithm for intrusion detection on IoT systems using automatic explainable feature selection," *PLoS One*, vol. 18, no. 10, pp. 1-12, 2023.