

Original Article

Virtual Gold Chain Try-on using Augmented Reality with Neck Occlusion Detector

Maitreyee Londhe¹, Anjali Naik²

^{1,2}Department of Computer Engineering, MKSSS's Cummins College of Engineering for Women, Maharashtra, India.

²Corresponding Author : anjali.naik@cumminscollege.in

Received: 09 March 2026

Revised: 25 July 2026

Accepted: 06 August 2026

Published: 30 September 2026

Abstract - The inability to try-on jewellery continues to limit customer trust in online jewellery shopping. This paper presents a virtual gold chain try-on system using Augmented Reality in real-time. The primary objective is achieving precise neck-based alignment and improving user confidence and interaction. The proposed research work integrates MediaPipe FaceMesh for landmark detection, neck occlusion detector for detecting obstructions caused by hair, clothing, hands and OpenCV for chain overlay, deployed through Streamlit interface. The occlusion model employs CNN classifier MobileNetV2 for feature extraction from the custom dataset which identifies whether the neck is visible or occluded. Evaluation on the 1,200-image dataset resulted in an overall classification accuracy of 92.5%. Experimental evaluation showed that the system maintained a frame rate of 30 FPS with an end-to-end latency of 150 ms, thereby supporting smooth real-time user interaction. This highlights the potential of enhanced user experience, reduces the return rates and can serve as a foundation for other AR applications in retail.

Keywords - Augmented reality, Landmark detections, Online jewellery, OpenCV, Virtual try-ons.

1. Introduction

Online shopping has gained momentum after the Covid-19. People prefer it over traditional shopping because of its convenience. One of the main aspects of shopping is trying out the product; whether it is beauty, footwear, clothing, accessories, etc. This gap is bridged in online shopping through the integration of Augmented/Virtual Reality in these systems. Out of all the e-commerce sectors, the one that still struggles are online jewellery shopping. People still hesitate to buy jewellery online one of its biggest reasons being that they aren't able to physically try the products on. This uncertainty often reduced confidence of the customer and contribute to product returns. Hence, to solve this problem, this study builds a virtual gold chain try-on system. Users can try available chain sizes virtually through a webcam. This system was built for the users, so that they are confident enough to buy the jewellery and are satisfied with their purchase.

Recent advancements in Augmented Reality, deep learning and computer vision have enabled virtual try-on applications across multiple domains, including clothing, footwear, eyewear, wristwatches and jewellery. These are employed by technologies such as MediaPipe, OpenCV, markerless tracking, three-dimensional rendering. They have significantly improved real-time visualization and user interaction. Several challenges remain unsolved despite these developments. Most jewellery - oriented virtual try-on systems concentrate on facial accessories, rings, or bracelets

whereas comparatively limited attention has been given to accurate real-time gold chain placement. In addition, previous studies have assumed an unobstructed neck region and rarely incorporate dedicated mechanisms for handling occlusions caused by hair, clothing, scarves or hand movements. These assumptions frequently affect overlay accuracy and realism. These limitations highlight the need for a robust neck-aware virtual jewellery try-on framework.

Motivated by these limitations, the present work proposed a real-time virtual gold chain try-on system which has integrated MediaPipe FaceMesh with a MobileNetV2- based neck occlusion detector to improve chain alignment. This framework incorporated neck visibility analysis before performing chain overlay, thereby improving placement robustness, unlike the conventional virtual jewellery systems that solely rely on facial landmark detection. The proposed system supports multiple chain sizes, performs accurate neck landmark-based alignment and provides real-time visualization through Streamlit interface. The system is evaluated using both quantitative and qualitative metrics.

2. Literature Review

Raval et al. [1] developed a 3D markerless AR-based mobile application which was an interior visualization tool that allowed users to preview furniture placement in real environments. This application employed Google's ARCore SDK to detect surfaces, estimate lighting and support



interactive functionalities such as scaling and color customization. This study highlighted AR's potential to enhance customer experience and innovate for digital retail markets.

Ghatge et al. [2] presented an approach that ensured faster processing, and a reliable user experience. Jewellery such as rings, bracelets, necklaces and earrings are placed by detecting body parts using real-time pose detection from MediaPipe and Google ML kit. This mobile-based application had a stability rate > 90% and a response time < 200ms and was built using Flutter, with a built-in camera. However, lighting sensitivity, lack of 3D realism and camera quality were subjective to the quality of the phone camera.

Rajan et al. [3] developed a usable system for clients looking for an interactive shopping experience. The paper introduced the system, which detected a face that was captured by a webcam and overlaid jewellery on the chin region using computer vision and image processing methods. The face_recognition library was used to detect the face. Image and video processing were demonstrated using libraries such as OpenCV, PIL.

Basumatary et al. [4] introduced BodoAR, a marker-based mobile AR application to teach Bodo alphabets and numbers to primary students from this endangered language. They used 3D visuals, pronunciation audio and interactive quizzes and carried out this study to compare the outcomes between AR and textbook instruction among native and non-native learners. Empirical findings revealed a significant improvement in the academic achievement of native speakers using AR while young learners struggled with phone handling. This study demonstrated that AR can serve as an effective tool for language preservation and early education.

Prajapat et al. [5] research presented a hand jewellery try-on system. The developed application achieved 95% accuracy using OpenCV, MediaPipe and AR foundation. A total of 21 hand landmarks were detected and then encoded using the MediaPipe encoding algorithms. A 3D model of the jewellery was created and then jewellery was augmented. Their future scope included facial jewellery such as mang tikka, earrings, necklaces and nose pins.

Yadav et al. [6] presented a comprehensive review of augmented reality, which discussed its history and investigated the technical components that enabled realistic overlays. This included displays, vision-based tracking and sensor-based recognition. They evaluated major AR SDK's such as Vuforia, Metaio, Wikitude and ARToolKit and outlined their features and licensing differences. The study emphasized increasing use cases in healthcare, education and marketing while also facing limitations due to environmental conditions, hardware dependency and complex user interaction.

Mushtaq et al. [7] presented a machine learning framework because of the increase in security threats in AR/VR systems caused by real-time data exchange. This was presented to counter spoofing attacks using anomaly detection methods. A comparative study between SVM, KNN and tree algorithms was conducted and the linear SVM offered the most effective accuracy, processing speed and minimal computational cost. This demonstrated that ML was a reliable solution for privacy in AR/VR environments.

Hongxia Chu [8] introduced a VR based jewellery design framework to transform 3D models into immersive displays. It used CNC plate making and optimized 3D point cloud algorithm to build precise digital models. The system streamlined collaboration between designers, clients, and manufacturers and achieved a product loss rate as low as 1.014%. It exhibited superior performance over traditional 3D printing and depth camera methods. Although this approach enhanced efficiency and design quality, it lacked accuracy.

Chandapatla Jaideep Sai et al. [9] proposed an integration of AR, VR and AI for virtual jewellery try-on. MyWebAR employed real world AR try-ons via a webcam, Three.js offered VR based 3D visualizations and csm.ai was used to generate 3D models from images. The use of csm.ai had resulted in cost reduction and saved time, while all together these technologies improved user engagement and satisfaction.

Lagharic et al. [10] analyzed immersive technology by review and comparison between AR and VR technologies. This mainly focused on their conceptual differences and device requirements. The technical components included stereoscopic displays and motion-tracking.

