Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

Try-On AI Fashion (FitSnap AI): A Computer Vision and 3D Reconstruction Framework for Virtual Outfit Fitting in E-Commerce

Mahesh Y N [4MG23CS030]
Dept. of Computer Science and Engineering
G Madegowda Institute of Technology
Mandya, India

Hemanth Kumar S [4MG22CS018]
Dept. of Computer Science and Engineering
G Madegowda Institute of Technology
Mandya, India

Hiteshkumar U R [4MG22CS019]
Dept. of Computer Science and Engineering
G Madegowda Institute of Technology
Mandya, India

Under the guidance of **Prof. Rashmi**

***Abstract* -** Shopping for clothes online remains a guessing game: a buyer can view a picture of an outfit but cannot see how it will actually look on their own body, so size and fit are decided on hope rather than evidence. This uncertainty is a leading cause of the high return rates that burden online fashion retailers. This paper presents Try-On AI Fashion (FitSnap AI), a web-based platform that lets a shopper virtually try on an outfit before purchase. The user uploads two photographs — one of themselves and one of the clothing item — and an AI pipeline combines them into a realistic try-on image within seconds. The system further validates the quality of the generated result, provides size advice derived from body measurements, builds a rotatable 360° 3D view of the fitted outfit, and generates a short walking preview video. The entire pipeline is built using free, open-source tools and runs on a standard laptop, requiring no paid software or dedicated cloud infrastructure. A review of recent virtual try-on literature shows that existing GAN-based, parser-free, and diffusion-based approaches stop at a flat two-dimensional output and do not offer an accessible application with 3D visualization, size guidance, and a low-cost setup — a gap this project addresses directly.

***Keywords* -** Virtual Try-On; AI Fashion; Computer Vision; 3D Reconstruction; Body Pose Estimation; Image Synthesis; E-commerce; OpenCV.

I. INTRODUCTION

Shopping for clothes online is still largely a guessing game. A buyer can see a picture of an outfit but cannot see how it will actually look on their own body, so size and fit are decided on hope rather than evidence. This uncertainty is a major reason why online fashion stores face high return rates, costing both buyers and sellers time and money. Try-On AI Fashion, also referred to as FitSnap AI, is built to close this gap with a simple idea: let the shopper try the outfit on virtually before they buy it.

The platform works as a straightforward web application. The user uploads two photos — one of themselves and one of the clothing item — and an AI model combines them into a realistic try-on image within seconds. The app then checks the quality of the result, gives size advice from body measurements, builds a rotatable 360° 3D view, and generates a short walking preview video. Every part of the pipeline runs on free, open-source tools on a normal laptop, requiring no paid software and no expensive cloud infrastructure.

The remainder of this paper is organized as follows. Section II reviews related work in virtual try-on and AI-based fashion synthesis. Section III states the problem addressed. Section IV discusses the motivation. Section V lists the objectives. Section VI presents the tools and technologies used. Section VII describes the proposed system architecture. Section VIII details the implementation. Section IX discusses results. Section X concludes the paper, and Section XI outlines future scope.

II. LITERATURE SURVEY

A. Nurhidayanti et al. — Virtual Try-On Using Deep Learning (ICELTICs, 2024)

This work uses a GAN-based approach combining U-Net with pix2pixHD, trained on the VITON dataset, achieving a best SSIM of 0.795 and an Inception Score of 2.20. The gap identified is that the output remains a flat 2D picture, and the approach needs human-parsing masks, keypoints, and lengthy GPU training.

B. Ge et al. — Parser-Free Try-On, PF-AFN (CVPR, 2021)

This paper proposes a parser-free flow network in which a student network learns how cloth flows onto the body without requiring human-parsing, achieving an FID of 10.09 on VITON. The gap is that the result is still a flat 2D image, and the method is tested only on small front-view top photos.

C. Ning et al. — PICTURE: Try-On from Unconstrained Designs (CVPR, 2024)

This work uses a two-stage Stable Diffusion pipeline, generating style first and then texture, which allows mix-and-match designs. The gap is that it needs an accurate body map, performs weakly in dark lighting or heavy overlap, and offers no 3D output.

D. Guo et al. — AI Assisted Fashion Design: A Review (IEEE Access, 2023)

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

This is a survey of AI applications in fashion, covering detection, design creation, and recommendation. The gap is that it is only a review, with no working try-on application, no 360° view, and no size advice.

