International Journal of Web of Multidisciplinary Studies
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
website: https://ijwos.com
OPEN ACCESS
E-ISSN : 3049-2424
Volume 3 Issue 6 , June 2026
DOI: https://doi.org/10.71366/ijwos0302618939

CLOSETAI: A MULTIMODAL AI-DRIVEN FASHION ASSISTANCE PLATFORM USING REAL-TIME VIDEO ANALYSIS

Sarvesh Dayma[1], Tejas Patil[2], Shantanu Kadam[3], Dia Magnani[4], Bhagyashree Thorat[5]
[1,2,3,4,5] *Department of Electronics and Telecommunication Engineering, International Institute of Information Technology (I²IT), Pune, India1*

Article Info

Article History:

Published: 25 June 2026

Publication Issue:
Volume 3, Issue 6
June-2026

Page Number:
314-323

Corresponding Author:
Tejas Patil

Abstract:

The access to personal stylistic advice is limited by the cost, geographical constraints, and the scarcity of trained professionals who can give advice.This paper introduces ClosetAI, which is a web browser application capable of giving feedback on the outfit chosen by the user via a live video chat interface.The system uses the WebRTC and MediaDevices APIs to capture video from the camera, takes samples of the video frame at regular intervals and sends each frame along with the user's query and a serialized history of the conversation to the multimodal vision-language model using the Google Gemini API.The model provides feedback on the color combination, pattern matching, shape balance, and appropriateness of the outfit, and its recommendation is transmitted token by token to the client.Front end is developed using Next.js, React, TypeScript and Tailwind CSS, back end relies on Node.js API routes using WebSocket transport and Redis as a session storage.This paper contributes an architecture and design patterns for creating a web application for low-latency, multimodal consultations adaptive frame sampling, persona-based system prompting, and streaming token delivery, along with the evaluation protocol. We describe the system design and the planned evaluation in detail so that the results can be independently reproduced.

Keywords: MULTIMODAL AI, COMPUTER VISION, NATURAL LANGUAGE PROCESSING, FASHION TECHNOLOGY, REAL-TIME VIDEO ANALYSIS

1. Introduction

The integration of artificial intelligence into the field of fashion has resulted in a shift in the way consumers select their wardrobe items.Traditionally, style advice has been offered mostly to individuals who could afford private consultations or visits to shops that offered such advice in person.Consumers living in small towns and rural areas and individuals with modest means have generally not had access to such resources, thereby increasing the divide between consumers who have access to fashion advice and those who do not [1].

The ubiquity of smartphones and high-speed Internet connections have increased consumers' expectations from digital products.They now demand rapid responses to their needs and personalized services in their interaction with various digital tools.Previous attempts at providing online fashion consultations such as outfits suggestions based on catalog images or image-based stylometry were limited by the inability to account for the user's current attire and the need for an interactive dialogue between the consumer and the tool [5].

Development in large vision-language models (VLMs) has allowed analyzing visual scenes as well as discussing these scenes using natural language.The Gemini API provided by Google has enabled this functionality via a production API interface without requiring developers to train their own models from scratch [13], [16].

Research gap. Currently existing fashion-technology systems can be classified into three categories, where each system has an obvious disadvantage.Recommendation engines and curation systems [1], [2], [4], [14] work with uploaded or catalog images, producing sorted products, not the conversation.Try-on and "mirror" systems [7] place garments

onto a static image but do not provide any form of communication or occasion reasoning.Finally, attribute prediction and classification systems [5], [10], [11], [15] classify the garments correctly but fail to provide stylistic suggestions.None of the aforementioned systems provides video input, multimodal conversation, and occasion reasoning in a plugin-less web session.ClosetAI aims at solving all these problems, and we define the gap in Section 2.

In this context, ClosetAI is a full-stack web application based on WebRTC live video streaming and multimodal inference from Gemini designed to provide the user experience of chatting with a human stylist through a video call.All computations are done exclusively in the browser, eliminating the installation requirement and thus lowering the adoption barrier for any user with access to a webcam and the internet.Contributions of this paper are:

- A browser-native video chat interface that streams live webcam frames and sends them to the AI backend in near real time;
- An efficient integration of the Gemini multimodal API where every inference request takes not only a sampled frame but also all conversational context of the session;
- Comparison with state-of-the-art recommendation, try-on, and consultation systems that highlights novelty in our system (see Table 1); and
- A pre-registered reproducible experimental methodology(Section-4) focusing on measuring latency, recommendation quality, user experience, and safety, specifying metrics and instruments used.