Applications across multiple industries and emphasized that while both enhance user experience, AR faces development and privacy challenges and VR risks social isolation. They concluded that AR was still in development, while VR was comparatively mature but both will contribute to future immersive multimedia systems.

Paul et al. [11] developed a real-time ring size estimation system using deep learning, AR and CV technologies. Finger dimensions were calculated using Euclidean distance, and converted into a standard ring size. The selected virtual ring was then rendered and aligned with the detected finger using OpenCV and augmented reality systems for real-time visualization. This approach focused on rings and the use of 2D .PNG images.

Tens et al. [12] introduced a augmented reality-based virtual clothing try-on platform for kids. The system was designed using a model trained using a dataset of ML algorithms for garment fitting, pose estimation and texture adaptation. The integration of CNN and GANs enhanced the

realism of fabric draping and movement. This system also integrated AR for real-time simulation and supported 3D body scan for precise sizing.

S. Patil et al. [13] introduced a mobile-based AR application for wristwatches built with Unity 3D, Blender and Vuforia SDK. This system supported marker-based AR where the user can scan the marker and see a watch virtually aligned with the marker on the wrist. The marker detection time was less than 1s and the model placement yielded an accuracy of 90%. Based on the surveys conducted, the overall user satisfaction rating was 4.6 out of 5.

Ting Chou [14] developed a virtual footwear try-on system based on AR. Among these, YOLO - 6D achieved the best accuracy and speed, with a performance of 53 fps. The limitations include performance under dynamic movements as it assumed rigid body movement.

Nair et al. [15] introduced a system in which a user can virtually try on a hairstyle through a live video or image dynamically. The system used OpenCV and deep learning for face and hair detection. ARCore /ARKit was used for real-time hair mapping. Dynamic hairstyle simulation was enabled using machine learning algorithms. The mapping accuracy was 92% and user satisfaction was 85%.

Mahek Agarwal et al. [16] research presented a Virtual Trial Room, image-based VTO system. OpenCV and OpenPose were used for pose detection. CNNs trained on the COCO dataset extracted keypoints and JPPNet, which was trained on the LIP dataset, was used for human parsing. Cloth Warping was achieved through STN and TPS deformation. Users reported it to be visually realistic and provided accurate representation of fit and style. However, this study struggled with image resolution and complex clothing detail production.

Prakriti Thapa et al. [17] proposed an augmented reality framework for virtual eyewear try-on and face shape-based recommendations. The system uses MTCNN for face detection, VGG-16 for face analysis and Three.js for real-time 3D rendering.

It achieved 92% accuracy with 85 % IoU alignment with low width error of ~5% in AR overlay. It has the advantage of

immersive interaction and personalized recommendations. However, the areas of improvement highlight the misclassification of face shapes and dataset limitations.

Deepa Sunny et al. [18] introduced a markerless AR foot detection method for virtual shoe try-on system. It has made use of LiDAR sensing and a seven - foot dimension model with graph-based algorithms. Testing with 30 participants showed higher recall and overall F-measure than marker-based approaches and ensured the accuracy of the proposed method. Its area of challenge remains precision and complex foot dimensions owing to the sensor resolution. This was a scalable solution for digital healthcare and e-commerce applications.

Charlee Kaewrat et al. [19] designed a framework for virtual clothing try-on. This was built using YOLOv5, U-Net and pix2pix GAN. It was trained on diverse datasets such as DeepFashion, FashionAI, COCO, MPII Human Pose, FGV6 and Chictopia to ensure robustness. It also had different modules such as outfit generation, history module and daily outfit recommendation modules, which helped users to identify and elevate their own style. However, it declined in accuracy for complex clothing details and overlapping accessories and could not maintain realism under varied poses.

Hoffmann and Mai [20] conducted a review of augmented shopping reality through 56 empirical studies analyzing how consumers adopt AR based shopping functions. This study mapped consumer responses and proposed a framework for adoption and behaviour. The results confirmed its ability to enhance consumer confidence and reduce return rates; however, limitations include fragmented theory, small samples and narrow product category focus. Although, it helped integrate online and in-store shopping experiences.

Fadilah Haya Kamilah et al. [21] presented a comprehensive review of research on Virtual Fitting Rooms in fashion. The study analyzed 20 journals published in the last five years identifying that they consistently link VFRs with higher purchase intention, reduced uncertainty and better customer engagement. The study was limited in geographic reach, with limited exploration of cultural and emotional dimensions. But still the study highlighted the potential of VFR to revolutionize e-commerce.

Table 1. Tabular Analysis across all domains for existing AR systems

Domain	No. of Papers	Models used/ Key tools	Dataset/ Sources
Furniture AR Visualization	1	ARCore, Surface Detection, Lighting Estimation	Real-world indoor scenes
Jewellery Try-On & Design	6	MediaPipe, OpenCV, ML Kit, AR Foundation, VR design tools, 3D reconstruction	Custom jewellery PNGs, 3D models, webcam captures
Bodo / Education AR	1	Marker-based AR, 3D models, audio integration	Custom Bodo alphabet dataset

Generic AR/VR Systems	2	AR SDKs (Vuforia, Metaio, ARToolKit), Sensor-based tracking, VR interactions	Technical comparisons, conceptual analysis
Security in AR/VR	1	SVM, KNN, Decision Trees, anomaly detection	AR/VR security datasets, custom experimental data
Clothing / Fashion Try-On	3	CNN, GANs, OpenPose, JPPNet, STN/TPS warping	DeepFashion, COCO, LIP, FashionAI, MPII
Accessories	5	MTCNN, VGG-16, Unity, Vuforia, YOLO-6D, ARCore/ARKit, LiDAR	Eyewear datasets, Foot datasets, hairstyle data, 3D models
AR/VR Review	2	Literature analysis, consumer behavior frameworks	56 empirical studies, 20 VFR journal papers

Table 1 summarizes the domains in which AR/VR has existing systems along with their tools and datasets/sources. In this study, 21 papers published between 2021 to 2025 were reviewed.

Augmented Reality and Computer vision techniques are being adopted and some studies have used OpenCV (six papers) and MediaPipe (four papers) for real-time virtual try-on systems.

Eight studies used deep learning models such as CNN, YOLO and GAN-based frameworks especially for clothing and footwear as they require complex pose estimation. Of the total reviewed work, three papers are based on fashion clothing, six on jewellery, five on accessories and two were generic e-commerce/ user study papers.

The reviewed literature indicates that Augmented Reality, computer vision and deep learning technologies have emerged as key technologies for enhancing virtual shopping experiences across multiple application domains. Mostly existing jewellery applications concentrate on hand or facial accessories while dedicated solutions are comparatively limited for real-time neck-oriented jewellery. These findings motivate the proposed research presented in the next section.

3. Research Gaps

The literature review indicated that several virtual jewellery try-on systems have been proposed using Augmented Reality and computer vision techniques. Despite these advancements, there are differences in their implementation and scope. This leaves opportunities for further improvement.