The common trend across these four works is that they stop at either a flat 2D image or a literature review — none of them offers an easy-to-use application that combines a 3D view, size advice, and a free, low-cost setup. Try-On AI Fashion is proposed to fill precisely this gap.

TABLE I. SUMMARY OF REVIEWED LITERATURE

Paper & Topic	What They Did	Gap Identified	Relevance to Project
Nurhidayanti et al., ICELTICs 2024 (GAN-based try-on)	GAN with U-Net + pix2pixHD trained on VITON; best SSIM 0.795, IS 2.20.	Flat 2D output only; needs masks, keypoints and long GPU training.	Confirms feasibility of GAN-based warping but motivates a lighter, non-GPU pipeline.
Ge et al., CVPR 2021 (PF-AFN, parser-free)	Student network learns cloth flow without human parsing; FID 10.09 on VITON.	Still 2D; tested only on small front-view top photos.	Supports using parser-free ideas to simplify the pipeline used here.
Ning et al., CVPR 2024 (PICTURE, diffusion-based)	Two-stage Stable Diffusion: style then texture; allows mix-and-match.	Needs accurate body map; weak in low light or overlap; no 3D.	Highlights the 3D gap that this project directly targets.
Guo et al., IEEE Access 2023 (review)	Survey of AI in fashion: detection, design, recommendation.	Review only; no working app, no 3D, no size advice.	Confirms no prior accessible app combines 3D + size advice + free setup.

III. PROBLEM STATEMENT

To design and develop an AI-powered virtual fashion try-on platform that allows users to upload a product image and their selfie, generate a personalized 3D representation, and preview the selected outfit interactively before making an online purchase. The two central problems addressed are the lack of a try-before-you-buy option in online fashion shopping, and the resulting wrong-size purchases and high return rates.

Conventional e-commerce fashion platforms present the same flat product photograph to every visitor, regardless of their body shape, height, or skin tone. The buyer is left to mentally extrapolate how a garment drapes on a mannequin or model onto their own body, a task that is inherently unreliable. Size charts attempt to bridge this gap numerically, but garments of the same labeled size vary across brands and cuts, and a chart cannot convey how a fabric will fold, stretch, or sit on an individual's frame. The absence of a dependable, personalized preview mechanism is therefore not a minor inconvenience but a structural weakness of online fashion retail that directly translates into lost revenue, logistics overhead from reverse shipping, and diminished customer trust.

IV. MOTIVATION

The motivation for this project arises from the uncertainty about how clothes will look on an individual customer, the high dependence on generic size charts and static product photographs, the need for a more personalized online shopping experience, the increasing adoption of AI and virtual visualization in e-commerce, and the need to reduce hesitation and improve buyer confidence before purchase. The project aims to provide an interactive and user-friendly virtual fitting experience that can support better online fashion decisions.

Beyond the immediate problem of fit, there is a broader industry shift toward experience-driven commerce, where customers increasingly expect digital interactions to mirror the richness of a physical store visit. Virtual try-on sits at the intersection of this trend and the practical economics of returns: industry reports consistently cite poor fit as one of the leading causes of apparel returns, and every returned item carries costs in reverse logistics, repackaging, and in many cases, disposal of unsellable stock. A lightweight, accessible try-on tool that runs without specialized hardware therefore has the potential to benefit small and medium fashion sellers who cannot afford expensive proprietary visualization systems, not just large retailers.

V. OBJECTIVES

The main objectives of the project are: (1) to allow users to upload a fashion product image and a personal selfie; (2) to process the user's image for personalized virtual try-on; (3) to generate a 3D representation of the user for visualization; (4) to provide an interactive 360° view of the virtual outfit; (5) to build a simple and responsive web-based user interface; and (6) to support a more personalized and engaging e-commerce experience.

In addition to these primary goals, the project targets a set of supporting objectives that shape the engineering approach: keeping the entire pipeline buildable from free and open-source components so that the solution remains reproducible without licensing cost; keeping per-request processing time low enough that the try-on experience feels interactive rather than batch-oriented; producing a size recommendation that is derived from measurable image cues rather than a fixed chart lookup; and structuring the codebase so that individual stages, such as pose estimation or garment warping, can be replaced with stronger models in the future without redesigning the whole system.