2. **Literature** Survey

2.1 Traditional Consultation and Digital Alternatives

One-to-one communication is fundamental for classical styling, which involves the stylist giving advice directly to the client, usually in a high-end retail store environment. These services are highly personalized and have poor scalability because each stylist works only with a few customers a day; moreover, the price level excludes the majority of potential customers from using the service [14].

Subscription services like Stitch Fix and Thread partially automate this process via user questionnaires and stylist selection.Pipeline delay makes subscription services asynchronous: recommendations come too late to affect the decision and do not represent an interactive experience but a curation process [6].

Image uploading virtual try-on systems allow placing the clothes on the photo of the customer.They enhance the visualization process but have low body shape precision, are unable to perform movement analysis of the garment and do not have any conversational component; thus, user intent can be conveyed only via filters [7].

2.2 AI and Computer Vision in Fashion

CNNs resulted in great improvements in garments recognition.Research into sequential, multimodal attribute prediction revealed the ability of deep learning to classify attributes like collar type, sleeve length, and fabric design with almost human-like precision by using both image and text features [5].Later research into classification and trend analysis [10], [11], [15] corroborated that deep models are useful in recognising types of clothing and trends, but the models only output labels and do not give styling advice.

Transformers have expanded those models by being able to model long-term dependencies in sequential and visual data of fashion, and language and vision-language models have made it possible to build systems capable of reasoning about images in natural language [12], [16].Surveys of the area [3], [13] document the transition from recognition to interactive, generative systems, but point out that there are very few deployed systems capable of providing real-time conversation-based advice.

2.3 Comparison with ClosetAI

In Table 1, ClosetAI is compared to selected previous work. It can be seen from the table that whereas each individual aspect – image input, recommendation, or classification – has been studied in detail, the aspects of live video, multimodal dialogue, occasion understanding, and browser-based zero install have not.

Table 1: Comparison of representative fashion-technology systems with ClosetAI

System	Input	Real-time	Multi-modal	Conver-sation	Occasion
Stitch Fix-type [6]	Form photos	+No	Partial	No	Partial
Try-on mirror [7]	Image	Partial	No	No	No
CNN models [5,10,11,15]	Image (+text)	No	Partial	No	No
DL rec. [1,2,4,14]	Image	No	Partial	No	Partial
ClosetAI	Live video + text	Yes	Yes	Yes	Yes

Here "Real-Time" refers to feedback within a few seconds of the live input, "Multimodal" refers to image and text reasoning in one model call, "Conversation" refers to multi-turn natural language conversations, and "Occasion-Aware" refers to occasion reasoning for the outfit that needs to be worn.

3. System Architecture and Methodology

3.1 High-Level Architecture

ClosetAI uses a client-server approach where the client deals with media capturing and user interface interaction, while the server performs the tasks of authentication, session handling, and AI functionality. All that is executed on the client side in a modern browser without plugins, thus reducing the barrier of entry to the product. With the help of Next.js, the client and server code bases become one and can be deployed at once using the server-side API routes and React client applications residing in one application [2]. The system consists of five logical layers illustrated in Fig. 1:

- **Media Capture Layer** -WebRTC and the MediaDevices API capture real-time audio and video streams from the user's hardware.
- **Streaming Layer** - WebSockets (Socket.IO) maintain a persistent, bidirectional channel carrying sampled frames and chat messages with low overhead.
- **Backend API Layer -** The Node.js router takes care of the authentication, rate limitation, and session handling with Redis used for in-memory state management.
- **AI Processing Layer -** The multimodal Gemini API receives joint image and text data input and gives context-sensitive text-based suggestions.
- **Presentation Layer -** The client-side Next.js application decorated with Tailwind CSS displays a live video stream with the running dialogue.

