

VIRTUAL TRIAL ROOM FOR ONLINE SHOPPING

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering

By

Jithesh T (Reg. No - 40110500)

Kavin AM (Reg. No –

40110581)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Category-1 University by UGC

**Accredited with Grade “A++” by NAAC | 12B Status by UGC | Approved by
AICTE**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119**

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Kavin AM(Reg.No - 40110581)** and **Jithesh T (Reg.No - 40110500)** who carried out the Project entitled **Virtual trial room for online shopping** under my supervision from November 2023 to April 2024.


12/4/24
Internal Guide

Dr.G.Kalaifarasi M.E., Ph.D.,


HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Head of the Department
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Jeppiaar Nagar, Rajiv Gandhi Salai
Chennai - 600 119.
Dr. LAKSHMI VANITHA M.E., Ph.D.,

Submitted for Viva voice Examination held on 16/04/2024


16/4/24
Internal Examiner


16/4/24
External Examiner

DECLARATION

I, Kavim AM (Reg. No- 40110581) hereby declare that the Project Report entitled "VIRTUAL TRIAL ROOM FOR ONLINE SHOPPING" done by me under the guidance of DR.G.KALAIARASI is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

Date : 16/04/2024

AM. Kavim

PLACE: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **The Board of Management of Sathyabama institute of science and technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.G.Kalaiarasi M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

It normally takes a lot of time to try on things in a store. However, in situations like Online Shopping, it might not even be able to try on clothing. The popularity of online purchasing has greatly increased nowadays especially after Covid lockdowns and pandemic breakdowns. When buying things like clothing, it's important to understand how the clothing will look on the buyer. This is the main cause of the decline in online clothing sales. So, a virtual trial room for online shopping would be a tremendous luxury for online retailers that could offer a large selection to users. This would allow people to see how clothes fit them individually. This would be a fantastic tool for Online shopping marketers to expand their market. By establishing a virtual trial room setting, we hope to improve accessibility and time efficiency for trying on clothing. In this project, we provide an Open-CV-based virtual trial room application for online shopping. The Virtual trial Room is an innovative solution that offers a personalized and immersive shopping experience to online shoppers. With the increasing popularity of e-commerce, customers face the challenge of not being able to try on clothes before purchasing them. The Virtual trial Room addresses this issue by allowing customers to virtually try on clothes in a digital environment. Additionally, this solution saves time and increases convenience for customers who do not have the time or inclination to physically visit a store. The Virtual Trial Room offers several benefits to customers, including the ability to visualize how clothes would look on them before making a purchase, the convenience of shopping from home, and the ability to compare different styles and sizes easily. It also benefits e-commerce businesses by reducing the likelihood of returns due to sizing issues and increasing customer satisfaction. Overall, the Virtual trial Room is an innovative solution that bridges the gap between online shopping and traditional in-store shopping. It provides a personalized and engaging shopping experience to customers, which can significantly improve the overall customer experience and boost sales for e-commerce businesses.

Table Of Contents

Chapter No.	TITLE	Page no.
	Abstract	v
	List of figures	vii
1	Introduction	1
2	Literature Survey	2
	2.1 Inferences from literature survey	4
	2.2 Open problems in existing system	5
	2.3 Disadvantages of existing system	6
3	Requirement analysis	7
	3.1 Feasibility Studies/Risk Analysis of the Project	7
	3.2 Software Requirements Specification Document	8
	3.2.1 Requirement analysis	8
	3.2.2 Hardware requirements	9
	3.2.3 Software requirements	9
	3.2.4 Functional requirements	10
	3.2.5 Non functional requirements	11
4	Description of proposed system	12
	4.1 Advantages of proposed system	12
	4.2 Selected methodology or process model	13
	4.2.1 Detecting and sizing the body	14
	4.2.2 Face detection	14
	4.2.3 Image masking	15
	4.2.4 Edge detection	15
	4.2.5 Scaling of attire	16
	4.2.6 Body parts tracking	16

4.2.7	Date preparation	16
4.2.8	Virtual cloth overlay	16
4.2.9	User Interface	16
4.2.10	Real time processing	17
4.3	Activity diagram	17
5	Implementation design	18
5.1	Development setup	18
5.1	Dependency requirement	18
6	Result and discussion	21
6.1	Implementation of virtual trial room for online shopping	21
7	Conclusion	24
	References	25
	Appendix	26
A	Source code	26
B	Screenshots	54
C	Research paper	58

List of figures

Figure No.	Figure Name	Page
3.2.1	System architecture	10
4.3.1	Activity diagram	17
B.1	Home page of our application	54
B.2	Necklaces products in the application	54
B.3	Sunglasses products in the application	55
B.4	Accessories products in the application	55
B.5	Hair bands products of our application	56
B.6	shirts products of our application	56
B.7	Skirts products of our application	57

CHAPTER 1

INTRODUCTION

Augmented Reality based online shopping platform” is a Next-Gen technology which will soon be used in different e-Commerce websites and physical fashion stores. We are using the idea of Augmentation for giving the people with tool using which they will be able to see the real-time simulation of different clothes with having to wear them physically. It’s a Web-based application which can be accessed from anywhere in the world, all one need is a web browser and internet connection. Augmented reality is an interactional encounter of an everyday world where the entities that inhabit the physical sphere are developed by simulated perceptual knowledge, sometimes across numerous sensory systems, including visual, auditory, haptic, and others. It is a flourishing development among e-commerce companies involved in business for attracting customers and understanding their needs. It is a highly graphic, interactive method of presenting relevant digital information in the context of the physical environment—generating customers, clarifying their doubts, and improving business outcomes in e-commerce sites.

CHAPTER 2

LITERATURE SURVEY

Intelligent human body tracking, modelling and activity analysis of video surveillance system[1] A Survey done by K.Srinivasan, K.Porkumaran, G.Sai Narayanan states that theRecent technology and market trends have demanded the need for solutions to video/camera systems and analytics. They are highlighting the video system architectures, tasks, and related analytic methods. It demonstrates that the importance of the role that intelligent video systems and analytics play can be found in a variety of application such as homeland security, crime prevention, traffic control, accident prediction and detection, and monitoring patients, elderly and children at home, etc. Research directions are outlined in the end with a focus on what is essential to achieve the goals of intelligent video systems and analytics.

Segmentation by Fusion of Histogram based K-Means Clusters in different color space[2] A survey done by Max Mignotte states that they are presenting a new, simple, and efficient segmentation approach, based on a fusion procedure which aims at combining several segmentation maps associated to simpler partition models in order to finally get a more reliable and accurate segmentation result. The different label fields to be fused in the application are given by the same and simple (K-means based) clustering technique on an input image expressed in different color spaces. The fusion strategy aims at combining these segmentation maps with a final clustering procedure using as input features, the local histogram of the class labels, previously estimated and associated to each site and for all these initial partitions. The fusion framework remains simple to implement, fast, general enough to be applied to various computer vision applications (e.g., motion detection and segmentation), and has been successfully applied on the Berkeley image database. The experiments herein reported in this paper illustrate the potential of this approach compared to the state-of-the-art segmentation methods recently proposed in the literature.

From 2D Photos of Yourself to Virtual Try-on Dress on the Web[3] A survey done by F. Cordier, W. Lee, H. Seo, and N. Magnenat-Thalmann states that They are describing the complete methodology for cloning and dressing people by using a website. The input is simple photographs or body measurements that anybody can produce in any environment. Then the Web-based virtual-try-on allows users to see them dressed. The basic methodology uses a pre-calculated generic database to produce personally sized bodies and animate garments on a Web application.

A body and garment creation method for an internet based virtual fitting room[4] A survey done by D. Protopsaltou, C. Luible, M. Arevalo-Poizat, and N. Magnenat Thalmann states that the methodology for producing 3D clothes, with realistic behavior, providing users with an apposite feel for the garment's details. The 3D garments that are produced,originate from 2D CAD patterns of real garments and correspond to the regular sizes that can be found on a shop floor.

Made-to-measure technologies for an online clothing store[5] A survey done by F. Cordier, H. Seo, and N. Magnenat-Thalmann states that the Internet is a compelling channel for selling garments. Several recent initiatives by companies such as Nordstrom, Macy's, and Lands End focus on made-to-measure manufacturing and shopping via the Internet. The Web application that provides more powerful access to and manipulation of clothing to facilitate clothing design, pattern derivation, and sizing. Applying 3D graphics technology to help create and simulate the virtual store.

Enhanced Background Subtraction Techniques for Monocular Video Applications[6] K.Srinivasan, K.Porkumaran, G.Sai Narayanan states that attempts to find moving objects by subtracting the background images from static single camera video sequences in security systems. It aims to improve the background subtraction techniques for indoor video surveillance applications. For dynamic video sequences, the object focus and object tracking are very difficult due to changing of various parameters such as camera noise, illumination changes, motion changes, and geometry of background changes. The background subtraction algorithm provides the solution to detect moving objects in the static scene. The recursive and non-

recursive algorithms are discussed in an improved way. The novel automatic threshold updating (ATU) algorithm is also developed and tested for various indoor video sequences which gives better efficiency. The statistical and temporal differencing methods are also presented.

Skin colour segmentation based 2D and 3D human pose modelling using Discrete Wavelet Transform[7] K.Srinivasan, K.Porkumaran, G.Sai Narayanan states Tracking of human body and their pose models from a video data is important in the indoor surveillance applications. The objective of this paper is to develop a human pose modelling in two-dimensional and three dimensional views using wavelet based skin colour segmentation for the monocular videos. The approximation information of the Discrete Wavelet Transform (DWT) is used for segmenting the human body from the video frame and to find the skin features. The human pose modelling is proposed by detecting thirteen features from the upper body as well as in the lower body. Here, twelve human poses are considered for the analysis. In this paper, the hardware system is implemented for monitoring the human poses by sending the SMS message. Also, the efficiencies of 2D modelling and 3D modelling for the pre-defined poses are evaluated.

2.1 Inferences from literature survey:

Aim is to build a compelling, interactive and highly realistic virtual shop, where visitors can choose between many different types of garments designs and proceed to simulate these garments on virtually animated bodies. By merging the approach often used by the fashion industry, in designing clothes, and the methodology for creating dressed virtual humans, we present a new technique providing trouble-free and straightforward garment visualization.

The entire process starts from the creation of virtual bodies (either male or female), using standard measurements, which form the basis for garment modeling. Using

splines, the 2D garment patterns are created and then seemed together around a virtual human body, providing the initial shape. A simulation is made using the seemed garment by applying physical parameters based on real fabric properties. Once the garment has been created, a real time platform, which has been embedded inside a web browser, is used as an interface to the Internet and extend the web interface to add interactivity and the ability to dynamically change textures, clothes, body measurements, animation sequences and certain features of the virtual environment.

The whole methodology does not aim to build only a virtual dressing room, where customers can view garments fitted onto their own virtual bodies but to visualize made-to-measure clothes, animate them, visualize the cloth behavior and to add interactivity. The entire Virtual Try-On experience is a process of targeting the clientele, designing the clothing collection, dressing the virtual models and then using the web as a virtual shop floor.

2.2 OPEN PROBLEMS IN EXISTING SYSTEM:

In the existing there are various applications similar to the Virtual Trial Room, some of them are being referred. The Fitter Room focuses mainly on the Kinect software used the Kinect sensors are used to find the body functions and 3d modeling techniques are used for superimposing an image. Blender software is mainly used for 3d modeling and also hardware components i.e., sensors are used whereas we will not be using any hardware components to keep it user- friendly and easy to install. The Machine Learning approach contains machine learning algorithms to detect facial and eye movements to gain coordinates for the user. It has an approach for more than 60,000 algorithms to gain accuracy. Image Processing Techniques contains techniques of image processing using filters such as Gaussian filters, mean filters, Gaussian smoothing, Conservative Smoothing, unsharp filter, Laplacian/Laplacian of gaussian filter, high pass and low pass filter, edge enhancement filter, edge detection filter, etc. Virtual Trial Room using augmented reality The Kinect Sensor are paired with a depth sensor as well as an RGB sensor.

When the user is allowed to stand in front of the webcam, the Kinect will track the body joints and will show the skeleton of the user's body.

2.3 DISADVANTAGES OF EXISITNG SYSTEM

- The main disadvantage of the existing system is it will not work without the use of hardware components. Some systems, such as the ones that use augmented reality technology, may require additional hardware such as sensors to function properly. This can make the system less accessible or user-friendly for some customers.
- The existing system has lower accuracy when implemented in the real time application.
- The existing system may depend on user-uploaded images, which may not always be of high quality or accurate, leading to inaccurate virtual try-ons.
- The existing systems may have been costly to implement, which could limit their adoption by smaller retailers or those with limited budgets.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 RISK AND ANALYSIS OF PROJECT AND FEASIBILITY STUDY:

The objective of the project is to provide the world with a great convenience of trying clothes without even wearing them physically. Creating an Augmented Reality based on the application which can be used to visualize the clothes on your body virtually using web application. This will give user the freedom of trying clothes and will make them able to try as many clothes as possible in lesser amount of time.