Table 2. Comparison of existing jewellery try-on systems with the proposed framework

Paper	Key Contribution	Limitations	Novelty
Rajan et al. [3]	Face jewellery overlay on chin through webcam	Only chin landmark used	Inclusion of left and right points of neck for accurate chain placement
Ghatge et al. [2]	Jewellery app try-on using pose detection	Depends on phone camera quality, lacks 3D realism, lighting sensitivity	Webcam-based system with Streamlit; lightweight desktop deployment
Prajapat et al. [5]	Hand jewellery try-on with AR foundation and 3D model overlay	Focus only on hands	Targeted on neck-based jewellery

The above Table 2 compares different studies of the existing papers on virtual jewellery try-on where their key contributions and limitations are discussed. However, there remains a significant research gap in neck-oriented jewellery especially gold chains, using webcams in real-time with neck occlusion detectors.

To bridge this gap, the proposed system uses MediaPipe FaceMesh for landmark detection and precise neck alignment of gold chains in three different sizes with a Streamlit-based deployment for an interactive try-on experience along with an occlusion detector.

4. Materials and Methods

The proposed system enabled real-time virtual gold chain try-on using Augmented Reality with a neck occlusion detector. The user can select a chain size from the three available chain sizes (16, 18 and 22) and check whether the neck was occluded or visible along with the chain overlay. Two major components are integrated into a single pipeline:

(i) Neck Occlusion detection using a MobileNetV2 CNN classifier

(ii) Chain overlay using MediaPipe FaceMesh

4.1. Environment Setup

The system was developed on Google Colab using the following libraries:

- Streamlit: Provided the user with a UI for selecting the chain size and also rendering the live output.
- streamlit-webrtc: Handles webcam real-time video streaming, it was an extension of Streamlit
- MediaPipe FaceMesh: Detects 468 facial landmarks from the detected face for the chain overlay.
- Av: Converts the webcam frames to NumPy arrays.
- OpenCV: used mainly for image processing and frame handling
- NumPy: For numerical operations.
- Python Image Library (PIL): mainly used for chain overlay and transparency
- Pyngrok: Public URLs from colab can be shared with this.
- TensorFlow/ Keras: Used for building, training and evaluating the MobileNet-based deep learning model
- Scikit-learn: Used to computing class weights and handle dataset imbalance during model training.

The implementation and testing of the system were performed on a MacBook Air (M1, 2020) with an 8-core Apple M1 processor, 8 GB RAM and macOS sequoia 15.0.1. The system used only the integrated Apple M1 GPU for real-time MediaPipe and OpenCV processing.

The proposed processing pipeline is as follows:

4.2. Real-Time Webcam Acquisition

A live camera feed was provided from the Streamlit web interface, where the user can select a size from the available chain sizes. Frames from the laptop's webcam were captured at 640×480 resolution. Each frame was captured in the BGR format (OpenCV default) and then converted to RGB for MediaPipe and CNN input. These were forwarded simultaneously to the occlusion model and landmark detector.

4.3. Neck-Occlusion Detection

The visibility of the neck for the accurate placement of virtual jewellery in AR depends on a clear view of the user's neck. In real-world scenarios, the neck becomes partially hidden due to long hair, clothes and hand placement, which introduce noise in landmark detection and can make the chain placement look unrealistic. To address this, the system integrated a dedicated neck occlusion detector that determined whether the user's neck region was sufficiently visible for the AR overlay and distinguished between usable and unusable frames before the chain overlay was attempted.

4.4. Dataset Preparation

A custom image dataset of 800 labelled images was created. In these typical real-world conditions, occlusions by hair and clothing images were mainly included. The dataset was divided into two classes:

Class 0 - neck_occluded: Neck was partially or completely obstructed by hair, scarves, masks, turtlenecks and hands.

Class 1 - neck_visible: The neck region was completely visible without significant obstruction.

The dataset was used for classifier development with a training-validation split. To strengthen generalization and real-world variability, the dataset was augmented with transformations such as zoom, rotation, flip and brightness adjustments. For subsequent evaluation of the model, dataset was further augmented to obtain 1,200 images, consisting of 600 neck-visible and 600 neck-occluded images.

4.5. Preprocessing Pipeline

All images were standardized before training by resizing them to $224 \times 224 \times 3$ and were normalized to range [0,1], processed in batch using ImageDataGenerator and divided into 80 % training and 20% validation subsets so that all the frames maintained the same format required by MobileNetV2.

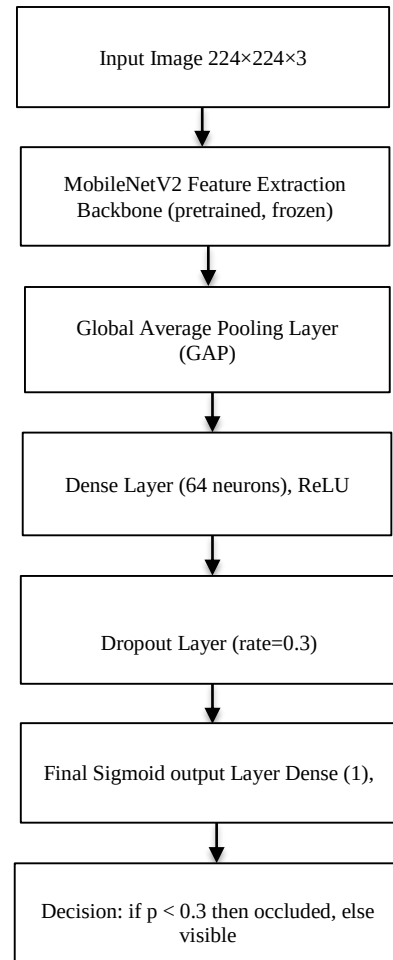


Fig. 1 Model architecture of MobileNetV2 model for neck occlusion detection

Figure 1 shows the pipeline of the MobileNetV2 model for training a neck occlusion classifier to identify whether a user's neck was visible or occluded.

MobileNetV2 is selected for this particular model as the backbone of this model because of its feature extraction capability and computational cost. Because the occlusion classifier must run on every incoming webcam frame, a lightweight architecture was essential.

MobileNetV2 pretrained was used on ImageNet served as the feature extraction backbone. Its convolutional layers were kept fixed throughout training, while only the newly introduced layers were optimized for the neck occlusion task. Training was performed for 15 epochs using a batch size of 16. This ensured a strong ability to generalize to new webcam frames, faster training and reduced risk of overfitting. These layers extracted high-level spatial features.

A Global Average Pooling (GAP) was used instead of flattening the convolutional layers to force the network to learn global patterns of occlusion and not the pixel-level noise, reduce parameters and remove spatial bias. Subsequently, a lightweight dense layer with 64 neurons was added, which captured fine-grained cues (shadow patterns from hair, overlapping natural skin tone vs hands). Representation remains sparse and efficient because of the ReLU activation.

A dropout layer with a rate of 0.3 was inserted to prevent overfitting, as the dataset was relatively small, where the augmented images may look similar. This ensured that the classifier generalized well to unseen necks and faces. The concluding sigmoid output layer produces a single probability p [0,1]. The thresholds were $p < 0.3$ for occluded (class 0) and $p \geq 0.3$ for visible (class 1) classes. This was chosen because the occluded images produced low scores. The final output was directly displayed on Streamlit-WebRTC, regardless of whether the chain was visible or occluded.