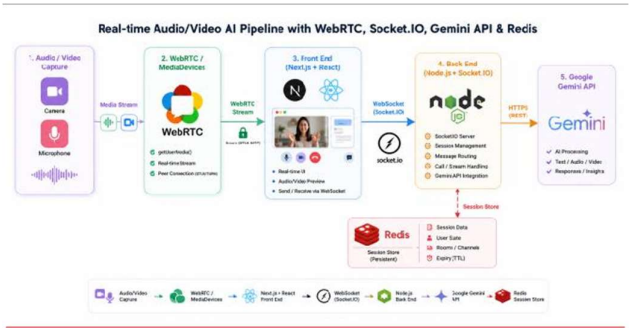

Fig. 1 High-level system architecture of ClosetAI (author-created).

3.2 Technology Stack

3.2.1 Frontend technologies: The UI comprises one-page React application with functional components, static typing via TypeScript, and responsive design via Tailwind CSS.The core component named FashionChat contains three nested components: VideoPreview shows a live camera stream with the indicator for audio activity; MessageThread displays alternating message bubbles with the user's and Assistant's messages in markdown format; InputBar handles queries in text and voice formats via Web Speech API.

3.2.2 Backend and AI integration: The Next.js API routes run on Node.js serve as the proxy between the browser and the Gemini API.The proxy is used to hide the API keys from the client bundle, apply rate limits based on the user's session to control the inference costs, and also include the system prompt that makes the model act like an expert and body-positive fashion assistant.

3.2.3 Gemini API integration: In each inference API call, the payload generated has three sections: one is the text section, which has the user query and the serialized dialogue history; another is the inline-data section, which holds the latest image frame as a base64-encoded JPEG; and the final section is the system instruction, which specifies the persona, domain, and behavioral constraints of the model (such as not allowing any body shaming).

3.3 Interaction Workflow

The process for a typical session is as follows. The client launches the application and allows the use of the camera and the microphone via a browser permission pop-up. The WebRTC stream is created and linked to an HTML video element. A background clock takes one snapshot every N seconds (default value of N=3) by capturing the current frame via the HTML5 canvas and then exporting the snapshot as canvas.toDataURL at JPEG quality 0.7. Upon submission of a query from the user, the client side sends a WebSocket request with the text of the query along with the most recent frame. The server side composes the multimodal input for Gemini, invokes the API, and streams back the results via the same socket.

3.4 Frame Sampling and Prompt Construction

3.4.1 Adaptive frame sampling: However, frame sampling has to find a compromise between visual integrity and processing efficiency.By default, sampling rate and quality is set to $\Delta t = 3$ sec and $q = 0.7$, respectively.Thus, each frame is kept under

approximately 150 kB, and there is enough information to identify colors and pattern from garments on the picture.The algorithm generates an invisible canvas with video size, draws the current frame on it, converts it into base64 string, removes the data-URI header, and transfers the frame.

3.4.2 Multimodal prompt construction: Each prompt is made up of three parts: (1) a system instruction block, which describes the model as an expert and positive about body image in fashion; (2) a serialisation of previous conversation; and (3) the current turn, which includes the user's prompt and the sampled frame.

3.5 Implementation Details for Reproducibility

To support independent reproduction, the implementation parameters are specified below.

3.5.1 Request payload: Every call made to the generateContent endpoint involves an array of contents that has one user turn having two components of [text, inline_data] along with a system_instruction. The component inline_data has mime_type of "image/jpeg" and the base64 frame; the text component has the query and the serialized history [16].

3.5.2 Frame and compression settings: Frames are captured at the negotiated webcam resolution and exported at JPEG quality 0.7, giving payloads below ≈150 kB at a default sampling period of 3 s.

3.5.3 WebSocket events: The channel uses four events: client→server user_message (query plus frame) and session_init; server→client assistant_token (streamed chunks) and assistant_done.Reconnection is handled by Socket.IO with exponential backoff.

3.5.4 Redis session schema: Every session is identified using session:{sessionId}, and holds a JSON data structure containing createdAt, updatedAt, and an array called turns consisting of {role, text, ts}, while the actual frames are never stored in this JSON object. Every key has a configurable lifespan, after which point the session expires.