VI. TOOLS AND TECHNOLOGIES

The system is implemented using Python as the primary programming language, with Flask and FastAPI as backend frameworks. The frontend is built using HTML, CSS, and JavaScript. SQLite, a relational database, is used for data storage. OpenCV along with other image-processing and AI models is used for the computer vision components, and web-based 3D rendering is used for visualization. Development tools include VS Code, a Python virtual environment, and Node.js.

Python was chosen for its mature ecosystem of computer vision and machine learning libraries, which minimizes the amount of low-level code that needs to be written for image handling, pose detection, and model inference. Flask is used for

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

lightweight, synchronous routes such as serving pages and handling form-based uploads, while FastAPI is used for the AI-processing endpoints that benefit from asynchronous request handling and automatic request validation, which reduces errors when images of varying resolution and format are submitted by users. SQLite was selected over a heavier relational database because the project does not require concurrent multi-user write-heavy workloads at this stage, and it allows the entire application, including its data layer, to run without any external database server, which keeps setup effort minimal for a student project and for future deployment on modest hardware.

VII. PROPOSED SYSTEM ARCHITECTURE

The architecture begins with the user uploading a selfie and a product image through the web platform. This input is passed to an AI processing stage that performs face and body detection, followed by 3D avatar generation. A virtual try-on model then fits the selected garment onto the generated avatar. The final output is a 360° rotatable model of the user wearing the outfit, accompanied by style and size recommendations.

The architecture is intentionally organized as a pipeline of loosely coupled stages rather than a single monolithic model, which keeps each stage independently testable and replaceable. The upload stage is responsible only for accepting and validating the two input images, rejecting files that are corrupted, too low in resolution, or do not contain a detectable face or garment. The AI processing stage is the computational core and is itself decomposed into face and body detection, followed by 3D avatar generation that maps detected keypoints into a parameterized body mesh. The virtual try-on stage consumes this avatar along with the segmented garment image and produces the composited result. A final presentation layer renders the 360° rotatable view in the browser and surfaces the size and style recommendation alongside it, so the user sees the visual result and the numeric guidance together rather than as separate outputs.

VIII. DATA FLOW AND SYSTEM REQUIREMENTS

Data moves through the system in a strict sequence that mirrors the architecture diagram presented in the project design: the browser client sends the selfie and product image to the upload endpoint; the backend persists the images temporarily and hands them to the AI processing service; the AI processing service returns a composited try-on image, a set of rotation frames, a short preview video, and a size/style recommendation object; and the backend forwards all of these to the client, which renders them in the results view. No stage is allowed to proceed until the previous stage has returned a valid result, and any failure at an intermediate stage is surfaced to the user as a retry prompt rather than a partial or broken result.

A. Hardware Requirements

The system was designed to run on a standard laptop with a multi-core CPU and at least 8 GB of RAM, without requiring a dedicated GPU. A webcam or phone camera is sufficient for capturing the user's selfie, and no specialized scanning or depth-sensing hardware is needed.

B. Software Requirements

The backend requires Python 3 together with Flask, FastAPI, and OpenCV, along with supporting image-processing and lightweight AI libraries. The frontend requires only a modern web browser capable of rendering HTML5, CSS3, and JavaScript, with no browser plugins or native applications needed. SQLite is bundled with Python's standard library, so no separate database server installation is required, and the complete development environment can be set up using VS Code, a Python virtual environment, and Node.js for any frontend tooling.

IX. SYSTEM MODULES AND WORKING

The working of Try-On AI Fashion follows the same user journey shown in the project presentation: the user lands on the web platform, uploads a selfie and a chosen product photo, and receives a processed, interactive result a few seconds later. This section describes each module that makes up that journey in more depth.

A. User Interface Module

The interface presents a clean two-step upload form — one slot for the selfie and one for the garment image — followed by a results view once processing completes. A progress indicator is shown while the AI pipeline runs so the user understands the system is working rather than unresponsive. The results view embeds the generated try-on image, the rotatable 360° viewer, the walking preview video, and the size/style recommendation panel on a single page.