4.6. Face Landmark Detection and Chain Alignment

Facial landmark detection was performed using MediaPipe FaceMesh, through which 468 facial points were generated for each frame. The chin baseline was defined as landmark number 152 and landmarks 234 and 454 represented the left and right lateral neck reference points respectively and were used to overlay the chain exactly on the neck. Before the landmarks were finalized, a validation experiment was conducted in which the bottom center of the detected face was identified as the chin position. This process confirmed that these landmarks were appropriate for the chain overlay. The system looks for two landmarks, specifically for width - landmark numbers 234 and 454. Using this method, the horizontal center of the neck was determined. Landmark number 152 then helped in determining the vertical position of the chain, indicating the precise height at which the chain should be positioned along the neck. The chain dynamically resized with a width proportional to the frame width and height based on the scale ratio per chain size. For example:

$$chainwidth = framewidth \times scale \quad (1)$$

$$chainheight = chainwidth \times heightratio \quad (2)$$

Equations (1) and (2) allowed the dynamic resizing of chains for different sizes and different frame widths.

$$pos_x = x_{chin} - \frac{chinwidth}{2} \quad (3)$$

$$pos_y = y_{chin} - \frac{chinheight}{4} \quad (4)$$

Equations (3) and (4) show that the chain positions were then determined with landmark 152 as reference where the chains were horizontally centered at the chin and vertically it was one fourth of the chain height which ensured proper neck alignment. The preprocessed frame with the chain overlay was converted to a standard video format that the streaming library could handle and returned to the live video stream.

4.7. Integration of Occlusion detector and MediaPipe FaceMesh

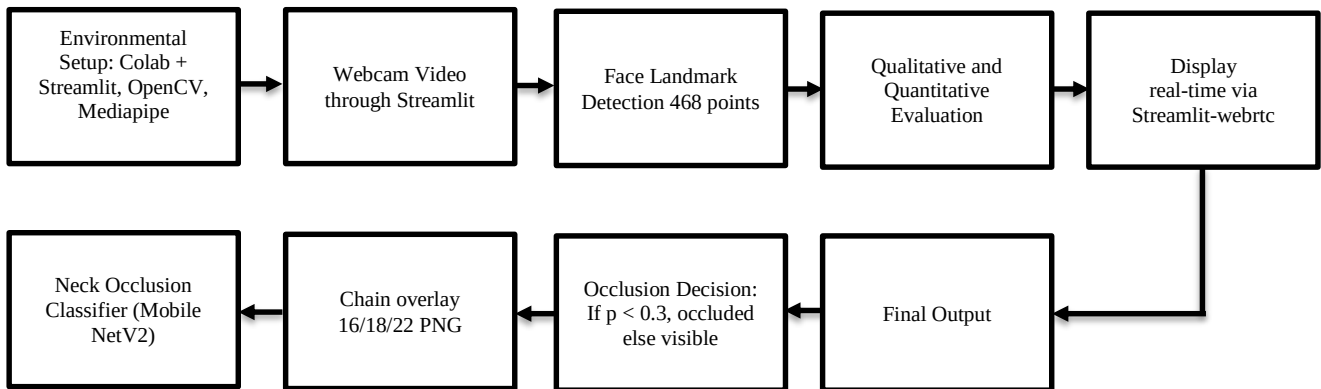


Fig. 2 Overall workflow of the proposed system

The above Figure 2 presents the processing pipeline of the proposed system. Through the URL, the Streamlit app was opened and chain size was selected. Once the webcam was started, the frames of the video were captured at 30 FPS. Each frame was captured in BGR (OpenCV) default format and then was converted to RGB format, as MediaPipe requires this format.

Using MediaPipe FaceMesh, 468 facial landmarks were detected and a neck occlusion detector was integrated into this pipeline, which predicts whether the neck region was visible or occluded depending on the occlusion threshold. Now, according to the user's selected size the chain image was loaded. The chain was then overlaid on the determined points. The frame was then returned to the live video stream.

4.8. Mathematical Model

4.8.1. System Definition

Let the incoming webcam frame at time t be:

$$F_t \in R^{h \times w \times 3} \quad (5)$$

Defining the complete system as a function:

$$S : F_t \rightarrow O_t \quad (6)$$

4.8.2. System State Representation

The system operated in one of the two states:

$$Q = \{Visible, Occluded\} \quad (7)$$

State is assigned on the basis of probability by the occlusion classifier as:

$$\delta(F_t) = Visible, \text{ if } p_t \geq 0.3, \quad (8)$$

$$\delta(F_t) = Occluded, \text{ if } p_t < 0.3. \quad (9)$$

4.8.3. Functional Modules

Preprocessing:

$$X_t = \phi(F_t) \quad (10)$$

where $\phi(\cdot)$ performed resizing and normalization.

Occlusion Probability

$$p_t = f_o(X_t) \quad (11)$$

The MobileNetV2 classifier produces the visibility probability:

(c) Landmark Mapping

The landmark extraction function is defined as:

$$L: F_t \rightarrow (x_i, y_i) i_1^{468} \quad (12)$$

Three required points extracted by the system are:

$$l_c = L(152), \quad l_L = L(234), \quad l_r = L(454) \quad (13)$$

4.8.4. Neck Geometry as Feature Vector

Geometry vector is defined as:

$$G_t = [NSE_t, CND_t, NW_t] \quad (14)$$

where:

- NSE_t : horizontal neck symmetry

$$NSE = |x_{left} - x_{right}| \quad (15)$$

- CND_t : chin-neck vertical distance

$$CND = |y_{chin} - y_{neck}| \quad (16)$$

- NW_t : neck width

$$NW = \sqrt{(x_r - x_l)^2 + (y_r - y_l)^2} \quad (17)$$

4.8.5. Chain Transformation Function

Let s be the chain size chosen by the user:

Define the transformation mapping:

$$T: (G_t, s) \rightarrow C_t \quad (18)$$

This function encapsulated:

- Chain scaling
- Chain positioning
- Geometric alignment.

Thus,

$$C_t = T(G_t, s) \quad (19)$$

4.8.6. Final Output:

$$O_t = F_t \oplus C_t \text{ if } \delta(F_t) = Visible \quad (20)$$

$$O_t F_t \text{ if } \delta(F_t) = Occluded \quad (21)$$

5. Results and Discussion

5.1. Quantitative Evaluation of Occlusion Detector

The occlusion detection model was assessed using multiple performance indicators, including precision, recall, F1-score, accuracy and ROC-AUC to test the classifier's capability to correctly distinguish images containing neck-occluded regions from visible neck ones. For this evaluation, the trained classifier was evaluated on an expanded set of 1,200 images, comprising of 600 neck-visible and 600 neck-occluded images.

Table 3. Class wise classification metrics table

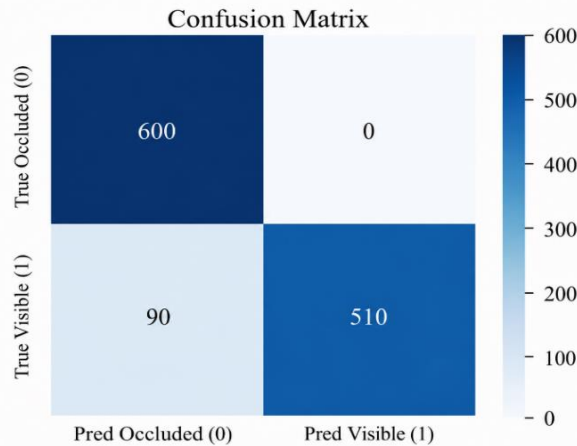
Class	Label	Precision	Recall	F1 - score	Support
0	neck_occluded	0.87	1.00	0.93	600
1	neck_visible	1.00	0.85	0.92	600

Table 3 shows the classification metrics for both classes. The model obtained a precision of 0.87 and a recall of 1.00 and resulted in an F1-score of 0.93, for the neck-occluded class which demonstrated that the model rarely has missed occluded necks but sometimes misclassified visible necks as occluded.