3.5.5 Security flow: Session validation is carried out using a JWT that is stored in an HTTP-only and same site cookie. The token is validated at every API route before the proxied call to the Gemini endpoint. Inference calls are rate limited per session.

3.5.6 Failure handling: Timeouts on API requests will provide the user with an option to retry, without any automatic retries beyond a maximum number; denial of camera permissions will prevent capturing and guide the user to restore the permission; and frames with an average brightness below a certain threshold value will provide low light warnings.

3.5.7 Frame-deletion verification: Frames are stored only locally as a variable, for one request at a time, and are never stored in the Redis object and on the disk. This claim will be verified empirically in Section 4.5 by analyzing the session store and the logs of the server after the session is complete, ensuring that no image data is left.

4. Evaluation Methodology

This section outlines the entire evaluation process.The measures, tools, and analytical procedures are predetermined in order to enable reporting and replication of the results in relation to them.Values are reported in the results table upon completion of the study; blanks labeled as [TO BE MEASURED] indicate where the values are reported.

4.1 User Study

The user study measures the experience against the baseline. The users (number of users, and age range if they are consented will be mentioned) perform a predefined set of styling tasks in two conditions in a counterbalanced manner:

(i) Live-video conversation on ClosetAI, and (ii) baseline condition, where static image-based fashion assistant takes one uploaded image and the question asked. Post each condition, the users answer questions based on Likert scales of five points with regards to the helpfulness, responsiveness, trustworthiness, and likeness to a human stylist. Means and

4.3 Illustrative Case Studies of Live Interactions

standard deviation values will be mentioned for each item and tested for differences.

4.2 Privacy and Safety Evaluation

4.2.1 Frame-deletion check: After a completed session we inspect the Redis object and server logs to confirm that no frame data is present, verifying the claim that frames are held only transiently in memory.

4.2.2 Bias and safety probes: We use a consistent set of probes that includes various shapes of bodies, colors of skin, cultural and religious attire, gender-neutral clothes, and formal or informal contexts. Each response is then coded by two independent raters on whether there is any bias, stereotyping, or offensive material using the pre-specified

coding scheme, and the ratio of flagged responses within each category is reported. This helps convert the ethical considerations mentioned in the paper to tangible outputs; any high flagging categories are noted as limitations.

In addition to the quantitative protocol defined in Sections 4.1–4.5, a set of live, single-session interactions with the deployed prototype was captured to qualitatively illustrate the categories of reasoning the system performs in practice. These examples are reported as qualitative, illustrative results and do not replace the statistically powered study described above; they are included to give an early, concrete demonstration of the conversational, occasion-aware, and multilingual behaviour referenced throughout this paper.

4.3.1 Occasion-appropriateness reasoning: A test user wearing a casual round-neck T-shirt asked the assistant whether the outfit was appropriate for a wedding ceremony (Fig. 2).