B. Face and Body Detection Module

Once the two images are received by the backend, OpenCV-based detectors locate the face and estimate body keypoints (shoulders, torso, waist, hips) on the selfie. These keypoints anchor every later stage: they tell the system where the garment needs to sit, how large it should be scaled, and what the user's approximate proportions are for the size-advice feature.

C. 3D Avatar Generation Module

The detected keypoints are mapped onto a parameterized 3D body mesh, producing a personalized avatar that approximates the user's height, build, and posture. This avatar is the bridge between a flat 2D selfie and the 3D try-on experience — it is what actually gets rotated in the 360° viewer, rather than a flat image being spun in place.

D. Virtual Try-On Module

The product image is segmented to remove its background, leaving only the garment. The garment is then geometrically warped so that its key points (collar, sleeves, hem) line up with the corresponding points on the user's avatar, and it is composited onto the avatar with blending at the edges to avoid a pasted-on look. This is the step most directly comparable to the GAN-based and parser-free approaches discussed in the literature survey, implemented here with lightweight, non-GPU-dependent warping so it can run on a normal laptop.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

E. Quality Check Module

Before the result is shown to the user, an automated check verifies that the composited image is not malformed — for example, that the garment has not been placed with an unrealistic offset or scale. If the check fails, the user is prompted to retry with a clearer or better-lit photo rather than being shown a broken result.

F. Size and Style Recommendation Module

Approximate body measurements derived from the avatar (shoulder width, torso length, waist-to-hip ratio) are compared against standard garment sizing bands to suggest the most likely correct size, and basic style cues from the garment image are used to suggest whether the fit is likely to run loose, regular, or tight on the user's frame.

G. 360° View and Preview Video Module

The fitted avatar is rendered from a sequence of camera angles around the vertical axis, which are stitched into an interactive viewer that the user can drag to rotate, and separately compiled into a short walking preview video that gives a more natural sense of movement than a static rotation.

X. IMPLEMENTATION

The implementation proceeds through the following modules: (1) a user image and product image upload module; (2) image validation and pre-processing; (3) AI-based face and body feature extraction; (4) personalized 3D representation generation; (5) virtual clothing fitting and rendering; (6) an interactive 360° preview interface; and (7) web integration to deliver a smooth end-to-end user experience.

A. Upload and Pre-processing

The upload module accepts standard image formats through a browser-based form and immediately runs validation checks, including file-size limits, resolution thresholds, and a sanity check that a human face and a garment region can be detected in the respective images. Pre-processing normalizes image orientation and lighting using OpenCV routines such as resizing, color-space correction, and background cleanup, so that downstream models receive images in a consistent format regardless of the device or camera used to capture them.

B. Feature Extraction and 3D Representation

Face and body feature extraction identifies key landmarks that describe the user's posture and proportions, which are then used to construct a personalized 3D avatar. This avatar acts as the surface onto which the selected garment is later fitted, and it is also the basis from which body-measurement estimates are derived for the size-advice feature.

C. Garment Fitting and Rendering

The garment image is segmented from its background and warped so that its contours align with the body landmarks of the avatar. The warped garment is then composited onto the avatar, and lighting and edge blending are applied so that the final image appears natural rather than pasted. The resulting fitted avatar is exported in a form that supports rotation, enabling the interactive 360° preview, and a short sequence of rendered frames is assembled into the walking preview video.

D. Web Integration

All modules are exposed through REST endpoints and consumed by a JavaScript-driven frontend, which presents the upload form, a progress indicator while AI processing runs, and the final interactive viewer. This separation between backend processing and frontend presentation allows the processing pipeline to be tested and improved independently of the user interface.

XI. TESTING AND VALIDATION

Testing was carried out at both the module level and the end-to-end pipeline level. Module-level testing verified that the upload validator correctly rejected corrupted or unreadable files, that the face and body detector correctly located keypoints on a range of sample selfies, and that the garment segmentation module correctly isolated the clothing item from varied product-photo backgrounds. Pipeline-level testing ran the complete upload-to-preview flow on multiple selfie and garment pairs to confirm that the final composited image, 360° view, and preview video were generated without errors and within an acceptable processing time.

Edge cases that were specifically tested included low-resolution images, images with partial occlusion of the body, and images captured in dim lighting. These cases highlighted the same limitations observed in the reviewed literature — performance is strongest for clear, front-facing, well-lit photographs and degrades as image quality or pose complexity increases — and they informed the validation checks added to the quality check module so that poor-quality results are caught before being shown to the user rather than silently delivered.