For the neck_visible class, the model obtained a precision of 1.00 and a recall of 0.85, with an F1-score of 0.92, which shows that all predictions labelled visible were correct but some were incorrectly classified as occluded. The proposed

classifier achieved an overall classification accuracy of 92.5%, on the evaluation dataset

The obtained results indicate that the proposed classifier effectively prioritizes the detection of neck occlusions, which is desirable for virtual jewellery applications. The high precision and recall values demonstrate that the classifier learned discriminative features both for neck-visible and neck occluded classes. The model consistently detected occluded necks thereby preventing unrealistic chain placement and improving the reliability.

**Fig. 3 Confusion matrix for the 1,200-image evaluation dataset**

The above Figure 3 presents the confusion matrix generated using the evaluation dataset of 1,200 images. The classifications of each class were illustrated visually, highlighting where mainly the misclassifications occurred. Most misclassifications occurred for the visible neck being predicted as occluded, which was for 90 cases whereas the occluded neck classified were always correct. This shows that the model favored predicting occlusion detection while being slightly conservative in predicting the visibility. The conservative prediction strategy reduces the possibility of displaying a chain over an occluded neck, maintaining realism and enhancing robustness. The confusion matrix provides an intuitive understanding of the model behaviour. Although a small number of visible neck images were classified as occluded and the absence of missed occlusion cases indicated that the model reliably detects situations where chain rendering should be suppressed.

The above Figure 4 illustrated a ROC curve for the neck occlusion detector model. The curve plots against the true positive and false positive rate and evaluated how the model

performs across a wide range of thresholds. The area under the curve was 0.98, which shows that the model can strongly distinguish and predict the visible and occluded parts of the neck. This high AUC value signified that this model could distinguish between neck occluded and neck visible images which makes it robust for real-world applications. The ROC curve showed that this model was highly reliable across different decision settings.

Overall, the model performed very well in detecting neck occlusions, with minor misclassifications occurring for visible necks. These results indicate the robustness of MobileNetV2 and that it captured the features well from the custom dataset used in this study. This performance was suitable for real-time virtual try-on applications, ensuring that chains are accurately placed only when the neck was visible.

This AR based virtual try-on system was evaluated through qualitative visual analysis and quantitative performance testing. The evaluation was performed to assess both technical robustness and real - time usability.

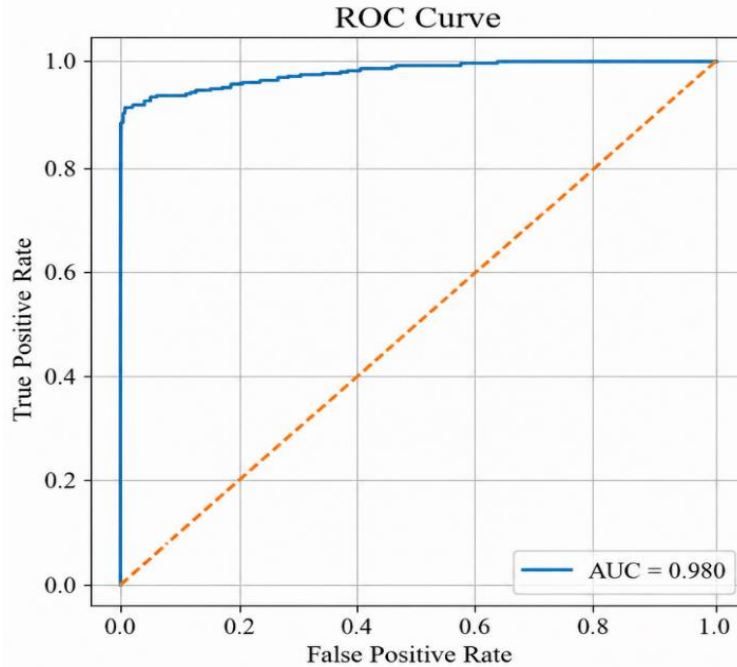


Fig. 4 ROC curve for the occlusion detector

5.1.1. Quantitative Evaluation

The system achieved an alignment accuracy of 94 % across 81 live trials conducted on users, where each user was tested for three different chain sizes. Of the 81 total overlays, 76 were accurately positioned in the neck region. The remaining cases showed slight misalignment due to hair, hand and clothing occlusions, where landmark detection was difficult.

The frame rate was 30 FPS on an average. This result showed that the Streamlit-WebRTC was giving a real-time consistent performance while maintaining stable chain alignment even when executed through Google Colab's remote interface.

This also validated that the use of Medipipe FaceMesh, OpenCV and Streamlit together was lightweight and could run smoothly without GPU acceleration. This has made it suitable for deployment on low resource systems.

Response Latency was ~150 ms which was between the frame capture and rendered output and this was recorded using timestamps. This observed delay was almost imperceptible to human users and was hence provided a try-on experience.

Frame rate and latency were evaluated over 300 consecutive video frames using Streamlit-WebRTC's built-in timestamping. These experiments were performed on Google Colab under a stable network. Latency measurements included both computation and network relay.

Table 4. Numerical comparison of key alignment parameters of the system

Frame	Neck Symmetry Error (px)	Chin-Neck Vertical Distance (px)	Neck Width
0	16.33	230.46	346.38
1	7.24	243.86	381.83
2	4.88	230.90	351.43
3	4.88	230.90	351.43

Table 4 shows the quantitative comparison of the different parameters of comparison for alignment of chain. They are as follows:

5.1.2. Neck Symmetry Error

Neck Symmetry Error measures the difference between left and right neck alignment points in pixels.

Equation (15) calculated the neck symmetry for this system. A high value indicates asymmetry, whereas a lower value indicates stable and centered alignment. In Table 4 it decreases from 16.33 to 4.88 which demonstrates improved stability after the initial misalignment.

5.1.3. Chin-Neck Vertical Distance

The Chin-Neck vertical distance is the distance from the chin to the neck landmark.

Equation (16) was the formula which calculated chin-neck vertical distance. Consistency across frames (230-244

px) confirmed that the landmarks do not drift and that the chain overlays maintained realistic proportions and accurate positioning.

5.1.4. Neck Width

Neck width, which is the horizontal distance between the left and right neck landmarks.

Equation (17) shows the formula for neck width. The minimal variation (346 - 381px) validates the robustness of the

system in real-time. Therefore, these metrics provide quantitative evidence for the chain placement and the accuracy of landmark detection along different frames. They demonstrate the stability and consistency of the proposed approach. The decrease in neck symmetry error with the relatively stable chin-neck distance and neck width indicated accurate and reliable chain placement across consecutive frames. Frames 2 and 3 show identical measurements across three parameters indicating that alignment had reached a stable state.