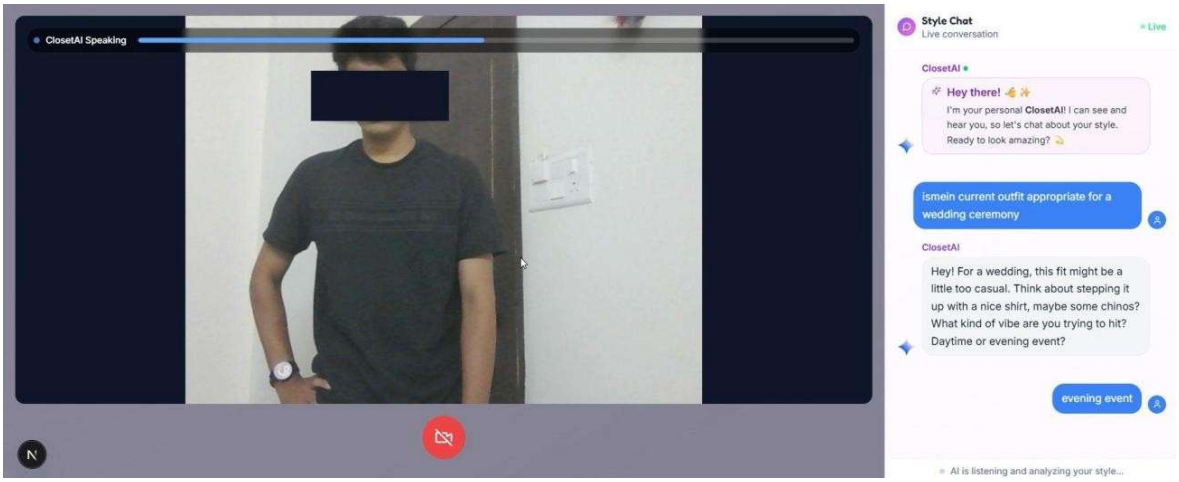

Fig. 2 Live session demonstrating occasion-appropriateness reasoning for a wedding ceremony.

The assistant judged the T-shirt as too informal for the occasion, proposed concrete alternatives (a collared shirt and chinos), and asked a clarifying follow-up question about the time of the event before refining its recommendation. This multi-turn behaviour, in which advice is narrowed using context supplied mid-conversation, illustrates the occasion-aware reasoning and conversational depth claimed as a contribution in Section 1 and distinguishes the system from the one-shot classification baselines discussed in Section 2.2.

4.3.2 Code-mixed query handling and professional-context assessment: In a separate session, a user wearing a graphic-print T-shirt asked, in a code-mixed Hindi-English (Hinglish) query, whether the outfit was suitable for a board meeting or client presentation (Fig. 3).

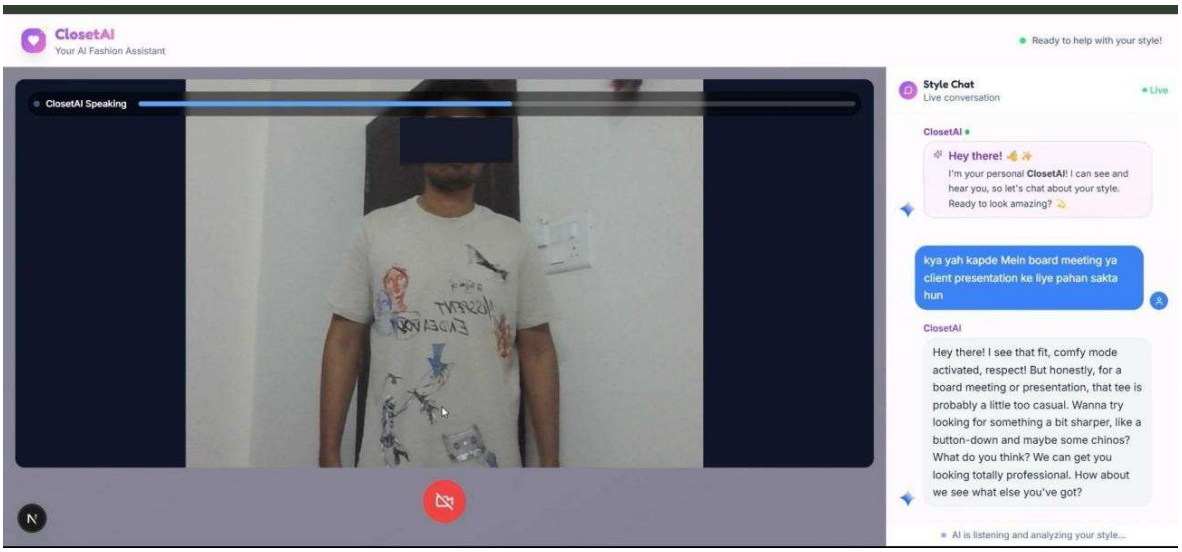

Fig. 3 Live session demonstrating Hinglish query handling and professional-context assessment.

Despite the mixed-language phrasing of the query, the assistant correctly parsed the intent, acknowledged the comfort-oriented styling in a positive tone, and explained why the garment was too casual for a formal business setting, suggesting a button-down shirt and chinos while inviting the user to continue refining the request. This exchange suggests that the natural-language understanding pipeline tolerates the code-mixed input patterns common among Indian users, supporting the accessibility goals discussed in Section 5.1.

4.3.3 Regional-language adaptation: A third session began with feedback on an olive-green shirt for a family gathering and a clarifying question on the formality of the event; the user then asked to continue the conversation in Marathi (Fig. 4).