Usability testing was also conducted informally with a small group of peer users who were asked to upload their own selfies and a selection of product images, then rate the clarity of the interface, the perceived realism of the generated preview, and the usefulness of the size recommendation. Feedback from this round was used to simplify the upload form, add clearer loading-state messaging during AI processing, and adjust the default framing guidance shown to users before they capture their selfie, which noticeably reduced the proportion of uploads that failed the initial face-and-body detection check.

XII. RESULTS

The developed prototype successfully accepts a user selfie and a product image, generates a realistic virtual try-on image, builds a rotatable 3D preview of the fitted outfit, and provides size recommendations based on extracted body measurements. The system runs entirely on free, open-source tools on standard hardware, confirming the feasibility of a low-cost, accessible virtual try-on solution.

Across repeated trials with different users and garment images, the pipeline consistently completed the upload-to-preview cycle within a few seconds on a standard laptop without a dedicated graphics processing unit, which supports the project's goal of accessibility. The generated try-on images preserved the overall silhouette and pose of the uploaded selfie

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

while correctly draping the selected garment over the detected body region in the majority of test cases, with the most reliable results obtained for front-facing, well-lit photographs. The 360° view and walking preview video functioned as intended, giving the user a sense of the outfit from multiple angles rather than a single static composite. As expected for a system built on lightweight, non-GPU-dependent models, performance degraded for images with unusual poses, poor lighting, or heavy occlusion, which is consistent with the limitations noted for comparable approaches in the literature survey.

TABLE II. QUALITATIVE RESULT SUMMARY

Evaluation Aspect	Observation	Condition
Processing time (upload to preview)	A few seconds on a standard laptop, no GPU	Front-facing, well-lit image
Garment alignment accuracy	Consistently correct drape over detected body region	Front-facing, well-lit image
360° view / preview video	Smooth rotation and walking preview generated	All valid inputs
Size recommendation	Matched expected size band in most trials	Clear full-body selfie
Robustness to pose/lighting	Noticeable quality drop	Side pose, low light, occlusion

A. Discussion and Limitations

The results confirm that a lightweight, non-GPU pipeline is sufficient to deliver a usable virtual try-on experience, which is the central claim of this project. At the same time, the dependence on front-facing, well-lit input images is a genuine limitation shared with the parser-free and GAN-based approaches reviewed in Section II, and it means the current prototype is best suited to a controlled selfie-capture flow rather than arbitrary, uncontrolled photographs. The absence of fabric-specific physics also means that heavier or looser garments are rendered with less realism than fitted garments such as T-shirts, an area identified for future improvement.

XIII. COMPARISON WITH EXISTING SOLUTIONS

It is useful to situate Try-On AI Fashion relative to both the academic approaches surveyed in Section II and the commercial virtual try-on features offered by a small number of large fashion retailers. Commercial solutions typically rely on proprietary, cloud-hosted models trained on large internal datasets and operated at significant infrastructure cost, which places them out of reach for smaller sellers. The GAN-based and diffusion-based academic approaches reviewed earlier generally achieve higher photorealism than a lightweight geometric-warping pipeline, but at the cost of GPU-dependent training and inference, and, in the case of the reviewed works, without any accompanying 3D visualization or size-advice feature.

Try-On AI Fashion occupies a middle ground: it trades some of the photorealism achievable with heavier generative models for accessibility, running on ordinary consumer hardware, and for the additional features of 3D rotation, a walking preview video, and size guidance that none of the four reviewed works provide together. This trade-off is appropriate for the target use case of small and medium online sellers who need a workable, low-cost try-on feature rather than a research-grade photorealistic renderer.

XIV. ADVANTAGES AND APPLICATIONS

A. Advantages

The system requires only two user-supplied photographs and no specialized hardware such as depth cameras or scanning rigs, which keeps the barrier to entry low for both shoppers and sellers. Because every component is free and open-source, the platform can be run and extended without licensing cost, making it realistic for small and medium fashion businesses to adopt. The modular pipeline design also means individual stages can be upgraded independently as stronger open-source models become available, without requiring a full system redesign.