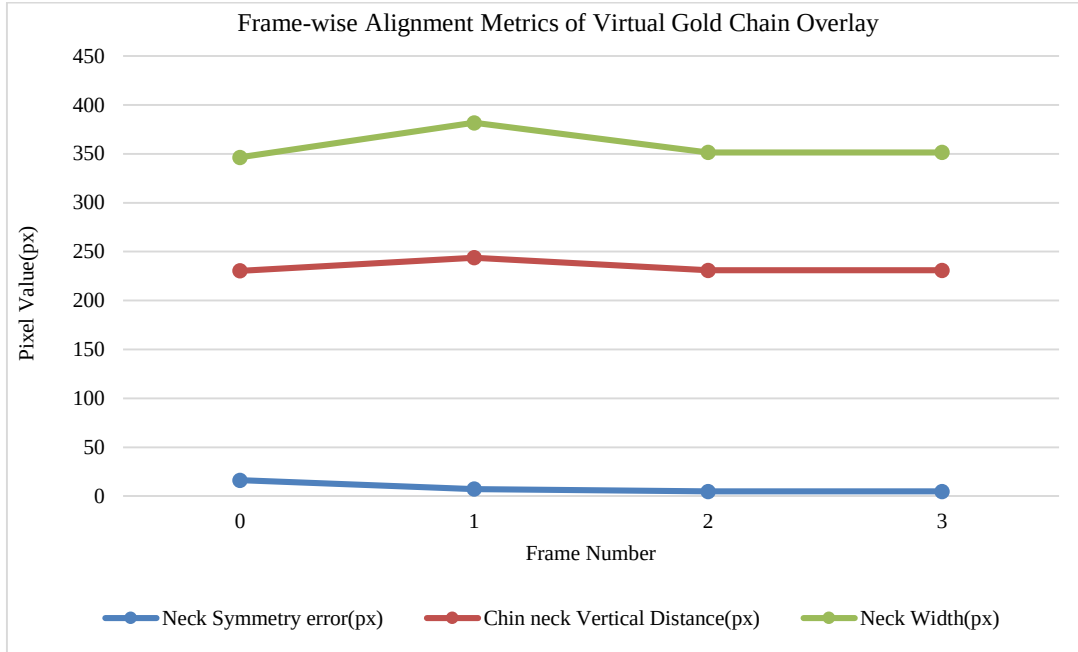


Fig. 5 Line graph of three metrics

Figure 5 shows a line graph of the three metrics of the system. The neck symmetry error decreases over frames, showing stable detection and both chin-neck vertical distance and neck width remain consistent across three frames resulting in realistic chain alignment.

5.2. Qualitative Evaluation

The qualitative evaluation is the alignment of the chain on the neck, how accurately and naturally the chain is aligned on the neck and realism

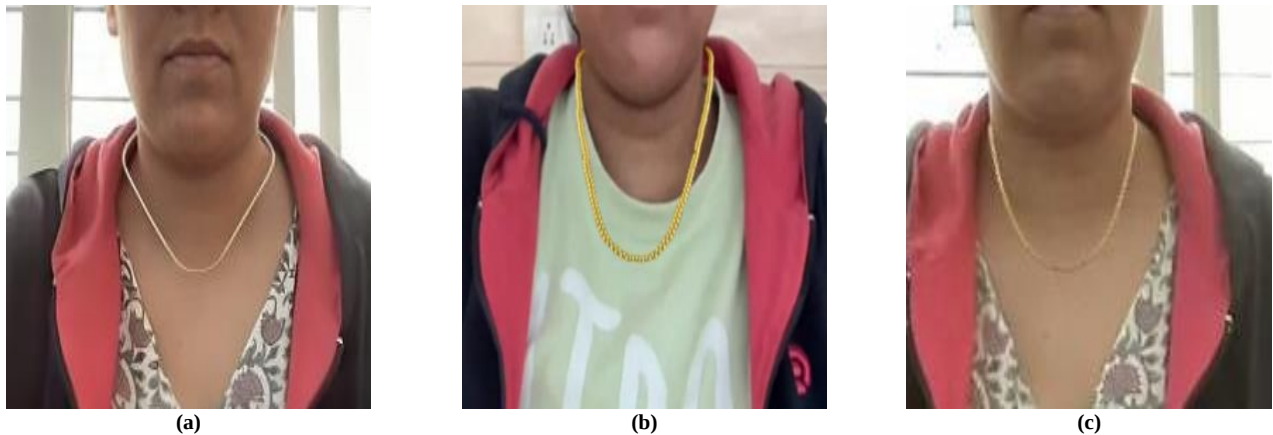


Fig. 6 Three different chains in three different sizes: (a) 16 inch, (b) 18 inch, and (c) 22 inch.

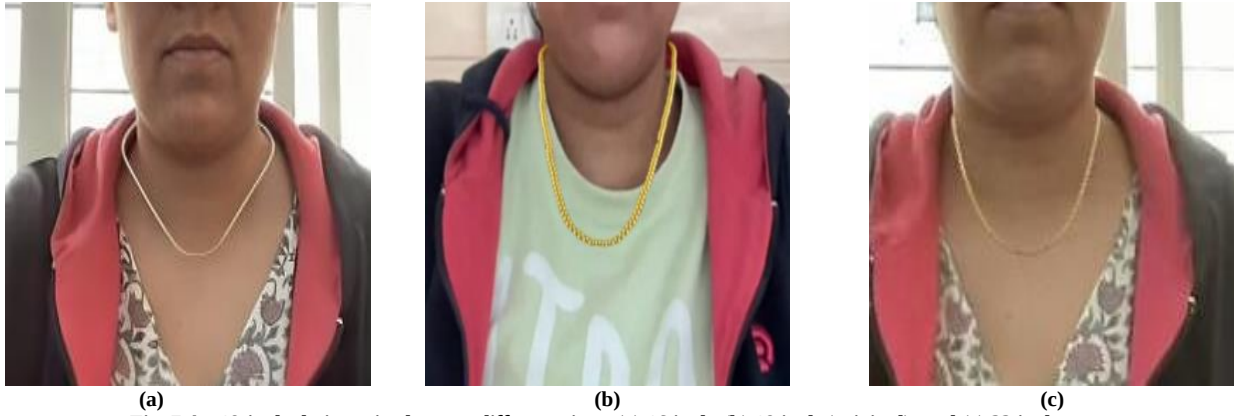


Fig. 7 An 18-inch chain resized to two different sizes: (a) 16 inch, (b) 18 inch (original), and (c) 22 inch.

Visuals of the overlay result in Figures 6 and 7 showed that the gold chains were well-aligned and adapted realistically with different chain sizes and different types of chains. The selected landmark numbers (152, 234 and 454) anchored along the chin baseline and vertical and horizontal points to the left and right side of the neck effectively which showed accurate placement.

The system correctly differentiated between different chain types and sizes which made the scaling transitions smooth.