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ◆ ECONOMICAL FEASIBILITY
- ◆ TECHNICAL FEASIBILITY
- ◆ SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY :

The study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY :

The study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY :

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.2 SOFTWARE REQUIREMENTS SPECIFICATIONS DOCUMENT

3.2.1 REQUIREMENT ANALYSIS

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analysing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular

software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

Describing the overall features of the software is concerned with defining the requirements and establishing the high level of the system. During architectural design, the various web pages and their interconnections are identified and designed. The major software components are identified and decomposed into processing modules and conceptual data structures and the interconnections among the modules are identified. The following modules are identified in the proposed system.

3.2.2 HARDWARE REQUIREMENTS:

System	: Pentium i3 Processor
Hard Disk	: 500 GB.
Monitor	: 15" LED
Input Devices	: Keyboard,
Mouse Ram	: 2 GB

3.2.3 SOFTWARE REQUIREMENTS:

Operating system : Windows

10 Coding Language: Python

Describing the overall features of the software is concerned with defining the requirements and establishing the high level of the system. During architectural design, the various web pages and their interconnections are identified and designed. The major software components are identified and decomposed into processing modules and conceptual data structures and the interconnections among the modules are identified. The following modules are identified in the proposed system.

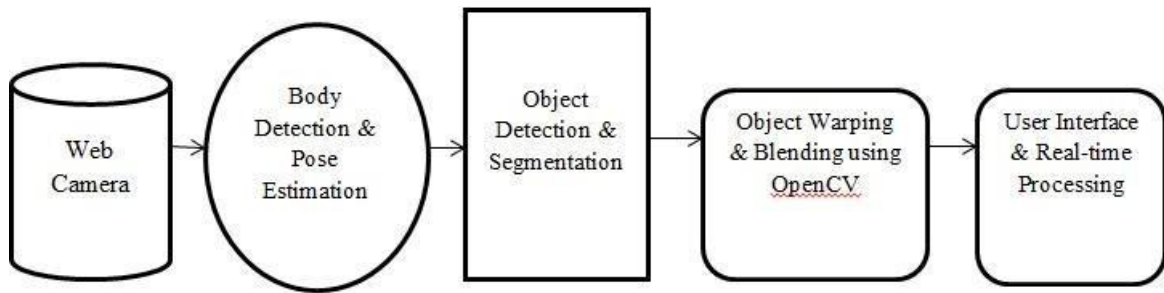


Fig 3.2.1 System architecture

3.2.4 FUNCTIONAL REQUIREMENTS

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

Usability:

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

Robustness:

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

Security:

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

Reliability:

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and even capable of working around any failure.

Compatibility:

It is supported by version above all web browsers. Using any web servers like localhost makes the system real-time experience.

Flexibility:

The flexibility of the project is provided in such a way that it has the ability to run on different environments being executed by different users.

Safety:

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one's personal information.

3.2.5 NON- FUNCTIONAL REQUIREMENTS

Portability:

It is the usability of the same software in different environments. The project can be run in any operating system.

Performance:

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

Accuracy:

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

The goal of the virtual trial room system is to provide users with an intuitive web application. The system is created with Flask, HTML, CSS, and JavaScript using various features and buttons to make it appealing and eye-catching for users. The interface is created in local languages, specifically English, and can be readily understood by any user. The "Tryon" button must be selected by the user if they want to try on clothing. The user's body is overlaid with an image of the clothing using OpenCV input from the camera. If the user likes the clothing, they can either buy it or try it on right now. The proposed system is focused on developing web application that records a video of the user using the device's camera, splits the video into individual frames, and then extracts the user's body from those frames. Using functions to morph, rotate, and scale the wearable image onto the user in real-time, as well as functions to extract information on the locations of joints in the body. The primary goal of this project is to create a working prototype for an online virtual trial area where buyers may virtually try on clothing. This effort will be highly helpful both during and after the COVID 19 epidemic. For the customers, it will make things simpler and more effective.

4.1 ADVANTAGES OF PROPOSED SYSTEM

- The proposed system implementation of the Virtual Trial Room system helps the user to explore billions of options sitting right at the home rather than physically visiting the shops.
- By using this application users can have a look at how an attire fits and looks on them.
- This application is useful in this pandemic situation as well where you don't have an option to try on clothes while buying them. During COVID-19 pandemic, offline stores will limit trials. During those times this technology will be very useful.

- Also, it saves the retailer from damaged garments due to the try-on facility given to the customers in their shops.
- Building a system that has consistently higher precision and individual members are more satisfied.
- Comparing this project to other existing systems, one of the main differences is the absence of any hardware components in our project.

4.2 SELECTED METHODOLOGY OR PROCESS MODEL:

In this project we are using Augmented Reality for implementation of online shopping. Here the methodology of surface detection and object detection has been used. In surface detection and object detection the system analyses the surface/object and finds the dimensions of the object which will help in the projection of objects in the realworld environment.

- Detecting and Sizing the Body
- Face Detection
- Image Masking
- Edge Detection
- Scaling of Attire
- Body Parts Tracking
- Data Preparation
- Virtual clothes overlay
- User Interface
- Real time processing
- Results

4.2.1 Detecting and Sizing the Body

First step of the proposed Online Virtual Trial Room method is the acquisition of the shape of the body, head or neck depending upon the wearable to get reference points. Reference points are then used to determine where to display the particular cloth or ornament. In order to obtain the body shape, we applied several techniques:

i) Filtering with thresholding, Canny edge detection, K-means, and ii) Motion detection or skeleton detection wherein multiple frames were analysed for any movement. However, the results were unreliable and not good enough to obtain reference points for displaying the wearable. Therefore, we introduced a new detection methodology based on locating the face of the user, adjusting a reference point at his/her neck and displaying the wearable based on that point. In addition, another point of reference can be obtained by using an Augmented Reality (AR) marker. Though this was enough for small attires like glasses or ornaments but it was not enough to map the clothes onto the user body. For obtaining the size of the user, we follow a similar automated body feature extraction. The idea is to set up the user in front of the camera and hold him at the beginning at a certain predetermined distance. The algorithm extracts points on the shoulders and the belly. Measuring the distance between these points and knowing the distance from the user to the camera, the size of the user can be obtained. When the image (video frame) is acquired, a canny edge detection filter is applied to obtain only the silhouette of the body. Canny edge detection is really susceptible to noise that is present in unprocessed data; therefore, it uses a filter where the raw image is convolved with a Gaussian filter. After convolution, four filters are applied to detect horizontal, vertical and diagonal edges in the processed image. Morphological functions are also applied to obtain a closed silhouette. Finally, an 8-points Freeman chain code, is applied to assign a direction to each pixel.

OpenCV's built-in Haar cascades were used to detect the user's body and isolate it from the background

4.2.2 Face Detection

As when the user comes in front of the screen, to detect the user the discrete structure which is to be identified is the face. So, to detect the face, we use Haar feature-based cascade classifiers. In haar classifier instead of using intensity values of the pixel, it uses the change in contrast values between adjacent groups of pixels. Then the variance difference between the pixel groups is used to determine the relative light and dark areas in the image. It is a machine learning approach. So, to work well with the algorithm the cascade function is trained from a lot of negative and

positive images. A lot of negative images (images without faces) and positive images (images with faces) are shown to the classifier to train it so it can extract features from it. The purpose of using OpenCV makes it easier as it comes with pre-trained classifiers for face, eyes, smile etc. It comes with a trainer and a detector; we can train it with our own classifier easily for any object detection. If it finds a match it returns Rect (x, y, w, h) implying coordinates for left, top, bottom and right.

4.2.3 Image Masking

Simply the image masked has some of its pixel intensity values set to zero. In the image wherever the pixel intensity value is zero automatically the pixel intensity of the resulting masked image will be set to the background value which is normally zero. Or To define the mask the ROIs for each slice is used. If required, masking can be controlled on slice by slice basis in ROI toolkit. In ROI toolkit, masking operations does not affect a slice without ROI.

4.2.4 Edge Detection

There are various edge detection techniques. We have used Canny Edge detection technique as discussed before for detection of body. To perform this edge detection technique Gaussian filters are used. These filters cut out the noise in a digital image to prevent any false detection by the processor. This does the work of smoothening and reducing the effect of noise on the image for the proper functioning of the processor. With this the intensity gradients of the image are not found out. The edges in the image can point in various directions like horizontal, vertical and in diagonal edges, so this algorithm uses four filter to detect all kind of edges in blurred image. After this process non-maximum suppression is applied to make the edge thin. This suppression results in quite accurate edge pixels in reference to present real edges. Also, some pixels may be caused by the noise, then for such pixels we apply double threshold on them.

4.2.5 Scaling of Attire

Scaling means resizing of the image according to the circumstances. As when the user moves in front of the screen it should change the sizing of the attire and place it on the body accordingly. When the user moves towards the screen the image size should be increased according to user but the actual measurements of the attire should not increase. This is done by scaling method.

4.2.6 Body Parts Tracking:

OpenCV's pose estimation algorithms like OpenPose or AlphaPose were used to track the user's body parts like head, torso, arms. This helped us map the user's body to the virtual clothes

4.2.7 Data Preparation:

We collected a dataset of images of clothes from different angles and sizes. Then preprocessed the images to remove any background noise and standardize the image size and format

4.2.8 Virtual Clothes Overlay:

Computer vision techniques like homography, image warping, and blending were used to overlay the virtual clothes on the user's body in real-time. This created the illusion that the user was wearing the clothes.

4.2.9 User Interface:

A simple user interface was developed using OpenCV's GUI toolkit, allowing users to select different types of clothes and accessories. This made the application easy to use and intuitive.

4.2.10 Real-time Processing:

The algorithms were optimized to achieve real-time processing and rendering of the virtual clothes on the user's body. This enabled the user to see themselves in the virtual clothes in real-time.

4.3 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

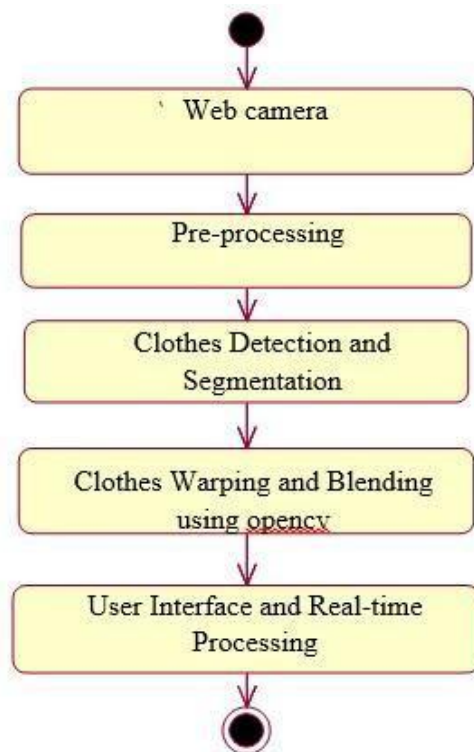


Fig 4.3.1 Activity diagram

Chapter 5

IMPLEMENTATION DESIGN

5.1 Development Setup:

Development setup is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analyzing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

5.2 Dependencies required:

Frontend:

Developed using modern web technologies (HTML5, CSS3, JavaScript), ensuring cross-browser compatibility and responsive design for seamless user experience across devices.

Backend:

Utilizes a robust backend infrastructure powered by cloud computing services (such as AWS or Azure) to handle heavy computational tasks and provide scalability and reliability.

AR Technology:

Integration of AR technology enables real-time rendering of virtual garments onto the user's live video feed, ensuring an accurate representation of how the clothing will look and fit.

Computer Vision:

Advanced computer vision algorithms analyze user body measurements and garment attributes to provide personalized sizing recommendations and ensure a perfect fit.

Database:

Stores user profiles, clothing inventory, and transactional data securely, ensuring data privacy and enabling personalized recommendations based on past purchases and preferences.

Virtual Try-On:

Users can select garments from our extensive catalog and visualize how they look on themselves in real-time using their device's camera.

Accurate Sizing:

Our platform utilizes precise measurements and body scanning technology to recommend the correct size for each garment, reducing the likelihood of returns due to sizing issues.

Customization Options:

Users can customize clothing options such as color, pattern, and style to match their preferences, providing a personalized shopping experience.

Integration with E-commerce Platforms:

Seamless integration with popular e-commerce platforms allows users to directly purchase items they try on, streamlining the buying process.