B. Applications

Beyond individual online stores, the approach is applicable to fashion marketplaces that aggregate many sellers and want a consistent try-on feature across listings, to rental and second-hand clothing platforms where physical trial is often impractical, and to in-store kiosks that let customers preview items without physically changing clothes. The size-advice component is also independently useful as a standalone fit-prediction tool that could be integrated into existing e-commerce checkout flows to reduce size-related returns even without the full 3D preview.

The underlying pipeline is also adaptable outside fashion retail: the same selfie-to-avatar and garment-warping approach could support virtual costume previews for events, uniform fitting for institutions that issue standardized clothing, or visualization tools for tailors and fashion design students who want to preview a design on a range of body types before committing to physical production.

XV. CONCLUSION

See it on you before you buy. The proposed Try-On AI Fashion platform demonstrates the application of AI, computer vision, and 3D visualization to create a personalized virtual fitting experience. By combining user images, product images, image processing, and interactive visualization, the system addresses a major limitation of conventional online fashion shopping. The platform can be extended into a scalable e-commerce solution with authentication, payments, product catalogs, and advanced AI-based fitting capabilities.

The project demonstrates that a useful virtual try-on experience does not require expensive proprietary infrastructure: a careful combination of open-source computer vision tools, a lightweight web stack, and a well-structured pipeline is sufficient to move from a flat product photograph to an interactive, personalized 3D preview with size guidance. This positions the system as a practical reference point for small and medium online fashion sellers who want to offer a try-

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

before-you-buy experience without the cost barriers associated with commercial virtual try-on platforms.

XVI. FUTURE SCOPE

The project can be enhanced in the future by improving virtual outfit fitting using advanced generative AI models, supporting multiple clothing categories and complete outfit combinations, adding real-time camera-based virtual try-on, integrating secure user authentication and personalized profiles, adding a payment gateway and complete e-commerce functionality, integrating location services for nearby stores, improving 3D body modelling and cloth physics, and deploying the platform as a mobile application.

Further directions include benchmarking the pipeline quantitatively against the SSIM and FID metrics reported in the reviewed literature, so that improvements to individual stages can be measured objectively rather than assessed only qualitatively. Incorporating fabric-specific physics, such as different draping behaviour for cotton, denim, or silk, would also improve realism for garment categories beyond simple tops. Finally, extending the system to support multi-user concurrent sessions and migrating from SQLite to a more scalable database would be necessary steps toward a production-grade deployment.

Longer term, the project could explore augmented-reality try-on through a phone camera, removing the need for a separate selfie-upload step entirely, as well as personalization features that learn a returning user's preferred fits and styles over multiple sessions. Collaboration features, such as sharing a try-on preview with friends or family for feedback before purchase, and integration with popular e-commerce platforms as a plug-in module, are also promising directions for turning this academic prototype into a deployable product.

ACKNOWLEDGMENT

The authors would like to thank Prof. Rashmi for her guidance, and the Department of Computer Science and Engineering, G Madegowda Institute of Technology, Mandya, for their support throughout this project.

REFERENCES

[1] Nurhidayanti et al., "Virtual Try-On Using Deep Learning," in Proc. Int. Conf. on Electrical Engineering, Computer and Information Technology (ICELTICs), 2024.

[2] Y. Ge et al., "Parser-Free Virtual Try-On via Distilling Appearance Flows," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2021.

[3] S. Ning et al., "PICTURE: Photorealistic Virtual Try-On from Unconstrained Designs," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2024.

[4] J. Guo et al., "AI Assisted Fashion Design: A Review," IEEE Access, vol. 11, 2023.

[5] X. Han, Z. Wu, Z. Wu, R. Yu, and L. S. Davis, "VITON: An Image-Based Virtual Try-On Network," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2018.

[6] B. Wang, H. Zheng, X. Liang, Y. Chen, L. Lin, and M. Yang, "Toward Characteristic-Preserving Image-based Virtual Try-On Network," in Proc. European Conf. on Computer Vision (ECCV), 2018.

[7] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 43, no. 1, 2021.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in Proc. IEEE Int. Conf. on Computer Vision (ICCV), 2017.

[9] OpenCV Documentation and Computer Vision Resources, available: https://opencv.org.

[10] Python, Flask, and FastAPI Framework Documentation, used for system implementation.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)