Neck Occlusion Cases

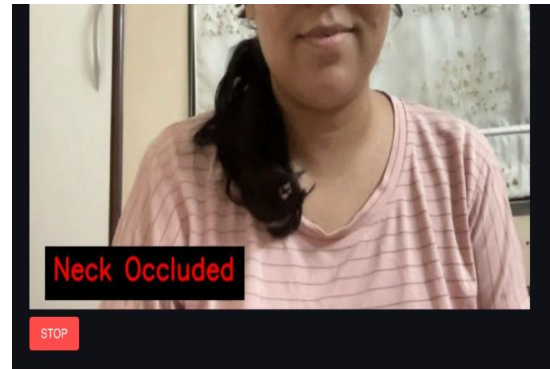


Fig. 10 Neck occluded because of hair

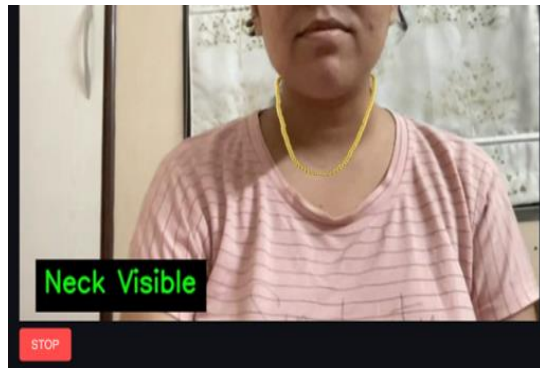


Fig. 8 Neck visible and chain aligned

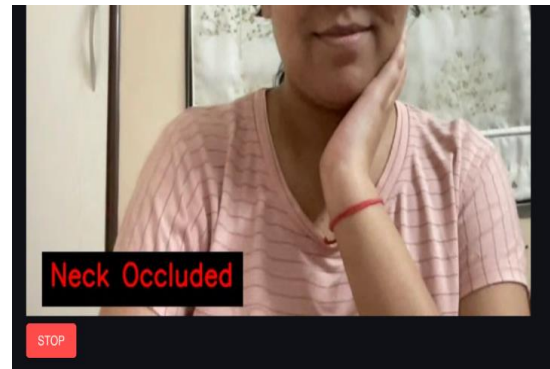


Fig. 11 Neck occluded because of hand

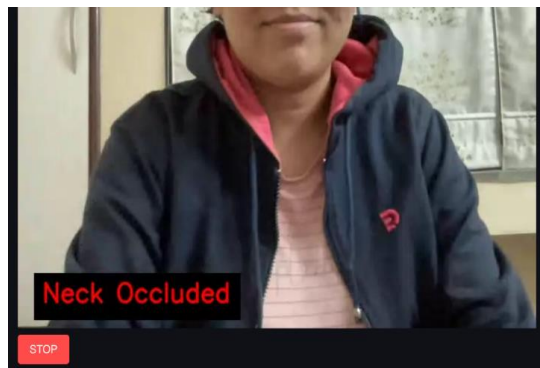


Fig. 9 Neck occluded due to hoodie

Figure 8 had the neck visible hence the chain gets aligned to it along with a message display that the neck was visible. Figure 9 has a neck occluded due to a hoodie and neck was not visible hence the chain was not visible and there was a message displayed of the neck being occluded. Similarly Figures 10 and 11 show hair and hand occlusion which results in no chain alignment display and the message of neck occlusion. These results validated that the occlusion detector successfully suppresses chain rendering whenever the neck region is obstructed by hair, clothing and hand movements. This behaviour has significantly improved visual realism of the proposed system and prevented unrealistic overlays.

5.3. Discussion

Several iterations were performed during the development of the system to achieve accurate alignment and stability to ensure a seamless interaction. The final implementation has achieved all of these and addressed multiple technical challenges that occurred during the development phase.

Technical challenges were identified during the early experimentation phase. Initially, OpenCV's `cv2.imshow()` was the approach for live streaming, but it proved to be incompatible with Colab and also frequently crashed the notebook. This issue was then resolved with the help of Streamlit-WebRTC integration which allowed browser streaming efficiently. Through the PyAV library, a continuous streaming of 30 FPS frame rate was achieved. Rendering accuracy was another focus for improvement.

Previously, the outputs showed an opaque overlay because OpenCV did not process transparency properly. Incorporating Pillow (PIL) for RGBA based blending made it naturally transparent and improved overlay realism. The performance was optimized by limiting the frame size to 640×480 . This maintained the quality visually while reducing the processing delay.

Performance delay was observed when MediaPipe processed high resolution frames, which caused the latency to exceed 300 ms. Downscaling the frame size balanced the time delay and the visual output. The latency was calculated using timestamps between the input capture and output display, which was ~ 150 ms, making it suitable for real-time use.

Further refinements were made in the height and width scaling logic as it led to the chain appearing slightly offset or misaligned. Adjustments were made to the scaling ratios and vertical offsets which aligned the chain accurately to the neck's curvature in all three available sizes.

A major challenge during occlusion detection was the inconsistent visual quality of the webcam frames. Lighting and shadows were some of the factors affecting and creating different variations in the neck region and often caused the model to misinterpret. To address this, a Global Average Pooling (GAP) layer was added so that the model learned global occlusion patterns and not pixel-level noise. The dropout rate also regularized the model.

The custom dataset was very small at first and contained many variations, which led the classifier to show signs of rapid overfitting where the training accuracy became high and validation accuracy remained low; hence, the model could not generalize well. To address this issue, the pretrained layers of MobileNetV2 were frozen and only the lightweight classification layers were trained. Heavy augmentation increased the dataset size and diversity and reduced

overfitting. This led to a well-generalized model. Integrating the occlusion classifier had introduced a performance overhead in the Streamlit-WebRTC pipeline. This led to a slow real-time prediction and increased latency. To solve this problem, the model was preloaded to avoid repetitive loading which reduced latency and made it suitable for real-time use.

A threshold value selection for classification was performed to distinguish between visible and occluded frames. To determine optimal threshold, values between 0.1 to 0.5 were evaluated. It was observed that occluded frames produced p less than 0.3 and visible frames produced it above 0.3. A threshold of 0.3 offered best balance, maximized the F1 score and ensured the detection of occluded necks while controlling the false positives. Reducing the thresholds resulted in more false positives while larger threshold values reduced sensitivity to partially occluded necks occasionally. Hence, this value was adopted.

Overall, these challenges made the system more robust and improved the user experience. These refinements made the real-time AR experience capable of producing realistic visuals.

Quantitatively comparing with the existing approaches, highlighted advantages of this approach. Ghatge et al. [2] achieved a stability rate of 90% with a response time of 200ms for the proposed jewellery try-on system. Prajapat et al. [5] obtained an accuracy of 95.7% in hand jewellery detection using MediaPipe. This proposed work reached 92.5% of accuracy with an AUC of 0.98 for occlusion classification. It delivered a real-time performance of 30 FPS and a latency of 150 ms. This therefore offered a competitive performance and enhanced occlusion detection and precise neck alignment compared to prior work. The competitive performance of this framework resulted from the integration of MediaPipe FaceMesh for accurate neck landmark localization and the MobileNetV2-based neck occlusion detector, which together improved chain alignment while maintaining real-time performance. Moreover, optimization techniques included frame resizing, transfer learning and data augmentation enhanced computational efficiency, model robustness and real-time responsiveness.

6. Conclusion

This study presented a real-time gold chain virtual try-on system using Augmented Reality with a neck occlusion detector. This system can help users to virtually try-on chains of different sizes by detecting the visible/occluded neck and visualizing the chain through a live webcam. This ensured the accurate alignment of the chain across different faces and different chain sizes. However, obstructions in the neck classified the frame as occluded. This system could be integrated into an online jewellery retail approach that will enhance confidence in users while purchasing the product and also help in reduction of return rates with an increase in

purchase rate. Potential challenges include the availability of only three fixed chain sizes, lack of realism owing to 2D PNG chain images and real-time performance lag depending on the lower-end devices.