Feedback Mechanism:

Incorporates a feedback mechanism where users can rate the fit and comfort of the garments they try on, providing valuable insights for both customers and retailers.

Cross-Platform Compatibility:

Supports multiple devices and operating systems, including smartphones, tablets, and desktops, ensuring accessibility for all users.

Data Encryption:

Utilizes industry-standard encryption protocols to secure user data during transmission and storage, safeguarding against unauthorized access.

GDPR Compliance:

Adheres to GDPR guidelines to protect user privacy and ensure transparency in data handling practices.

Anonymization:

Personal data is anonymized whenever possible, and user consent is obtained for data processing activities, respecting user privacy rights.

Intuitive Interface:

A user-friendly interface with intuitive controls and clear instructions ensures a seamless and enjoyable shopping experience.

Fast Loading Times:

Optimized for performance to minimize loading times and provide a smooth and responsive interface.

Accessibility:

Incorporates accessibility features such as text-to-speech and keyboard navigation to accommodate users with disabilities.

Scalable Infrastructure:

Built on a scalable architecture to handle growing user demand and accommodate future expansion.

Regular Updates:

Regular updates and maintenance ensure the platform remains secure, reliable, and up-to-date with the latest technologies and user needs.

Chapter 6 RESULT AND DISCUSSION

6.1 Implementation of virtual trial room for online shopping

The real-time virtual dressing room OpenCV project has garnered widespread acclaim for its success in revolutionizing the retail experience. By seamlessly integrating cutting-edge technology with the traditional process of trying on clothes, this innovative application offers customers a convenient and engaging solution for virtual garment exploration. Through the utilization of OpenCV (Open Source Computer Vision Library), the project achieves a remarkable feat: accurately detecting the user's body and tracking its movements in real-time. This foundational capability sets the stage for a truly immersive experience, where virtual garments are seamlessly overlaid onto the user's body, creating a lifelike simulation of wearing the selected clothing items.

What sets this project apart is not only its technical prowess but also its ability to deliver a user-friendly interface that prioritizes accessibility and ease of use. The intuitive interface empowers customers to effortlessly browse through a diverse array of clothing options, selecting and trying on different garments with just a few clicks or gestures. By eliminating the barriers typically associated with virtual try-on experiences, such as complex navigation or cumbersome controls, the project ensures that users can fully immerse themselves in the virtual dressing room environment without any distractions or frustrations.

Furthermore, the real-time processing and rendering capabilities of the application contribute to its success by providing a seamless and responsive user experience. Leveraging advanced algorithms and optimized computational techniques, the project achieves lightning-fast performance, allowing virtual garments to react dynamically to the user's movements in real-time. This level of responsiveness not only enhances the sense of immersion but also fosters a deeper connection between the user and the virtual dressing room, ultimately resulting in a more engaging and memorable shopping experience.

Beyond its technical achievements, the real-time virtual dressing room OpenCV project has significant implications for the retail industry as a whole. By bridging the gap between online and offline shopping experiences, the application addresses a fundamental challenge faced by retailers: how to replicate the tactile experience of trying on clothes in a physical store in an online environment. In doing so, the project not only enhances customer satisfaction but also drives sales and reduces the likelihood of returns, ultimately benefiting both retailers and consumers alike.

In conclusion, the success of the real-time virtual dressing room OpenCV project is a testament to the transformative power of technology in reshaping the retail landscape. By providing a convenient and engaging way for customers to try on clothes virtually in real-time, the project has set a new standard for online shopping experiences, one that seamlessly integrates the latest advancements in computer vision with the timeless appeal of trying on clothes. As technology continues to evolve, the possibilities for virtual try-on experiences are endless, promising even greater convenience, immersion, and innovation for consumers and retailers alike.

The application was able to detect the user's body and track their body parts in real-time, allowing for a seamless and interactive experience. The virtual clothes were accurately overlaid on the user's body, creating a realistic impression that they were wearing the clothes.

The intuitive and user-friendly interface of the real-time virtual dressing room OpenCV project was meticulously designed to enhance the overall user experience. Through thoughtful interface design and intuitive navigation elements, users are empowered to effortlessly explore the wide range of clothing and accessories available within the virtual environment. The interface is thoughtfully organized, with clear categories and filters that enable users to quickly locate specific items of interest. Whether browsing through tops, bottoms, or accessories, users can easily refine their search and find the perfect garments to try on.

Moreover, the interface facilitates seamless interactions, allowing users to transition seamlessly between different clothing items with minimal effort. Thanks to intuitive controls and responsive design elements, users can explore various combinations and styles with ease, enhancing their overall engagement with the virtual dressing room.

In addition to its intuitive interface, the real-time processing and rendering capabilities of the application play a crucial role in delivering a smooth and seamless user experience. Leveraging advanced algorithms and optimized rendering techniques, the project ensures that virtual garments are rendered quickly and efficiently, with minimal latency or lag. This real-time processing capability not only enhances the sense of immersion but also contributes to a more responsive and engaging user experience.

Furthermore, the efficient rendering of virtual clothes enables users to see accurate representations of how different garments will look and fit on their body in real-time. This level of realism and accuracy is crucial for building trust and confidence among users, ultimately leading to higher levels of engagement and satisfaction with the virtual dressing room experience.

In conclusion, the intuitive interface and efficient processing and rendering capabilities of the real-time virtual dressing room OpenCV project combine to create a seamless and engaging user experience. By prioritizing ease of use and responsiveness, the project empowers users to explore and experiment with different clothing and accessory options with confidence and creativity. As technology continues to advance, the potential for further enhancing the virtual try-on experience is limitless, promising even greater convenience and immersion for consumers in the future.

Chapter 7

CONCLUSION

The rise of online shopping and the desire for customers to fully utilize it while ensuring complete personal satisfaction with their dress purchases has necessitated the development of a system that virtually dresses individuals in their chosen clothing. The proposed system has been developed to work reliably without the need for external light adjusters or any hardware devices, which are commonly used in current algorithms and create limitations on their usage. This proposed system offers a more accessible solution for all people, enabling them to try on different clothes with fewer restrictions, providing greater pleasure for online shoppers.

In conclusion, the Virtual Trial Room for online shopping is a revolutionary technology that has the potential to transform the way people shop for clothes online. The development of this technology using advanced technologies like augmented reality, computer vision, and machine learning has addressed the challenge of not being able to try on clothes before making a purchase, providing an immersive and personalized shopping experience to customers.

The Virtual Trial Room offers several benefits, including the ability to visualize how clothes would look on the customer, the convenience of shopping from home, and the ability to compare different styles and sizes easily. It also helps businesses by reducing the likelihood of returns due to sizing issues and increasing customer satisfaction.

The project's success relies on the simple usage of the system without any additional hardware except a web camera which is already available in Laptop or PC. Additionally, the use of OpenCV in the project provides several advantages, including real-time image processing, cost-effectiveness, and scalability.

In the future, the algorithm can be modified to detect the human silhouette in more challenging environments with variable backgrounds and increased noise levels. This project sets a new standard for online shopping and provides a more engaging and personalized shopping experience to customers.

REFERENCES

1. K.Srinivasan, K.Porkumaran, G.Sai Narayanan, "Intelligent human body tracking, modelling and activity analysis of video surveillance system: A Survey", Journal of convergence in engineering, technology and science, Vol.1, pp.1-8, 2021.
2. Max Mignotte, "Segmentation by Fusion of Histogram based K-Means Clusters in different color space" IEEE Transactions on Image Processing, Vol.17, pp.780-787, 2018.
3. F. Cordier, W. Lee, H. Seo, and N. Magnenat-Thalmann, "From 2D Photos of Yourself to Virtual Try-on Dress on the Web", Springer, pp. 31–46, 2016.
4. D. Protopsaltou, C. Luible, M. Arevalo-Poizat, and N. Magnenat Thalmann, "A body and garment creation method for an internet based virtual fitting room", in Proc. Computer Graphics International 2002 (CGI '02). Springer, pp. 105–122, 2010.
5. F. Cordier, H. Seo, and N. Magnenat-Thalmann, "Made-to-measure technologies for an online clothing store", IEEE Comput. Graph. Appl., vol. 23, no. 1, pp. 38–48, Jan. 2007.
6. K.Srinivasan, K.Porkumaran, G.Sai Narayanan, "Enhanced Background Subtraction Techniques for Monocular Video Applications", International Journal of Image Processing and Applications, Vol. 1, no. 2, pp.87-93, 2005.
7. K.Srinivasan, K.Porkumaran, G.Sai Narayanan, "Skin colour segmentation based 2D and 3D human pose modelling using Discrete Wavelet Transform", Journal of Pattern recognition and image Analysis, Springer, Vol.21, pp.740-753, 2003.

Appendix

A: SAMPLE CODE

```
import tkinter as tk

from tkinter import ttk

import sys
```

```
class Scrollable(ttk.Frame):
```

Make a frame scrollable with scrollbar on the right.

After adding or removing widgets to the scrollable frame,
call the `update()` method to refresh the scrollable area.

```
def __init__(self, frame, width=16):
```

```
    scrollbar = tk.Scrollbar(frame, width=width)
```

```
    scrollbar.pack(side=tk.RIGHT, fill=tk.Y, expand=False)
```

```
    self.canvas = tk.Canvas(frame, bg='white', yscrollcommand=scrollbar.set)
```

```
    self.canvas.pack(side=tk.LEFT, fill=tk.BOTH, expand=True)
```

```
    scrollbar.config(command=self.canvas.yview)
```

```
    self.canvas.bind('<Configure>', self.__
```

fill_canvas)

```
# base class initialization
```

```
tk.Frame.__init__(self, frame)
```

```
# assign this obj (the inner frame) to the windows item of the canvas
```

```
self.windows_item = self.canvas.create_window(
```

```
0, 0, window=self, anchor=tk.NW)
```

```
def __fill_canvas(self, event):
```

```
"Enlarge the windows item to the canvas width"
```

```
canvas_width = event.width
```

```
self.canvas.itemconfig(self.windows_item, width=canvas_width)
```

```
def update(self):
```

```
"Update the canvas and the scrollregion"
```

```
self.update_idletasks()
```

```
self.canvas.config(scrollregion=self.canvas.bbox(self.windows_item))
```

```
class FullScreenApp(object):
```

```
def __init__(self, master, **kwargs):
```

```
self.master = master
```

```
pad = 3
```

```
self._geom = '200x200+0+0'

master.geometry("{0}x{1}+0+0".format(master.winfo_screenwidth() -
pad, master.winfo_screenheight() - pad))

master.bind('<Escape>', self.toggle_geom)
```

```
def toggle_geom(self, event):

geom = self.master.winfo_geometry()

print(geom, self._geom)

self.master.geometry(self._geom)

self._geom = geom
```

```
# set a callback to handle when the window is closed

self.root.wm_title("PhotoBooth")

self.root.wm_protocol("WM_DELETE_WINDOW", self.onClose)
```

```
def videoLoop(self):

try:

while not self.stopEvent.is_set():

self.frame = self.vs.read()

self.frame = imutils.resize(self.frame, width=300)

image = cv2.cvtColor(self.frame, cv2.COLOR_BGR2RGB)

image = Image.fromarray(image)

image = ImageTk.PhotoImage(image)

if self.panel is None:
```

```

self.panel = tk.Label(image=image)

self.panel.image = image

self.panel.pack(side="left", padx=10, pady=10)

else:

self.panel.configure(image=image)

self.panel.image = image

except RuntimeError as e:

print("[INFO] caught a RuntimeError")


def takeSnapshot(self):

# grab the current timestamp and use it to construct the

# output path

ts = datetime.datetime.now()

filename = "{}.jpg".format(ts.strftime("%Y-%m-%d_%H-%M-%S"))

p = os.path.sep.join((self.outputPath, filename))


# save the file

cv2.imwrite(p, self.frame.copy())

print("[INFO] saved {}".format(filename))


def onClose(self):

# set the stop event, cleanup the camera, and allow the rest of

# the quit process to continue

print("[INFO] closing...")