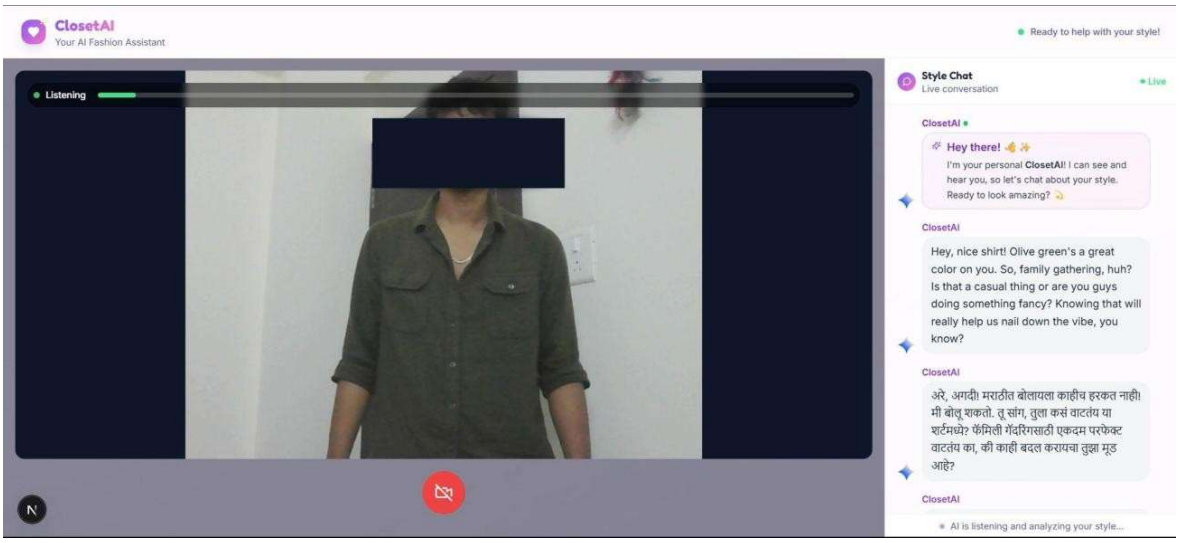

Fig. 4 Live session demonstrating mid-conversation adaptation to Marathi.

The assistant switched to Marathi while preserving the same line of clarifying questions raised earlier in English, without restarting the dialogue or losing conversational context. This case provides early qualitative evidence for the feasibility of the regional-language support extension proposed in Section 5.4.

4.3.4 Activity-based recommendation and language-preference switching: In a fourth session, a user asked, in Hindi, what to wear for a gym workout; after the assistant replied in English asking about workout type and garment

preference, the user stated that Hindi was the only language they understood, and the assistant switched accordingly (Fig. 5).