6.1. Limitations

This evaluation primarily focuses on static frames. Rapid head movements, turning tilting can make the landmark detection unstable and cause misalignment in the chain overlay. The system was currently working for three fixed sizes only (16, 18 and 22) and it cannot dynamically resize the chain for different neck sizes. The chain overlay was in 2D PNG and hence, lacks realism. The system was optimized only for gold chains and was not generalized to other jewellery items such as pendants, earrings, nose pins and bracelets. Real-time performance depends on the lower-end devices too and can face Streamlit-WebRTC lag. Landmark detection becomes less reliable under poor or extreme lighting conditions.

6.2. Future Work

A detailed evaluation of dynamic performance which will assess the landmark stability, lag, jitter during rapid head movements will be included as a part of future scope as it will evaluate chain under more realistic and movement intensive conditions. Currently, this prototype relies on Google Colab, Streamlit-WebRTC for experimentation. Future work will be developing a production-ready system with a dedicated backend, GPU-enabled servers and optimized rendering for real-time application. The future scope also includes extending this to 3D chain models to achieve more realism in the virtual try-on experience. It could also be extended to

integrate with e-commerce platforms. The inclusion of multiple jewellery items such as bangles, earrings, etc. will increase the applicability of the system.

Author Contributions

Maitreyee Londhe developed and implemented the system, conducted the experiments and evaluation and wrote and edited the manuscript under the guidance and direction of Anjali Naik. Anjali Naik conceived the research idea, supervised the implementation, directed the structure and revision of the manuscript and approved the finalized content. Both authors read and approved the final version of the manuscript.

Conflicts of Interest

No potential conflict of interest was reported by the authors in connection with this work.

Funding Statement

This work did not receive any specific grant or financial assistance from funding agencies in the public, commercial and non-profit sectors.

Acknowledgment

The authors sincerely thank Mr. Satish Ranade, Adjunct Professor at MKSSS's Cummins College of Engineering for Women, for suggesting the original concept that motivated this work, and the Department of Computer Engineering, Cummins College of Engineering for Women, for providing support and facilities required for the successful completion of this project.

References

- [1] Raj Raval et al., "Developing Markerless Augmented Reality for Furniture Mobile Application," *International Journal of Education and Management Engineering (IJEME)*, vol. 11, no. 1, pp. 11-18, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [2] Dipali Ghatge et al., "Virtual Jewellery Try-on: Enhancing Consumer Engagement and Personalization," *International Research Journal on Advanced Engineering Hub*, vol. 2, no. 12, pp. 2811-2815, 2024. [[CrossRef](#)] [[Publisher Link](#)]
- [3] Bhadra Rajan, and Pauline Paul, "Real-Time Virtual Jewellery Try-on System using Augmented Reality," *Proceedings of the National Conference on Emerging Computer Applications (NCECA)*, vol. 5, no. 1, pp. 636-638, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [4] Dipali Basumatary, Debojeet Das, and Ranjan Maity, "Impacts of AR in Learning Tribal Bodo Language," *International Journal of Modern Education and Computer Science (IJMECS)*, vol. 16, no. 6, pp. 64-79, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [5] Jai Prajapat et al., "Jewellery Try-on using AR," *International Journal of Computer Engineering in Research Trends*, vol. 9, no. 4, pp. 66-72, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [6] Nimesh Yadav, and Aryan Sinha, "Augmented Reality and its Science," *International Journal of Education and Management Engineering (IJEME)*, vol. 12, no. 6, pp. 33-44, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [7] Momina Mushtaq et al., "Attack Modeling and Security Analysis using Machine Learning Algorithms Enabled with Augmented Reality and Virtual Reality," *International Journal of Computer Network and Information Security (IJCNIS)*, vol. 16, no. 4, pp. 118-129, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [8] Hongxia Chu, "Research on 3D Jewelry Design based on Virtual Reality Technology," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, pp. 1-8, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [9] Chandapatla Jaideep Sai et al., "Enhancing Jewellery E-Commerce Experiences: Integrating Augmented Reality (AR) with MyWebAR and Virtual Reality (VR) with Three.js Library," *International Research Journal of Engineering and Technology (IRJET)*, vol. 12, no. 3, pp. 365-370, 2025. [[Publisher Link](#)]

- [10] Asif A. Lagharic et al., “Systematic Analysis of Virtual Reality & Augmented Reality,” *International Journal of Information Engineering and Electronic Business (IJIEEB)*, vol. 13, no. 1, pp. 36-43, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [11] Jeel L. Paul, and Sona Maria Sebastian, “AI-Powered Ring Size Estimation for a Realistic Virtual Try-On,” *Proceedings of the National Conference on Emerging Computer Applications (NCECA)*, vol. 6, no. 1, pp. 494-499, 2025. [[CrossRef](#)] [[Publisher Link](#)]
- [12] Diona Tens, and Jinson Devis, “Virtual Try-on Solutions for Kids’ Wear,” *Proceedings of the National Conference on Emerging Computer Applications (NCECA)*, vol. 6, no. 1, pp. 454-456, 2025. [[CrossRef](#)] [[Publisher Link](#)]
- [13] Saiprasad Patil et al., “Augmented Reality Try-On Watch Application,” *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, vol. 2, no. 2, pp. 138-144, 2022. [[CrossRef](#)] [[Publisher Link](#)]
- [14] Ting Chou, Chih-Hsing Chu, and Shengjun Liu, “Virtual Footwear Try-on in Augmented Reality using Deep Learning Models,” *Journal of Computing and Information Science in Engineering*, vol. 24, no. 3, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [15] Gayathri G. Nair, and Lisha Varghese, “Virtual Try on Hairstyles using Augmented Reality,” *Proceedings of the National Conference on Emerging Computer Applications (NCECA)*, vol. 6, no. 1, pp. 479-483, 2025. [[CrossRef](#)] [[Publisher Link](#)]
- [16] Mahek Agarwal et al., “Virtual Trial Room: Real-Time Image-based Virtual Cloth try-on System,” *International Journal of Innovative Science and Research Technology (IJISRT)*, vol. 8, no. 5, 2023. [[CrossRef](#)] [[Publisher Link](#)]
- [17] Prakriti Thapa et al., “A Web-based AR-Powered Virtual Eyewear Try-on System,” *International Journal on Engineering Technology*, vol. 2, no. 2, pp. 114-125, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] Deepa Sunny et al., “Style Up: A Virtual Try-On System,” *Research and Reviews: Advancement in Robotics (RRAR)*, vol. 7, no. 3, pp. 11-19, 2024. [[Publisher Link](#)]
- [19] Charlee Kaewrat, Poonpong Boonbrahm, and Bukhoree Sahoh, “The Design and Development of a Foot-Detection Approach based on Seven-Foot Dimensions: A Case Study of a Virtual Try-on Shoe System using Augmented Reality Techniques,” *Informatics*, vol. 10, no. 2, pp. 1-14, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [20] Stefan Hoffmann, and Robert Mai, “Consumer Behavior in Augmented Shopping Reality: A Review, Synthesis, and Research Agenda,” *Frontiers in Virtual Reality*, vol. 3, 2022. [[Google Scholar](#)]
- [21] Fadilah Haya Kamilah, Afina Hasya, and Ismi Darmastuti, “The Contribution of Virtual Fitting Room in Fashion Clothing Business: A Systematic Literature Review,” *International Journal of Economics, Business and Accounting*, vol. 2, no. 2, pp. 184-201, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]