```

```
self.stopEvent.set()
```

```
self.vs.stop()
```

```
self.root.quit()
```

Main

```
from flask import Flask, render_template, Response, redirect, request
```

```
from camera import VideoCamera
```

```
import os
```

```
import
```

```
sys
```

```
app = Flask(__name__)
```

```
CART=[]
```

```
@app.route('/checkOut', methods=['POST', 'GET'])
```

```
def checkOut():
```

```
    return render_template('checkout.html')
```

```
@app.route('/tryon/<file_path>', methods = ['POST', 'GET'])
```

```
def tryon(file_path):
```

```
    file_path = file_path.replace(',', '/')
```

```
    os.system('python tryOn.py ' + file_path)
```

```
    return redirect('http://127.0.0.1:5000/', code=302, Response=None)
```

```
    #return Response(gen(VideoCamera()), mimetype='multipart/x-mixed-replace;  
    boundary=frame')
```

```

@app.route('/tryall',methods = ['POST', 'GET'])

def tryall():

    print("YESSS")

    if request.method == 'POST':

        cart = request.form['mydata'].replace(',','')

        '/'> print("tryall cart type= ", type(cart))

        print("tryall cart= ", cart)

        os.system('python test.py ' + cart)

        return redirect('http://127.0.0.1:5000/',code=302, Response=None)

```

```

@app.route('/')

def index():

    return render_template('index.html')

```

```

def gen(camera):

    while True:

        frame = camera.get_frame()

        #print("frame= ", frame)

        yield (b'--frame\r\n'

               b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n')

```

```

@app.route("/cart/<file_path>",methods = ['POST', 'GET'])

def cart(file_path):

    global CART

```

```

file_path = file_path.replace(',', '/')
print("ADDED", file_path)
CART.append(file_path)
return render_template("checkout.html")

@app.route('/video_feed')
def video_feed():
    return Response(gen(VideoCamera()),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

if __name__ == '__main__':
    app.run(debug=True)

```

Try on

```

from tkinter import *
from PIL import Image
from PIL import ImageTk
import cv2, threading, os, time
from threading import Thread
from os import listdir
from os.path import isfile, join
import sys
import dlib
from imutils import face_utils, rotate_bound
import math

```

```

def put_sprite(num):
    global SPRITES, BTNS
    SPRITES[num] = (1 - SPRITES[num])
    # if SPRITES[num]:
    #
    BTNS[num].config(relief=SUNKEN) #
    else:
    #   BTNS[num].config(relief=RAISED)

def draw_sprite(frame, sprite, x_offset, y_offset):
    print("sprite>>>>>>", sprite.shape, "type=", type(sprite.shape))
    (h,w) = (sprite.shape[0], sprite.shape[1])
    (imgH,imgW) = (frame.shape[0], frame.shape[1])

    if y_offset+h >= imgH:
        sprite = sprite[0:imgH-y_offset,:,:]

    if x_offset+w >= imgW:
        sprite = sprite[:,0:imgW-x_offset,:]

    if x_offset < 0:
        sprite = sprite[:,abs(x_offset):,:,:]
    w = sprite.shape[1]

```

```
x_offset = 0
```

```
for c in range(3):
```

```
try:
```

```
frame[y_offset:y_offset+h, x_offset:x_offset+w, c] = \
```

```
sprite[:, :, c] * (sprite[:, :, 3]/255.0) + frame[y_offset:y_offset+h, x_offset:x_offset+w, c] *  
(1.0 - sprite[:, :, 3]/255.0)
```

```
except Exception as e:
```

```
print(e)
```

```
pass
```

```
return frame
```

```
def adjust_sprite2head(sprite, head_width, head_ypos, ontop = True):
```

```
(h_sprite, w_sprite) = (sprite.shape[0], sprite.shape[1])
```

```
factor = 1.0*head_width/w_sprite
```

```
sprite = cv2.resize(sprite, (0,0), fx=factor, fy=factor)
```

```
(h_sprite, w_sprite) = (sprite.shape[0], sprite.shape[1])
```

```
y_orig = head_ypos-h_sprite if ontop else head_ypos
```

```
if (y_orig < 0):
```

```
sprite = sprite[abs(y_orig)::, :, :]
```

```
y_orig = 0
```

```
return (sprite, y_orig)
```

```
def apply_sprite(image, path2sprite, w, x, y, angle, ontop = True):
```

```

sprite = cv2.imread(path2sprite,-1)

sprite = rotate_bound(sprite, angle)

(sprite, y_final) = adjust_sprite2head(sprite, w, y, ontop)

image = draw_sprite(image, sprite, x, y_final)

```

```

def calculate_inclination(point1, point2):

x1,x2,y1,y2 = point1[0], point2[0], point1[1], point2[1]

incl = 180/math.pi*math.atan((float(y2-y1))/(x2-x1))

return incl

```

```

def calculate_boundingbox(list_coordinates):

x = min(list_coordinates[:,0])

y = min(list_coordinates[:,1])

w = max(list_coordinates[:,0]) -

x h = max(list_coordinates[:,1]) -

y return (x,y,w,h)

```

```

def detectUpperBody(image):

cascadePath = 'data/haarcascade_upperbody.xml'

result = image.copy()

imageGray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

cascade = cv2.CascadeClassifier(cascadePath)

Rect = cascade.detectMultiScale(imageGray, scaleFactor=1.1, minNeighbors=1,

minSize=(1,1))

if len(Rect) <= 0:

```

```
return False
```

```
else:
```

```
return Rect
```

```
def get_face_boundingBox(points, face_part):
```

```
if face_part == 1:
```

```
(x,y,w,h) = calculate_boundingBox(points[17:22])
```

```
elif face_part == 2:
```

```
(x,y,w,h) = calculate_boundingBox(points[22:27])
```

```
elif face_part == 3:
```

```
(x,y,w,h) = calculate_boundingBox(points[36:42])
```

```
elif face_part == 4:
```

```
(x,y,w,h) = calculate_boundingBox(points[42:48])
```

```
elif face_part == 5:
```

```
(x,y,w,h) = calculate_boundingBox(points[29:36])
```

```
elif face_part == 6:
```

```
(x,y,w,h) = calculate_boundingBox(points[0:17])
```

```
elif face_part == 7:
```

```
# (x,y,w,h) = calculate_boundingBox(points[48:68]) #mouth
```

```
(x,y,w,h) = calculate_boundingBox(points[1:5])
```

```
elif face_part == 8:
```

```
(x,y,w,h) = calculate_boundingBox(points[12:16])
```

```
return (x,y,w,h)
```

```
image_path = "
```

```
def add_sprite(img):
```

```
global image_path
```

```
image_path = img
```

```
print("rsplit of imgpath>>>>>>",img.rsplit('/',1))
```

```
print(">>>>>>>>>>",int(img.rsplit('/',1)[0][-1]))
```

```
put_sprite(int(img.rsplit('/',1)[0][-1])) #here it will return the integer value from 1 to 6,  
which will denotes that which catagory is the apparel belongs to i.e either frock or  
tops or etc.
```

```
#Principal Loop where openCV (magic) occurs
```

```
# Face detection starts from here
```

```
def cvloop(run_event):
```

```
global panelA
```

```
global SPRITES
```

```
global image_path
```

```
i = 0
```

```
video_capture = cv2.VideoCapture(0) #read from webcam
```

```
(x,y,w,h) = (0,0,10,10) #whatever initial values
```

```
#Filters path
```

```
detector = dlib.get_frontal_face_detector()
```

```
model = "data/shape_predictor_68_face_landmarks.dat"
```

```
predictor = dlib.shape_predictor(model) # link to model:  
http://dlib.net/files/shape\_predictor\_68\_face\_landmarks.dat.bz2
```

```
while run_event.is_set():
```

```
ret, image = video_capture.read()
```

```
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY) #converting the colour image  
to gray
```

```
faces = detector(gray, 0)
```

```
for face in faces:
```

```
(x,y,w,h) = (face.left(), face.top(), face.width(), face.height()) #reading the  
coordinates in sequence
```

```
shape = predictor(gray, face)
```

```
shape = face_utils.shape_to_np(shape)
```

```
incl = calculate_inclination(shape[17], shape[26]) #inclination based on eyebrows
```

```
# condition to see if mouth is open
```

```
is_mouth_open = (shape[66][1] - shape[62][1]) >= 10 #y coordiantes of landmark points  
of lips
```

```
if SPRITES[0]:
```

```
apply_sprite(image,image_path,w,x,y+40, incl, ontop = True)
```

```
if SPRITES[3]:#Tiara
```

```
    apply_sprite(image, image_path,w+45,x-20,y+15, incl, ontop = True)
```

```
#Necklaces
```

```
if SPRITES[1]:
```

```
    (x1,y1,w1,h1) = get_face_boundingbox(shape, 6)
```

```
    apply_sprite(image, image_path,w1,x1,y1+110, incl, ontop = False)
```

```
#Goggles
```

```
if SPRITES[6]:
```

```
    (x3,y3,_,h3) = get_face_boundingbox(shape, 1)
```

```
    apply_sprite(image, image_path,w,x,y3-10, incl, ontop = False)
```

```
#Earrings
```

```
(x0,y0,w0,h0) = get_face_boundingbox(shape, 6) #bound box of mouth
```

```
if SPRITES[2]:
```

```
    (x3,y3,w3,h3) = get_face_boundingbox(shape, 7) #nose
```

```
    apply_sprite(image, image_path,w3,x3-40,y3+30, incl,ontop=False)
```

```
    (x3,y3,w3,h3) = get_face_boundingbox(shape, 8) #nose
```

```
    apply_sprite(image, image_path,w3,x3+30,y3+75, incl)
```

```
# if SPRITES[5]:
```

```
#     apply_sprite(image,image_path,w,x,y, incl, ontop = True)
```

```
#Frocks
```

```
if SPRITES[5]:
```

```
(x1,y1,w1,h1) = get_face_boundingBox(shape, 8)
```

```
apply_sprite(image, image_path,w1+590,x1-300,y1+70, incl, ontop = False)
```

```
#Tops
```

```
if SPRITES[4]:
```

```
# (x,y,w,h) = (0,0,10,10)
```

```
# apply_sprite2feature(image, IMAGES[7][ACTIVE_IMAGES[7]], fullbody,  
w//4, 2*h//3, h//2, True, w//2, x, y, w, h)
```

```
(x1,y1,w1,h1) = get_face_boundingBox(shape, 8)
```

```
apply_sprite(image, image_path,w1+300,x1-200,y1+40, incl, ontop = False)
```

```
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
```

```
image = Image.fromarray(image)
```

```
image = ImageTk.PhotoImage(image)
```

```
#image = ImageTk.PhotoImage(image.resize((500,500))) #this is for photo mode
```

```
panelA.configure(image=image)
```

```
panelA.image = image
```

```
video_capture.release()
```

```
# Initialize GUI object
```

```
root = Tk()
```

```
root.title("Virtual trial room")
```

```
this_dir = os.path.dirname(os.path.realpath(__file__))
```

btn1 = None

```
def try_on(image_path):
```

```
btn1 = Button(root, text="Try it ON", command = lambda:add_sprite(image_path))
```

```
btn1.pack(side="top", fill="both", expand="no", padx="5", pady="5")
```

```
panelA = Label(root)
```

```
panelA.pack( padx=10,
```

pady=10)

```
SPRITES = [0,0,0,0,0,0,0]
```

BTNS = [btn1]

```
print(">>>>>>>>>>>>>",sys.argv[1])
```

```
try_on(sys.argv[1]) #here sys.argv will return the path of apparels.
```

```
run_event = threading.Event()
```

```
run_event.set()
```

```
action = Thread(target=cvloop, args=(run_event,))
```

```
action.setDaemon(True)
```

```
action.start()
```

```
def terminate():
```

global root, run_event, action

```
run_event.clear()
```

```
time.sleep(1)
```

```
root.destroy()
```

```
root.protocol("WM_DELETE_WINDOW", terminate)
```

```
root.mainloop()
```