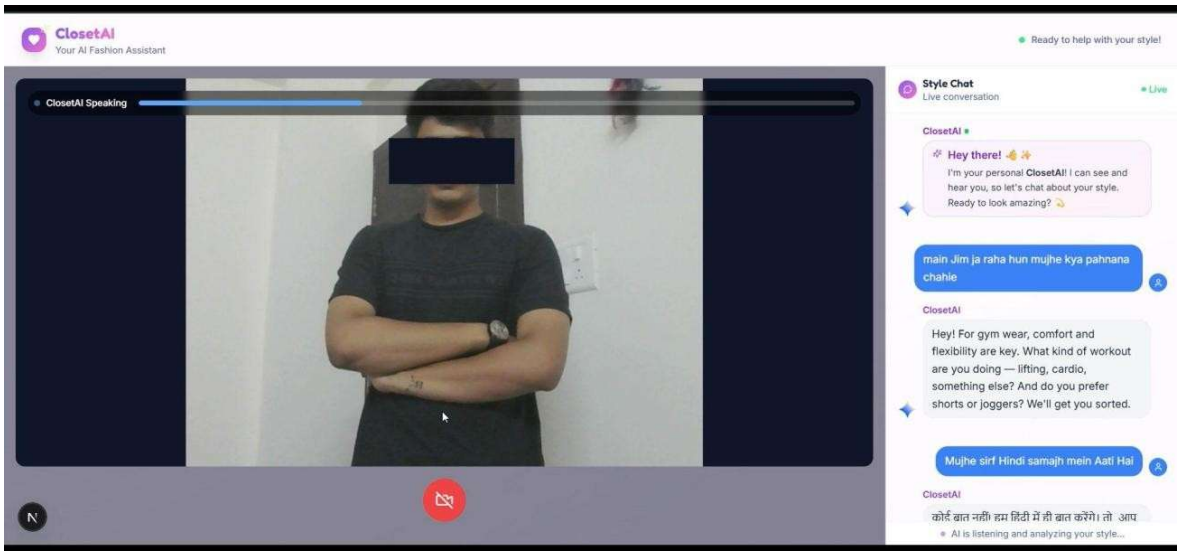

Fig. 5 Live session demonstrating activity-based recommendation reasoning and on-demand switching to Hindi.

Beyond occasion-based formal-wear judgements, this example shows the assistant reasoning about functional, activity-specific clothing requirements (gym attire) and adapting its output language on the basis of an explicit user statement rather than the language of the incoming query alone, indicating a degree of conversational flexibility that goes beyond static, single-turn translation.

5. Discussion, Impact, and Future Scope

5.1 Societal and Environmental Impact

ClosetAI might open up professional styling to disadvantaged sections of society through the reduction in cost and location barriers that comes from remote styling; this would include disadvantaged students, consumers in underdeveloped countries, and those unable to get out due to their condition. In addition, the technology can take on an educational function, teaching individuals how to style themselves and allowing them to form their own tastes. The chances are more that it would take care of the lack of any stylistic service rather than substitute stylists whose expertise in touching and cultural context cannot be replicated by AI at present. As for the environment, it would lead to less pollution through avoiding travel emissions and encourage reuse of existing clothes rather than fast fashion trends and waste.

5.2 Ethical Considerations

Access to the camera is done via the typical permission prompt by the web browser and can be rescinded anytime. The frames are temporarily stored in the memory of the server and will not be stored permanently anywhere, which is better in terms of privacy compared to the approaches where the image of the user gets stored; see Section 4.5 for details about the verification of this point. The prompt for the system should have body-positive language as a constraint to the response. Nevertheless, the tool possesses all known disadvantages of VLMs, including data bias and reliance on API providers.

5.3 Limitations

- *Lighting sensitivity.* Strong fluorescent or backlit conditions reduce colour-prediction accuracy. Client- or server-side brightness normalisation before the API call is a planned mitigation.
- *Inference cost at scale.* A 3s sampling period implies roughly 300 calls per fifteen-minute session per user.

Moving from time-based to event-based sampling would reduce cost substantially.

- *Model bias.* Despite the system prompt, the model may exhibit Western-centric fashion assumptions; evaluation across diverse demographic groups is required and is built into the safety protocol.

5.4 Future Extensions

The planned extensions are linked directly to the limitations and contributions above.

- **Wardrobe management.** Cataloguing clothes so that the system can create outfits by mixing existing clothes in the user's wardrobe, contributing to the sustainability argument raised in Section 5.1.
- **Augmented-reality try-on.** Utilizing WebXR along with lightweight 3D rendering of garments to show suggestions on top of live camera input, tackling the missing try-on issue stated in Section 2.
- **Event-based sampling.** Running inference on detected changes rather than periodically doing it regardless, overcoming the cost issue mentioned in Section 5.3.
- **Multilingual support.** Extending interaction to regional Indian languages (Hindi, Marathi, Tamil, Telugu) and others, broadening the accessibility goal in Section 5.1.
- **Personalised modelling.** Using longitudinal interaction data and explicit consent to generate user-specific lightweight models.

6. Conclusion

The paper has introduced ClosetAI, a multimodal fashion assistant that uses live video through WebRTC along with the vision-language inference of Google's Gemini API to provide outfit suggestions via conversational interaction. The main novelty in this work lies in the architecture itself along with some design patterns – adaptive frame sampling, persona-

based system prompting, streaming token delivery, and a comparison (Table 1) that highlights the differences between ClosetAI and previous recommendation, try-on, and classification systems, which are the use of live video, multimodal conversation, occasion reasoning, and zero-install browser delivery.