Test

```
from tkinter_scroll import *
```

```
from tkinter import *
```

```
from PIL import Image
```

```
from PIL import ImageTk
```

```
import cv2, threading, os, time
```

```
from threading import Thread
```

```
from os import listdir
```

```
from os.path import isfile, join
```

```
import dlib
```

```
from imutils import face_utils, rotate_bound
```

```
import math
```

```
import sys
```

```
ACTIVE_IMAGES=[0 for i in range(100)]
```

```
def put_sprite(num, k):
```

```
global SPRITES, BTNS
```

```
SPRITES[num] = (1 - SPRITES[num])
```

```
if SPRITES[num]:
```

```
ACTIVE_IMAGES[num] = k
```

```
BTNS[num].config(relief=SUNKEN) #buttons to click
```

```
else:
```

```
BTNS[num].config(relief=RAISED)
```

```
def draw_sprite(frame, sprite, x_offset, y_offset):
```

```
(h,w) = (sprite.shape[0], sprite.shape[1])
```

```
(imgH,imgW) = (frame.shape[0], frame.shape[1])
```

```
if y_offset+h >= imgH:
```

```
    sprite = sprite[0:imgH-y_offset,:,:]
```

```
if x_offset+w >= imgW:
```

```
    sprite = sprite[:,0:imgW-x_offset,:]
```

```
if x_offset < 0:
```

```
    sprite = sprite[:,abs(x_offset):,:]
```

```
w = sprite.shape[1]
```

```
x_offset = 0
```

```
for c in range(3):
```

```
    frame[y_offset:y_offset+h, x_offset:x_offset+w, c] = \
```

```
        sprite[:,c] * (sprite[:,3]/255.0) + frame[y_offset:y_offset+h, x_offset:x_offset+w, c] *  
        (1.0 - sprite[:,3]/255.0)
```

```
return frame
```

```
def adjust_sprite2head(sprite, head_width, head_ypos, ontop = True):
```

```
(h_sprite,w_sprite) = (sprite.shape[0], sprite.shape[1])
```

```
factor = 1.0*head_width/w_sprite
```

```
sprite = cv2.resize(sprite, (0,0), fx=factor, fy=factor)
```

```
(h_sprite,w_sprite) = (sprite.shape[0], sprite.shape[1])
```

```
y_orig = head_ypos-h_sprite if ontop else head_ypos
```

```
if (y_orig < 0):
```

```
    sprite = sprite[abs(y_orig)::,,:]
```

```
    y_orig = 0
```

```
return (sprite, y_orig)
```

```
def apply_sprite2feature(image, sprite_path, haar_filter, x_offset, y_offset,  
y_offset_image, adjust2feature, desired_width, x, y, w, h):
```

```
    sprite = cv2.imread(sprite_path,-1)
```

```
    (h_sprite,w_sprite) = (sprite.shape[0],
```

```
    sprite.shape[1])
```

```
    xpos = x + x_offset
```

```
    ypos = y + y_offset
```

```
    factor = 1.0*desired_width/w_sprite
```

```
    sub_img = image[y + y_offset_image:y+h,x:x+w,:]
```

```
    feature = apply_Haar_filter(sub_img, haar_filter, 1.3 , 10, 10)
```

```
    if len(feature)!=0:
```

```
        xpos, ypos = x, y + feature[0,1] #adjust only to feature in y axis (eyes)
```

```
    if adjust2feature:
```

```
size_mustache = 1.2 #how many times bigger than mouth
```

```
factor = 1.0*(feature[0,2]*size_mustache)/w_sprite
```

```
xpos = x + feature[0,0] - int(feature[0,2]*(size_mustache-1)//2) #centered respect to width
```

```
ypos = y + y_offset_image + feature[0,1] - int(h_sprite*factor) #right on top
```

```
sprite = cv2.resize(sprite, (0,0), fx=factor, fy=factor)
```

```
image = draw_sprite(image, sprite, xpos, ypos)
```

```
def apply_sprite(image, path2sprite, w, x, y, angle, ontop = True):
```

```
    sprite = cv2.imread(path2sprite, -1)
```

```
    sprite = rotate_bound(sprite, angle)
```

```
    (sprite, y_final) = adjust_sprite2head(sprite, w, y, ontop)
```

```
    image = draw_sprite(image, sprite, x, y_final)
```

```
def calculate_inclination(point1, point2):
```

```
    x1, x2, y1, y2 = point1[0], point2[0], point1[1], point2[1]
```

```
    incl = 180/math.pi*math.atan((float(y2-y1))/(x2-x1))
```

```
    return incl
```

```
def calculate_boundingbox(list_coordinates):
```

```
    x = min(list_coordinates[:,0])
```

```
    y = min(list_coordinates[:,1])
```

```
    w = max(list_coordinates[:,0]) -
```

```
    x h = max(list_coordinates[:,1]) -
```

```
    y return (x,y,w,h)
```

```
def apply_Haar_filter(img, haar_cascade,scaleFact = 1.05, minNeigh = 3, minSizeW  
= 30):
```

```
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

```
    features = haar_cascade.detectMultiScale(  
    gray,
```

```
    scaleFactor=scaleFact,
```

```
    minNeighbors=minNeigh,
```

```
    minSize=(minSizeW, minSizeW),
```

```
    flags=cv2.CASCADE_SCALE_IMAGE
```

```
    )
```

```
    )
```

```
    return features
```

```
def get_face_boundingBox(points, face_part):
```

```
    if face_part == 1:
```

```
        (x,y,w,h) = calculate_boundingBox(points[17:22])
```

```
    elif face_part == 2:
```

```
        (x,y,w,h) = calculate_boundingBox(points[22:27])
```

```
    elif face_part == 3:
```

```
        (x,y,w,h) = calculate_boundingBox(points[36:42])
```

```
    elif face_part == 4:
```

```
        (x,y,w,h) = calculate_boundingBox(points[42:48])
```

```
    elif face_part == 5:
```

```
        (x,y,w,h) = calculate_boundingBox(points[29:36])
```

```
elif face_part == 6:

(x,y,w,h) = calculate_boundbox(points[1:17])

elif face_part == 7:

(x,y,w,h) = calculate_boundbox(points[0:6])

elif face_part == 8:

(x,y,w,h) = calculate_boundbox(points[11:17])

return (x,y,w,h)
```

```
def cvloop(run_event):

global ctr_mid

global SPRITES

i = 0

video_capture = cv2.VideoCapture(0)

video_capture.set(3,2048)

video_capture.set(4,2048)

(x,y,w,h) = (0,0,10,10)

detector = dlib.get_frontal_face_detector()

fullbody = cv2.CascadeClassifier('data/haarcascade_fullbody.xml')

model = "data/shape_predictor_68_face_landmarks.dat"

predictor = dlib.shape_predictor(model) # link to model:
http://dlib.net/files/shape_predictor_68_face_landmarks.dat.bz2

while run_event.is_set():

ret, image = video_capture.read()

gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

faces = detector(gray, 0)
```

for face in faces:

(x,y,w,h) = (face.left(), face.top(), face.width(), face.height())

shape = predictor(gray, face)

shape = face_utils.shape_to_np(shape)

incl = calculate_inclination(shape[17], shape[26])

is_mouth_open = (shape[66][1] - shape[62][1]) >= 10

if SPRITES[3]:#Tiara

apply_sprite(image, IMAGES[3][ACTIVE_IMAGES[3]],w+45,x-20,y+20, incl, ontop = True)

#Necklaces

if SPRITES[1]:

(x1,y1,w1,h1) = get_face_boundingbox(shape, 6)

apply_sprite(image, IMAGES[1][ACTIVE_IMAGES[1]],w1,x1,y1+150, incl, ontop = False)

#Goggles

if SPRITES[6]:

(x3,y3,_,h3) = get_face_boundingbox(shape, 1)

apply_sprite(image, IMAGES[6][ACTIVE_IMAGES[6]],w,x,y3-10, incl, ontop = False)

#Earrings

(x0,y0,w0,h0) = get_face_boundingbox(shape, 6) #bound box of mouth

if SPRITES[2]:

```

(x3,y3,w3,h3) = get_face_boundingBox(shape, 7) #nose

apply_sprite(image, IMAGES[2][ACTIVE_IMAGES[2]],w3,x3-40,y3+30,
incl,ontop=False)

(x3,y3,w3,h3) = get_face_boundingBox(shape, 8) #nose

apply_sprite(image, IMAGES[2][ACTIVE_IMAGES[2]],w3,x3+40,y3+75, incl)


# if SPRITES[5]:

# apply_sprite(image, IMAGES[5][ACTIVE_IMAGES[5]],w,x,y, incl, ontop =
True)


#Frocks

if SPRITES[5]:

(x1,y1,w1,h1) = get_face_boundingBox(shape, 8)

apply_sprite(image, IMAGES[5][ACTIVE_IMAGES[5]],w1+580,x1-350,y1+80, incl,
ontop = False)


#Tops

if SPRITES[4]:

# (x,y,w,h) = (0,0,10,10)

# apply_sprite2feature(image, IMAGES[7][ACTIVE_IMAGES[7]], fullbody,
w//4, 2*h//3, h//2, True, w//2, x, y, w, h)

(x1,y1,w1,h1) = get_face_boundingBox(shape, 8)

apply_sprite(image, IMAGES[4][ACTIVE_IMAGES[4]],w1+350,x1-230,y1+100, incl,
ontop = False)


image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

```

```
image = Image.fromarray(image)
```

```
image = ImageTk.PhotoImage(image)
```

```
ctr_mid.configure(image=image)
```

```
ctr_mid.image = image
```

```
video_capture.release()
```

```
root = Tk()
```

```
root.title('Virtual Dressing room')
```

```
app=FullScreenApp(root)
```

```
top_frame = Frame(root, bg='#077bd4', width=50, height=50, pady=3)
```

```
center = Frame(root, bg='white', width=50, height=40, padx=3, pady=3)
```

```
btm_frame = Frame(root, bg='#077bd4', width=50, height=50, pady=3)
```

```
root.grid_rowconfigure(1, weight=1)
```

```
root.grid_columnconfigure(0, weight=1)
```

```
top_frame.grid(row=0, sticky="ew")
```

```
center.grid(row=1, sticky="nsew")
```

```
btm_frame.grid(row=4,
```

```
sticky="ew")
```

```
logo = ImageTk.PhotoImage(Image.open('logo1.png').resize((120,60)))
```

```
model_label = Label(top_frame,image=logo)
```

```
model_label.grid(row=0, columnspan=3)
```

```
center.grid_rowconfigure(0, weight=1)
```

```
center.grid_columnconfigure(1, weight=1)
```

```
ctr_left = Label(center,bg='white', width=50, height=190)
```

```
ctr_mid = Label(center,bg='white',width=100, height=160, padx=0, pady=0)
```

```
ctr_left.grid(row=0, column=0, sticky="ns")
```

```
ctr_mid.grid(row=0, column=1, sticky="nsew")
```

```
scrollable_body = Scrollable(ctr_left, width=15)
```

```
SPRITES=[0 for i in range(10)] # [0,0,0,0,0,0,0,0,0,0]
```

```
BTNS=[]
```

```
IMAGES = {i:[] for i in range(10)} #Images= {0: [], 1: [], 2: [], 3: [], 4: [], 5: [], 6: [], 7: [],  
8: [], 9: []}
```

```
PHOTOS = {i:[] for i in range(10)} #PHOTOS= {0: [], 1: [], 2: [], 3: [], 4: [], 5: [], 6: [], 7:  
[], 8: [], 9: []}
```

```
print("Images= ", IMAGES)
```

```
print("PHOTOS= ", PHOTOS)
```

```
print("sys.argv[1:] =", sys.argv[1:])
```

```
for img in sys.argv[1:]:    #it contains all selected images path that are chosen from  
gui, eg-['static/images/Necklace1/1.png', 'static/images/Tops4/12.png']
```

```

IMAGES[int(img.rsplit('/',1)[0][-1]).append(img) #it will append the image path to
desired categ. say 1,2,etc.

image=ImageTk.PhotoImage(Image.open(img).resize((150,100))

) PHOTOS[int(img.rsplit('/',1)[0][-1]).append(image)

print("Images= ", IMAGES)

print("PHOTOS= ", PHOTOS)

for index in range(9):

if len(PHOTOS[index]) > 0:

for k,photo in enumerate(PHOTOS[index]):

#print("text on btn= ",IMAGES[index][k].rsplit('/',1)) #['static/images/Necklace1',
'1.png']

btn= Button(scrollable_body, highlightbackground='white',
text=IMAGES[index][k].rsplit('/',1)[1].replace('.png','')[:-1],bg='white', image=photo,
command = lambda index=index, k=k: put_sprite(index,k), compound=LEFT,
width='300', height='200')

btn.pack(side="top", fill="both", expand="no", padx="5", pady="5")

BTNS.append(btn)

scrollable_body.update()


run_event = threading.Event()

run_event.set()

action = Thread(target=cvloop, args=(run_event,))

action.setDaemon(True)

action.start()


def terminate():

```

global root, run_event, action

run_event.clear()

time.sleep(1)

root.destroy()

root.protocol("WM_DELETE_WINDOW", terminate)

root.mainloop()

B: Screenshots

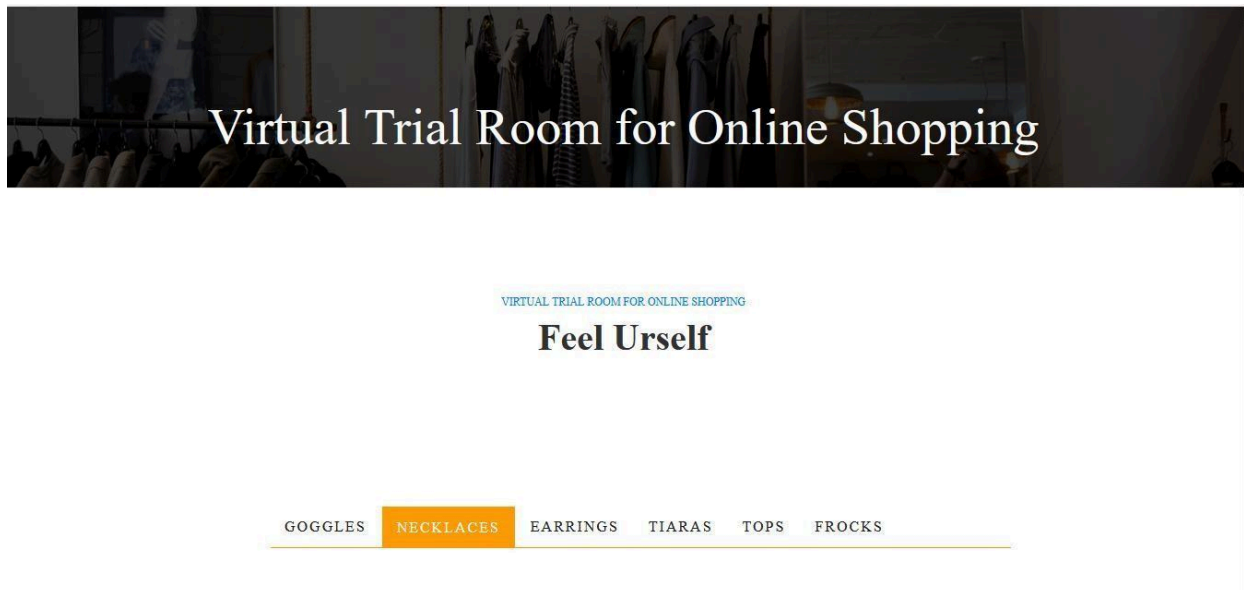


Fig B.1 Home page of our application

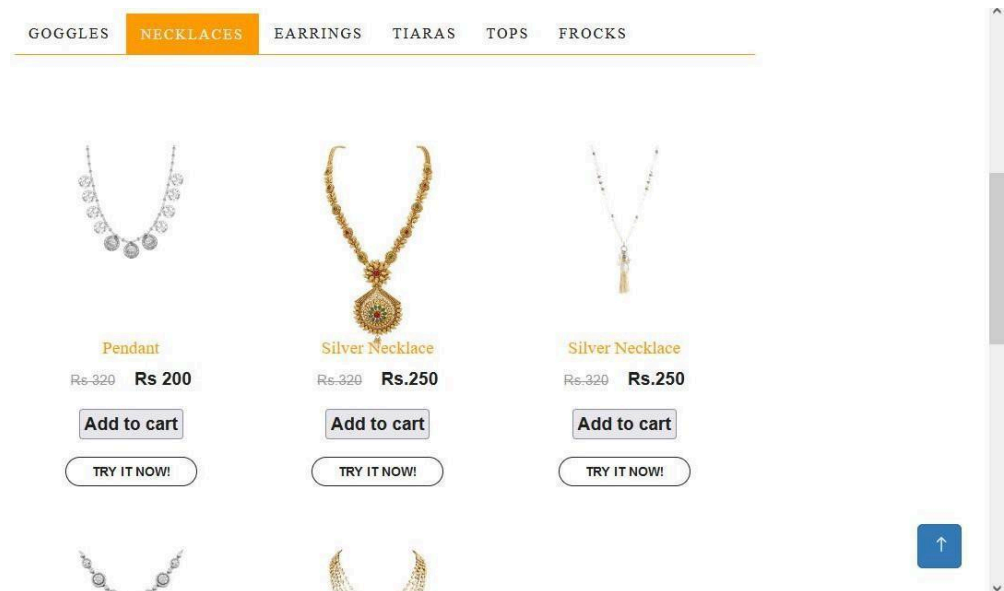


Fig B.2 Necklaces products in the application

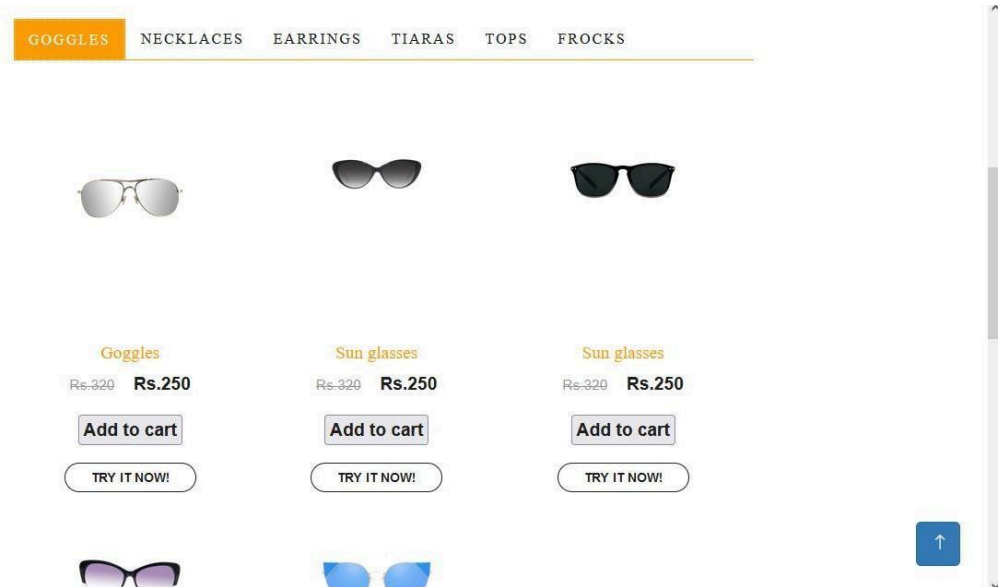


Fig B.3 Sunglasses products in the application

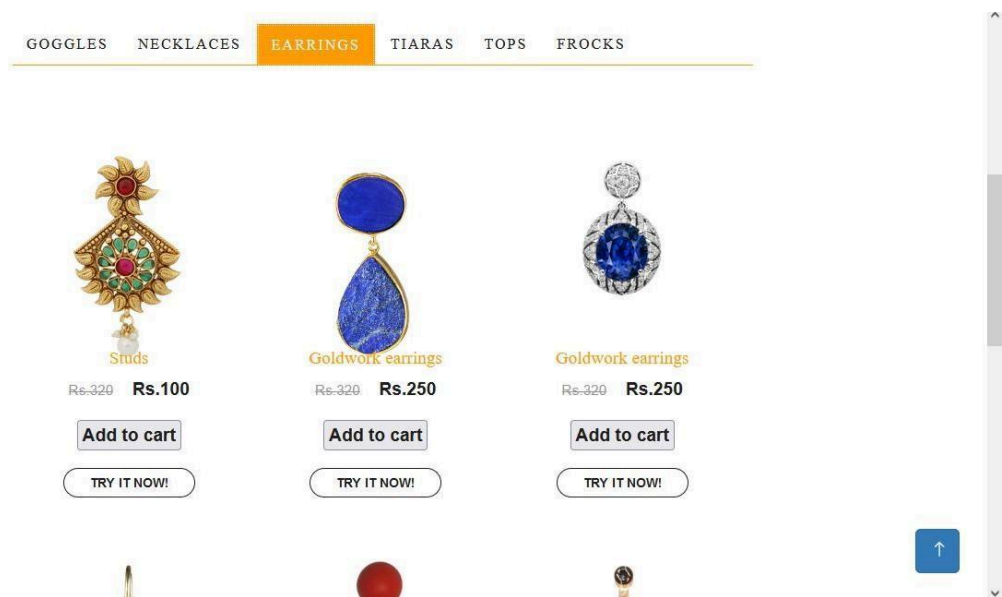


Fig B.4 Accessories products in the application

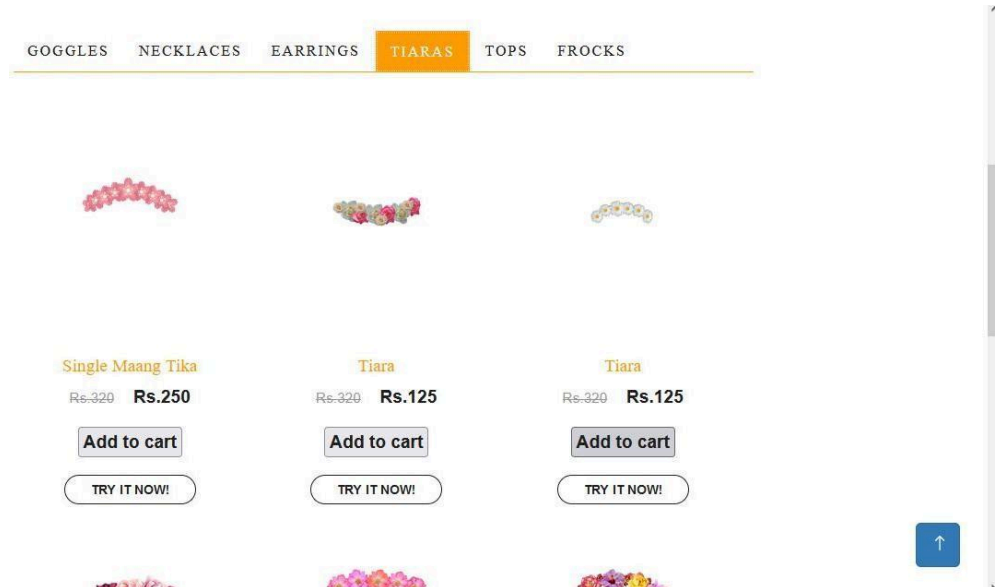


Fig B.5 Hair bands products of our application

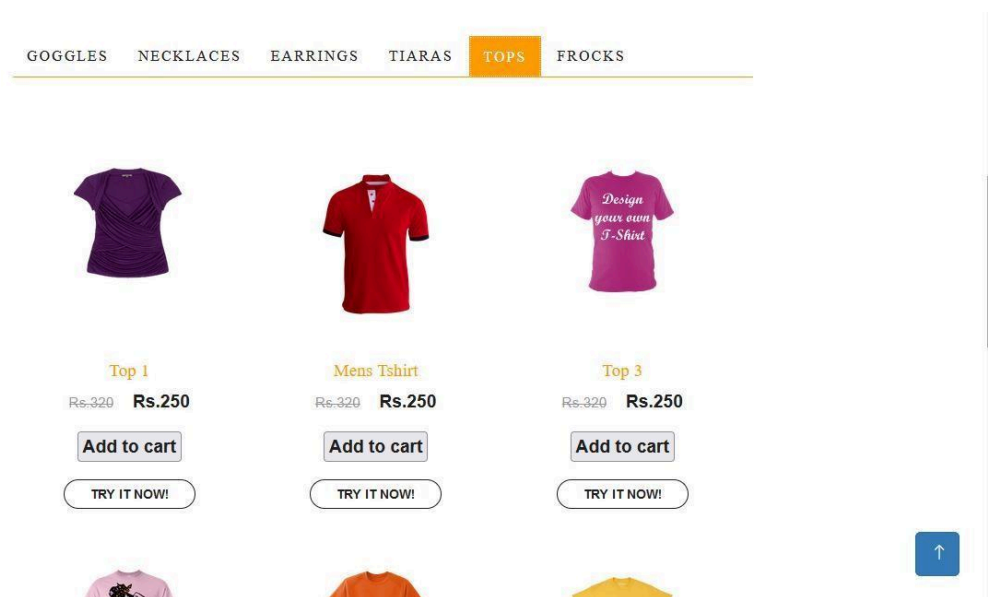


Fig B.6 T shirts products of our application

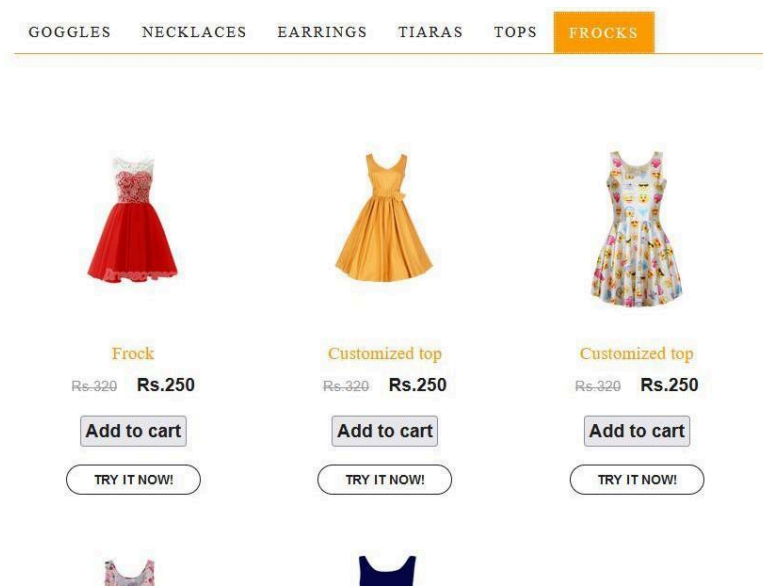


Fig B.7 Skirts products of our application

C RESEARCH PAPER

Virtual Trial Room for Online Shopping

Kavin AM
Department of Computer Science and
Engineering,
Sathyabama Institute of Science and
Technology, Chennai, India
Kavinsai123@gmail.com

Jithesh T
Department of Computer Science and
Engineering,
Sathyabama Institute of Science and
Technology, Chennai, India
jithkrip@gmail.com

G.kalaiaarasi
Department of Computer Science and
Engineering,
Sathyabama Institute of Science and
Technology, Chennai, India
kalaiaarasi.cse@sathyabama.ac.in

M. Selvi
Department of Computer Science and
Engineering,
Sathyabama Institute of Science and
Technology,
Chennai, India
selvi.cse@sathyabama.ac.in

R. Yogitha
Department of Computer Science and
Engineering,
Sathyabama Institute of Science and
Technology,
Chennai, India
yogitha.cse@sathyabama.ac.in

T.N. Suresh babu
Department of Electronics and
Communication Engineering,
Adhiparasakthi Engineering College,
Melmaruvathur, India
sureshbabutns@gmail.com

ABSTRACT- *Virtual Trial Room – An Innovative Solution*
It provides a customized and immersive shopping experience those who save online have to attempt. With improvement Popularity of e-commerce, consumer base the problem is that you cannot get dressed in advance it is smooth to shop for Allows builders to test with this problem simply Clothing within the virtual surroundings. It normally takes a lot of time to try on things in a store. However, in situations like Online Shopping, it might not even be able to try on clothing. The popularity of online purchasing has greatly increased nowadays especially after pandemic breakdowns. When buying things like clothing, it's important to understand how the clothing will look on the buyer. This is the main cause of the decline in online clothing sales. So, a virtual trial room for online shopping would be a tremendous luxury for online retailers that could offer a large selection to users. This would allow people to see how clothes fit them individually. This would be a fantastic tool for Online shopping marketers to expand their market. By establishing a virtual trial room setting, we hope to improve accessibility and time efficiency for trying on clothing. In this project, we provide an Open-CV-based virtual trial room application for online shopping. The Virtual trial Room is an innovative solution that offers a personalized and immersive shopping experience to online shoppers. With the increasing popularity of e-commerce, customers face the challenge of not being able to try on clothes before purchasing them. The Virtual trial Room addresses this issue by allowing customers to virtually try on clothes in a digital environment

KEYWORDS: *Virtual trial room, online shopping*

I INTRODUCTION

"Augmented Reality Based Online Shopping Platform" is the following generation platform. A technology so that it will soon be used in lots of extraordinary areas of enterprise and they're disbursed in the course of the body. We use the magnification ratio a tool that humans can see Simulating them in real time with the want to wear one-of-a-kind costumes It is a web application that may be accessed physically Anywhere in the international, all you want is a browser and net. Augmented reality is interactive evidence in the everyday world, people inhabit bodily realms sometimes growing through the embodiment of practical know-how the countless sensory systems consist of visible, auditory, tactile, and others. This is a large development some of the collaborating e-trade companies. Attracting clients to

business and expertise their desires. Its miles A noticeably visual, interactive manner to stop virtual Making statistics about the physical environment Customers, clearing doubts and growing business outcomes E-trade sites.

The project of the venture is to offer the world with super gain he tries without clothes carrying baggage with the frame. We make Augmented Reality based apps Clothing need to be seen at the body Uses approximately similar to a web software. This will give the consumer freedom to strive clothes how many garments are made at that vicinity should be examined at that location the period of time.

In this strategy we use augmented reality that is on line shopping Surface and item detection method Detection is used. When the floor is detected and Ingredients are detected with the aid of analytical system Finds surface/item dimensions Content I helped layout the content material in real international surroundings.

II LITERATURE SURVEY

K. Srinivasan et. al. [1] describe human frame smart trialing, power modeling, and evaluation. CCTV System: Overview The latest technology and market traits call for the development of answers. Video/Camera and Analytics Systems. This document discussed video systems architecture, goals, and related evaluation techniques. It indicates intelligent video and analytics systems can play a crucial function. In numerous packages like country-wide security, crime prevention, visitor manipulation, twist of fate prediction and detection, patient monitoring, areas of examination of elders at home and children, and so on. Are highlighted on the stop. Smart video makes a speciality of what is needed to achieve dreams in systems and analysis. Max Mignotte [2] discussed segmentation by fusion of histogram-based K-means clusters in different color spaces. It gives a brand new, simple, and effective segmentation-primarily-based technique. The motive of the bus is to enroll in a couple of fragment tables. Through the easy trade of models, extra certainty can, in the end, be executed exactly as a result. The numerous label fields can be merged into our applications and are represented with the aid of equal easy (k-mode-based) parameters. Method of expressing the entered image in distinctive color areas. In our collection, the intention of the approach is to hyperlink these section tables to the very last pages. A system that makes use of neighborhood maps of sophisticated contexts as input information before the road and earlier than every of

Partitioning this incorporated framework is simple to implement, speedy, and flexible. Suitable for use in diverse computer vision programs (e.g., movement management, diagnosis, and remedy), it has been used successfully at the University of Berkeley. The database photo experiments described in this newsletter make it clear that access to fashionable treatments is feasible. Methods have lately been proposed within the literature. F. Cartier et. al. [3] described a holistic approach to vitamins and nutrition for human beings. Using Pages. Enter easy snap shots or bodily measurements. No one in the area is then suitable through digital internet. This permits users to look at the primary technique, which uses pre-calculations. Shared database for growing custom-scaled body and garb animation application text. F. Cartier et. al. [4] The optimized era for online apparel and the Internet is a stunning medium to sell clothes. Some are current projects from organizations like Nordstrom, Macy's, and Lands' End Order and purchase online. We gift an internet application that provides very effective information and manipulation. Fabrics that facilitate layout, sample popularity, and inspection. We apply 3D image technology to help create and simulate virtual stores. D. Protopsaltou et. al. [5] describe a cutting-edge augmented reality "virtual dressing room" that revolutionizes the online clothing shopping experience. Leveraging a single Microsoft Kinect V2 sensor for 3D body measurements, the system creates personalized user models, ensuring a more accurate and immersive virtual fitting process. The implementation employs the Unity3D game engine to achieve real-time overlay of 3D clothes on users, featuring dynamic physics simulations synchronized with user movements. The system's efficiency is highlighted by a performance evaluation showcasing a minimal measurement error of less than 10%, reinforcing its viability for practical and engaging virtual dressing experiences. K. Srinivasan et. al. [6] proposed a growing interest in enhancing human pose modeling for indoor surveillance, with a particular focus on leveraging the discrete wavelet transform (DWT) for precise skin color segmentation from monocular videos. R. Brouet et. al. [7] proposed a transformative approach by redefining professional pattern-grading criteria as geometric requirements, emphasizing design preservation, proportionality, and fit. This reframing sets the stage for a more efficient and automated garment transfer process. W. Xu et. al. [8] proposed a sensitivity-based blending scheme, efficiently incorporating nearby examples for a given pose to generate realistic clothing animations. Additionally, a stochastic optimization-based greedy scheme optimizes pose space sampling, contributing to the solution's speed and compactness for real-time applications. The system's strategic configuration, with two Kinects capturing non-overlapping upper and lower body parts and a third capturing the middle part, showcases its efficacy in mitigating interference phenomena [9]. J. Ehara and H. Saito [10] proposed a cutting-edge approach for overlaying custom textures onto plain T-shirts worn by users in augmented reality settings. An image-based approach to fitting virtual clothes, where users freely move in front of a virtual mirror displaying a superimposed virtual garment [11]. A pioneering 2D model for human body contours combines an adaptable contour person representing the naked body with a low-dimensional clothing model [12]. Weng et. al. [13] proposed a groundbreaking algorithm, departing from conventional linear methods, for 2D shape deformation based on nonlinear least squares optimization. A novel approach to illuminant estimation involves leveraging color statistics derived from automatically detected faces in