Instead of providing performance statistics that are not verified, we provided a protocol that will help us to evaluate the system in terms of latency, recommendation quality, user experience compared to the static image baseline, and also privacy and safety of the system. Once we implement this protocol, we will have an idea about the success rate of the system, and also what are the limitations that the system might be having like sensitivity to lighting conditions, inference cost, and also model bias.

Acknowledgements

The authors thank the Department of Electronics and Telecommunication Engineering, International Institute of Information Technology (I²IT), Pune, for guidance and support during this work.

References

[1] Shirkhani, S., Mokayed, H., Saini, R., et al.: 'Study of AI-driven fashion recommender systems', SN Comput. Sci., 2023, 4, (5),

art. 514

[2] Kalinin, A., Jafari, A.A., Avots, E., et al.: 'AI-driven personalized fashion curation with deep learning', in 'Intelligent Systems and Applications' (Springer, 2024), pp. 1–12

[3] Chen, H.-J., Shuai, H.-H., Cheng, W.-H.: 'A survey of artificial intelligence in fashion', IEEE Signal Process. Mag., 2023, 40, (3), pp. 64–73

[4] Niveditha, A.S., Subha, R., Selvadass, M.: 'AI-based fashion recommendations aligned with current trends from sketch to style', SN Comput. Sci., 2025, 6, (1), art. 21

[5] Arslan, H.S., Sirts, K., Fishel, M., et al.: 'Multimodal sequential fashion attribute prediction', Information, 2019,

10, (10), art.

308

[6] Vyas, S., et al.: 'Enhancing fashion intelligence for exceptional customer experience'. Proc. IEEE Int. Conf. Electronics, Computing and Communication Technologies (CONECCT),

Bangalore, India, July 2024, pp. 1–6

[7] Nguyen-Ngoc, H., et al.: 'Virtual fashion mirror'. Proc. IEEE Int. Conf. Consumer Electronics (ICCE), Las Vegas, USA, January 2021, pp. 1–4

[8] He, J., et al.: 'Visually-aware fashion recommendation and design with generative image models'. Proc. IEEE Int. Conf. Data Mining (ICDM), Barcelona, Spain, December 2016, pp. 207–216

[9] Kalashi, K., Teimourpour, B.: 'Optimizing fashion recommendation systems: a study of deep learning models and distance metrics'. Proc. IEEE Int. Conf. Machine Learning and Applications (ICMLA), Jacksonville, USA, December 2023,

pp. 1–6

[10] Chen, M., et al.: 'Research on clothing image recognition and classification algorithm based on deep learning'. Proc. IEEE Int. Conf. Computer Science and Information Technology, 2024,

pp. 1–6

[11] Pang, S., et al.: 'Analyzing fashion trends using CNN and cloth classification with respect to season and state'. Proc. IEEE Int. Conf. Consumer Electronics-Asia (ICCE-Asia), 2022, pp. 1–4

[12] Kumar, R., et al.: 'Survey of different large language model architectures: trends, benchmarks, and challenges', IEEE Access, 2024, 12, pp. 147037–147061

[13] Patil, A., et al.: 'Toward fashion intelligence in the big data era: state-of-the-art and future prospects', IEEE/CAA J. Autom.

Sinica, 2023, 10, (7), pp. 1515–1542

[14] Mate, N., Nangare, S., Banchhor, C., et al.: 'Personalized AI-based outfit recommendation system', in Shrivastava, V., Bansal, J.C., Panigrahi, B.K. (Eds.): 'Power Engineering and Intelligent Systems (PEIS 2025)', Lecture Notes in Electrical Engineering, vol. 1459 (Springer, Singapore, 2026)

[15] Nakamura, N., et al.: 'Prediction of most confident labelled output for fashion image using deep learning model with TensorFlow'. Proc. IEEE Int. Conf. Electrical, Computer and Communication Technologies (ICECCT), 2024, pp. 1–6

[16] Google: 'Image understanding — Gemini API', https://ai.google.dev/gemini-api/docs/image-understanding, accessed 21 June 2026