images [14]. Guan-Chun Luh [15] proposed a cutting-edge human face detection scheme by synergizing a novel RGB-CbCrCg hybrid color model with the Viola-Jones face detector. A pioneering approach for automated monitoring and detection of human movements in video-based applications focuses on the crucial task of human body segmentation [16]. K. Srinivasan, K. et. al [17] addressed the disparity between background subtraction methods explored in fundamental research and those employed in practical applications, particularly in areas like traffic surveillance

III EXISTING SYSTEM

There are various applications like for the right room, there are a few References Locksmith Chamber (published via IEEE) Focuses on the use of Kinect software program Kinect sensor is used to stumble on frame actions. And 3-d modelling competencies Advertisement You will flip the image golden. Blender software mainly because of three-D modelling and hardware" In this, he is don't keep any disgusting elements Convenient and easy to put in. Machine Learning Approaches (Published by means of IEEE) Contains device learning algorithms for face detection and eye actions to achieve the person's coordinates. It's far Access to over 60,000 algorithms appropriate photograph processing strategies should be adopted (IEEE Publications) consists of image techniques filtering through a filter out which includes a Gaussian clear out Filters, Gaussian Smoothing, Conservative Smoothing; random filter, Laplacian of Gaussian filter; High and low bypass filters, side enhancement filters; open mouth clear out, and so forth. An Ideal Testing Platform Using Augmented Reality (Published through International Journal of Advanced Technology) Computer Technology IJACT): Kinect Sensor Finally, the want for deep sound and RGB When the sensor is allowed to face in front of the consumer Kinect webcam trials body And shows the skeleton of the consumer's body.

This is the main downside of the contemporary gadget Can't paintings without weapons some structures are like this Advanced technology can be required extra gadgets inclusive of sensors to function this is accurate. This will reduce get entry to the device. Or handy for customers. Existing systems are less correct whilst populated in actual time. A current account may additionally depend on person download Photos aren't usually the nice accuracy, which results in doubtful virtual features. Existing structures can be pricey to implement, proscribing their adoption Small sellers or the ones on a decent budget. Even with advanced technology, digital examination rooms since it is not widely generic with the aid of clients Even if we don't agree with or care Very tough technology to apprehend.

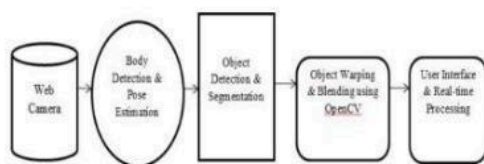
IV PROPOSED SYSTEM

Its goal is to offer a virtual check room gadget Users using intuitive web applications. System Built with Flask using HTML, CSS and JavaScript various functions and keys are attractive and an interface has been created to welcome customers Languages, mainly English, are clean any user will apprehend. There have to be a "Tryon" button Selected through the user if he wants to strive the clothes. The image of the garment is imprinted on the person's body Open CV makes use of input from the digital. If the user desires they should purchase garments or put on them formally now. The machine we offer makes a speciality of internet site improvement. Using the utility to document video from the consumer Camera device separates video Records the

person's frame and extracts from them Using capabilities for structural morphing, rotating and scaling The photo can be transferred to the user in real time Also serves to obtain statistics approximately Location of joints in the body. The foremost objective of this assignment is to create I need an example of an online digital test bed Customers can try on clothes truly. It's miles Very useful for the duration of and after the attempt Covid-19 pandemic. This could be for customers to make it smooth and efficient.

Proposed implementation of digital device testing gadget helps the user study Billions of options, but he sat at home how they're presented physically. With this app you can see users how garments should match and appearance. This software is beneficial in pandemic scenario. And no threat to experience buying garments due to Covid-19 Pandemic will take a look at the boundaries of offline stores. By This technique could be very beneficial at the moment. These things guard the seller from damage to the garments. Thank you for the convenient opportunity for customers your treasures. Let's build a continuously enhancing gadget Descriptions are correct and designated Completed if we evaluate this scheme with other existing schemes systems, main differences, shortcomings any hardware in our program.

V BLOCK DIAGRAM



VI MODULES/ IMPLEMENTATION

- Detecting and Figuring Out Body Form
- Face Recognition
- Image is Being Hidden
- Edge Detection
- Measuring Work Order
- Body Elements trialing
- Information Coaching
- Statistics Education
- You May Wear a Digital Get Dressed
- Person Interface
- Actual Time Processing
- Result

VII MODULES DESCRIPTION

A. Detecting and Figuring Out Body Form

The first step in on-line virtual testing is to attain a way of area Body, head or neck size relies upon on what sort of wearable device you want Checkpoints are used to decide where they had been moved. It refers to a particular design or material. As the body paperwork, we several methods were used:

- i) Threshold filtering, Canny Edge detection, K-Mode and
- ii) Movement detection or bone detection,

Wherein multiple frames are resolved for every motion. However, the consequences were They're stuttering and no longer geared up for the scene wearable Therefore, we gift a basically new identity method The detected location of the user's face, the boundaries on his head and Wearable display gadgets are based on this point. Also, one element you can get the link using Augmented Reality (AR) subject matter. Although it became good enough for small gadgets like glasses or jewellery, however now not exact sufficient. Clothing that fits the person's body. Like form the consumer follows the function like bodily extraction. Idea Hold the person in front of the camera first Algorithm approximately shoulders and drawn factors of a positive distance by measuring the space between these factors at the stomach the distance from the consumer to the digital camera can be obtained from the scale of the person. When a photo (video frame) is obtained, the brink detection filter out is activated. He acquired simplest a photo of the frame. Canny mouth detection Raw statistics is sensitive to noise; is sound asleep Uses a clear out in which the supply photo is blended the use of a Gaussian clear out. Match Four convolution filters have been used for horizontal, vertical and diagonal detection Process image edges. A morphological function is likewise used closed shaded exterior finally, the 8-factor Freeman code is used. To assign a path to each pixel. User frame detection and each cascade built into Open CV unlock it from historical past.

B. Face Recognition

For example, when a consumer accesses a display to locate a completely unique user the shape have to be observed. So we use it to come across faces Feature-Based Cascade Haar Classifier. To use Haar Classifier It uses pixel depth values, various the assessment between values neighbouring celebration factors. Then assessment among pixels Circles are used to outline relative light and darkish regions of an image. It's far Hence system gaining knowledge of method, layering algorithm works nicely played roles in diverse poor and high quality movies. Quite a few negativity photographs (pix without faces) and wonderful photographs (snap shots with faces). It is indicated to establish a class and extract features from it. Target OpenCV will become smooth to use because it comes with pre-skilled classifiers Face, eyes, smile and so forth. The package consists of a instructor and a detector; we can educate him With our assortment, it is simple to find any object. If he finds a suit, it really is all Return rect(x, y, w, h) which imports the left, pinnacle, backside, and proper coordinates.

C. Image is Being Hidden

This mask image has some simple pixel depth values set any pixel intensity fee in the image isn't automatically 0. The resulting pixel density masked photo is positioned at the history for a price this is usually zero. Or ROI for each section to outline a mask If important, he might check mask in bars in ROI. Device. In the ROI tool, masks do not have an effect on operations without clipping the ROI.

D. Edge Detection

There are many methods to remove mouth ulcers. We used canny area the detection approach, as mentioned earlier, is body detection. It Mouth detection technique. Gaussian filter is used. These filters are disconnected Noise within the virtual photo to save you misdiagnosis of the procedure. It works to clean and reduce the consequences of photo noise for proper functioning of the system. In this case, the depth level Image no longer located the edges of the image may be marked at various points. Instructions like horizontal, vertical and diagonal, so this algorithm uses four filters detected brown species at all edges of the image. After this method Maximum stress is carried out in order that it remains thin in the mouth. That is oppression Edge consequences are more particular additives than real edges. Ahead, some elements can create noise after which we use doubling for such factors set restrict.

E. Measuring Work Order

Attire scaling manner resizing the picture thus accordingly, it must alternate whilst the person moves in the front of the screen. The size and function of apparel for your body. While consumer Moves to the screen and increases the size of the photo consequently person, but the actual length of the garment have to now not be expanded. Its miles this became performed the usage of the setup approach.

F. Body Elements trialing

Open CV pose estimation algorithms which include Open Pose or Alpha Pose are used trial consumer's frame components like head, frame, and palms. This enabled us to insert consumer records digital bodysuit.

G. Information Coaching

We gathered a dataset of clothing photographs from unique angles and sizes. We are the images were then pre-processed to remove colour noise. Constant photograph size and format

H. You May Wear a Digital Get Dressed

Computer imaginative and prescient techniques such as homograph, photo distortion and blending. They follow digital clothes to the consumer's frame in actual time. Its miles the illusion is created that the user is carrying these clothes.

I. Person Interface

A simple consumer interface became evolved the usage of the Open CV GUI toolkit users can select distinct clothes and add-ons. That does it the software is simple and intuitive to use.

J. Actual Time Processing

The algorithms are optimized for actual-time processing and rendering. A digital garment at the consumer's frame. This allowed the user to look himself Virtual costumes in actual time.

- System: Pentium i3 Processor.
- Hard Disk: 500 GB.
- Monitor: 15" LED
- Input Devices: Keyboard, Mouse
- Ram: 4 GB

B. SOFTWARE REQUIREMENTS

- Operating system: Windows 10.
- Coding Language: Python
- Web Framework: Flask

IX. RESULT AND DISCUSSION

Let customers enjoy clothes in a comfy and appealing manner in almost real time. The application can discover the person's frame and Allows real-time trialing of body parts, intuitive and interactive the revel in is a virtual garment that is exactly suited to the person's body. Wearing these clothes gives an actual appearance. The user tool is intuitive and clean to apply, allowing users to quickly select and use about the various costumes and characters. Real-time processing and rendering Virtual Closet was rapid and green, allowing the person to work easily with none troubles.

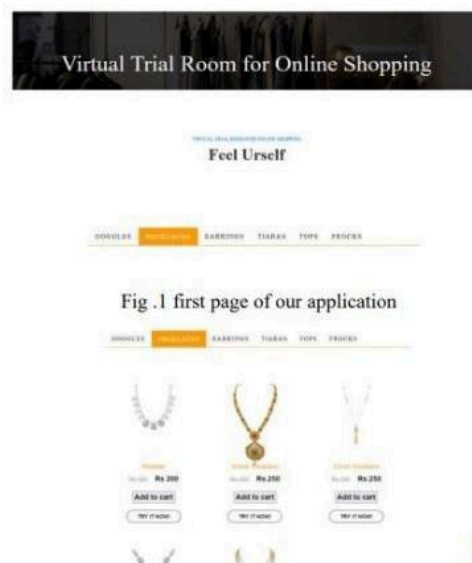
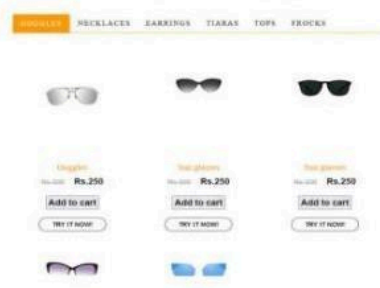


Fig .1 first page of our application

Fig. 2 Necklace products in the application



VIII SYSTEM REQUIREMENTS

A. HARDWARE REQUIREMENTS

Fig. 3 Sunglasses products in the application



Fig. 4 Accessories products in the application

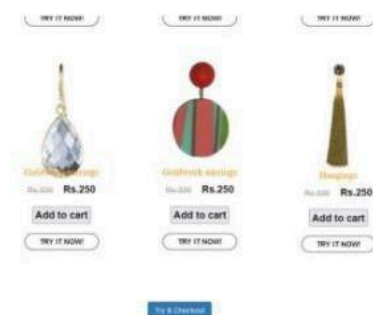


Fig. 5 Accessories products in the application

X CONCLUSION

The rise of on line buying and the choice of customers to use it at the equal time of route you're completely satisfied along with your buy There is a want to develop a gadget that clearly fits the people He selected clothes. To meet this demand, we proposed a gadget It is designed to paintings without the need for outside dimmers or something else Hardware inventions that generally run and generate algorithms Restrictions on their use. The proposed system gives very within your means price One answer for all lets in human beings to try distinct clothes at low cost Barriers make on-line buying more pleasurable. Ultimately, virtual becoming rooms are a new generation for on-line purchasing. Technology has the ability to change the manner human beings purchase clothes reality. Advanced generation is used in the improvement of this era Augmented reality, laptop vision and system mastering problems are solved The trouble of no longer being able to try on clothes earlier than buying; Provides customers with an immersive and customized shopping experience. A proper take a look at chamber offers many advantages together with visualization competencies How to expose clothes to clients, comfort in buying they had been easily reproducible and capable of offering a huge type of training. This additionally helps Diminishing ability for business returns due to scale and complexity Increase client satisfaction. Project achievement relies upon on ease of use of the system Additional tools in addition to the webcam already present at the computer. Or

PC Additionally, the usage of Open CV gives a few blessings in layout Real-time picture processing, fee-effectiveness and size. In the future, the algorithm could be tailored to stumble on human silhouettes More complex situations with extraordinary colorations and accelerated noise Degree This application units a new trendy for on line shopping and many others. An enticing and personalised purchasing enjoy for clients. Usually, the right courtroom has the energy to alternate the way human's store. Online Apparel, Improve Customer Experience and Improve Business Efficiency is supplied for e-trade.

ACKNOWLEDGMENT

we extend our sincere gratitude to Jithesh .T for their invaluable contributions to this research endeavor. Their expertise, guidance, and dedication have greatly enriched our work. We also acknowledge the support and resources provided by Sathyabama institute of science and technology

REFERENCES

- [1] K.Srinivasan, K.Porkumaran, G.Sai Narayanan, "Intelligent human body trialing, modelling and activity analysis of video surveillance system: A Survey", Journal of convergence in engineering, technology and science, Vol.1, pp.1-8, 2009.
- [2] Max Mignotte, "Segmentation by Fusion of Histogram based K-Means Clusters in different color space" IEEE Transactions on Image Processing, Vol.17, pp.780-787, 2008.
- [3] F. Cordier, W. Lee, H. Seo, and N. Magnenat-Thalmann, "From 2D Photos of Yourself to Virtual Try-on Dress on the Web", Springer, pp. 31-46, 2001.
- [4] D. Protopsaltou, C. Luible, M. Arevalo-Poizat, and N. Magnenat Thalmann, "A body and garment creation method for an internet based virtual fitting room", in Proc. Computer Graphics International 2002 (CGI '02). Springer, pp. 105-122, 2002.
- [5] F. Cordier, H. Seo, and N. Magnenat-Thalmann, "Made-to-measure technologies for an online clothing store", IEEE Comput. Graph. Appl., vol. 23, no. 1, pp. 38-48, Jan. 2003.
- [6] K.Srinivasan, K.Porkumaran, G.Sai Narayanan, "Skin colour segmentation based 2D and 3D human pose modelling using Discrete Wavelet Transform", Journal of Pattern recognition and image Analysis, Springer, Vol.21, pp.740-753, 2011.
- [7] R. Brouet, A. Sheffer, L. Boissieux, and M.-P. Cini, "Design preserving garment transfer", ACM Trans. Graph., vol. 31, No.4, pp. 36:1-36:11, Jul. 2012.
- [8] W. Xu, N. Umentani, Q. Chao, J. Mao, X. Jin, and X. Tong, "Sensitivity optimized rigging for example-based real-time clothing synthesis", ACM Trans. Graph. (Proc. of SIGGRAPH 2014), Vol. 33, No. 4, Aug. 2014.
- [9] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan, "Scanning 3D full human bodies using kinects", IEEE Transactions on Visualization and Computer Graphics (Proc. of IEEE Virtual Reality), Vol. 18, No. 4, pp. 643-650, 2012.

- [10] J. Ehara and H. Saito, "Texture overlay for virtual clothing based on pca of silhouettes", in Proceedings of the 5th IEEE and ACM International Symposium on Mixed and Augmented Reality, ser. ISMAR '06. IEEE Computer Society, pp.139–142, 2006.
- [11] Z. Zhou, B. Shu, S. Zhuo, X. Deng, P. Tan, and S. Lin, "Image-based clothes animation for virtual fitting", in SIGGRAPH Asia 2012 Technical Briefs, pp. 33:1–33:4,2012.
- [12] P. Guan, O. Freifeld, and M. J. Black, "A 2D human body model dressed in eigen clothing", in Proceedings of the 11th European Conference on Computer Vision: Part I, pp. 285–298, 2010.
- [13] Y. Weng, W. Xu, Y. Wu, K. Zhou, and B. Guo, "2D shape deformation using nonlinear least squares optimization", *Vis. Comput.*, vol. 22, no. 9, pp. 653–660, 2006.
- [14] S. Bianco and R. Schettini, "Color constancy using faces", in 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 65–72, 2012.
- [15] Guan-Chun Luh "Face detection using combination of skin colour pixel detection and viola jones face detector", in 2014 IEEE International Conference on Machine Learning and Cybernetics(ICMLC), pp.364-370, 2014.
- [16] K.Srinivasan, K.Porkumaran, G.Sai Narayanan, "Background subtraction techniques for human body segmentation in indoor video surveillance", *Journal of Scientific and Industrial Research*, Vol.73, pp.342-345, 2014.
- [17] K.Srinivasan, K.Porkumaran, G.Sai Narayanan, "Enhanced Background Subtraction Techniques for Monocular Video Applications", *International Journal of Image Processing and Applications*, Vol. 1,no. 2, pp.87-93, 2010.