ISSN 2277-2685
IJESR/April-June. 2026/ Vol-16/Issue-2/1081-1086
V Bhavana *et. al.,* / International Journal of Engineering & Science Research
Full Length Article

FabriFit - Smart Stitch Advisor

Ms B Tejaswi[1], V Bhavana[2], A Bhavani[3], V Deekshitha[4]
[1]Assistant Professor; Department Of Computer Science And Engineering Bhoj Reddy Engineering College For Women Hyderabad India.
[2,3,4]B.Tech Students; Department Of Computer Science And Engineering Bhoj Reddy Engineering College For Women Hyderabad ,India.
Mail Id's; deekshithavalaboju@gmail.com[4] , bhavanavandanam46@gmail.com[2], bhavaniavula2005@gmail.com[3]
Accepted 13-06-2026

ABSTRACT
The rapid growth of the fashion and textile industry has increased the demand for intelligent systems that assist users in making informed clothing decisions. This project proposes ***FabriFit****, an AI-based mobile application that automatically identifies fabric types from uploaded cloth images and helps users select cost-effective clothing options. The system utilizes* ***Artificial Intelligence (AI)****,* ***Machine Learning (ML)****, and* ***Convolutional Neural Networks (CNNs)*** *to analyze fabric textures and classify them into categories such as cotton, silk, wool, rayon, and other textile materials. AI-driven image analysis provides an efficient method for recognizing textile patterns and attributes, enabling automated fabric identification while reducing the need for manual inspection.*
In addition to fabric classification, the application estimates the required cloth length for stitching using user body measurements such as height and weight. Based on fabric costs and tailoring charges, the system performs a comparative analysis between ready-made garments and custom-stitched clothing, allowing users to identify the most economical option. By integrating ***fabric recognition****,* ***cost estimation****,* ***image classification****, and a user-friendly mobile interface, FabriFit functions as an intelligent fashion assistant. The proposed system enhances customer decision-making, minimizes unnecessary expenditure, and promotes smarter clothing purchases in the modern digital fashion ecosystem.*

Keywords- *Artificial Intelligence (AI), Machine Learning (ML), Convolutional Neural Networks (CNN), Fabric Identification, Image Classification, Textile Analysis, Cost Estimation, Custom Stitching, Ready-Made Garments, Mobile Application, Fashion Technology.*

Introduction

The rapid advancement of artificial intelligence has enabled the development of intelligent solutions in various domains, including the fashion and textile industry. FabriFit is an AI-powered mobile application designed to identify fabric types from uploaded cloth images through image classification techniques. The system employs machine learning algorithms to recognize different textile materials such as cotton, silk, rayon, wool, and other fabrics. In addition to fabric identification, the application estimates the amount of cloth required for stitching garments based on user measurements, including height and weight. Furthermore, the system compares the cost of ready-made garments with custom stitching expenses, allowing users to determine the most economical choice. By integrating machine learning and mobile application technologies, FabriFit serves as a smart fashion assistant that simplifies clothing-related decision-making.

Existing System

Fabric identification is traditionally performed by experienced tailors or textile experts who rely on their knowledge and observation skills. Although some industries utilize artificial intelligence for textile classification, such systems are generally designed for industrial purposes and are not easily accessible to ordinary users. Most existing fashion applications focus on online shopping, outfit recommendations, and style suggestions rather than fabric recognition. Moreover, current solutions do not provide facilities for estimating cloth requirements based on body measurements. The absence of integrated cost comparison features between ready-made garments and custom stitching further limits their usefulness for consumers.

Proposed System

The proposed FabriFit system introduces an intelligent and automated solution for fabric identification and clothing cost analysis. The application uses artificial intelligence techniques to analyze images uploaded by users and accurately determine the corresponding fabric type. Based on user measurements such as height and weight, the system estimates the amount of fabric required for stitching garments. It also performs a comparative analysis between the cost of ready-made clothing and custom stitching expenses, helping users select the most cost-effective option. Designed with a simple and user-friendly interface, the application can be conveniently used by both customers and tailoring professionals. The overall objective of the system is to support informed decision-making while reducing unnecessary expenditure on clothing.

Requirement Analysis

Requirement analysis is an important stage in software development that helps define the functions and performance expectations of the proposed system. The requirements of FabriFit are categorized into functional requirements, non-functional requirements, and computational resource requirements.

Computational Resource Requirements

Hardware Requirements

The minimum hardware configuration required for developing and deploying the application includes an Intel Core i5 processor, 8 GB RAM, and a storage capacity of at least 500 GB. These resources are sufficient to support machine learning model execution, database operations, and application development activities.

Software Requirements

The application development environment consists of Windows 11 as the operating system and Visual Studio Code as the integrated development environment. The user interface is developed using HTML and CSS, while JavaScript is utilized for client-side functionality. The backend is implemented using Python frameworks such as Django or Flask. MySQL Server is used for database management, and Firebase services provide authentication, cloud storage, and real-time database support. Machine learning functionalities are implemented using TensorFlow Lite and Google Teachable Machine.

Life Cycle Model

The development of FabriFit follows the Waterfall Software Development Life Cycle model, which adopts a systematic and sequential approach to software development. The model consists of multiple phases that are completed one after another, ensuring proper documentation and process control throughout the project.

The first phase, Requirement Analysis, involves gathering user needs and documenting both functional and non-functional requirements. During the System Design phase, the architecture of the application, database structures, user interface components, and machine learning integration strategies are planned. The Implementation phase focuses on coding and integrating various modules of the application. Once development is completed, the system undergoes rigorous Testing to verify functionality, performance, and reliability while identifying and correcting potential errors. The Deployment phase involves hosting and releasing the application for user access. Finally, the Maintenance phase ensures continuous monitoring, bug fixing, system updates, and the introduction of future enhancements to improve functionality and user experience.

The Waterfall model is suitable for the proposed project because it provides a clear structure, straightforward management, and systematic progression through all stages of software development. Although modifications become more challenging after development begins, the model remains effective when project requirements are clearly defined and stable.

Design

System Architecture

The proposed FabricFit system is designed as an intelligent fashion analysis platform that combines mobile computing, deep learning, and cloud-based services to identify clothing categories and fabric types from images. The architecture consists of three major layers: the mobile application layer, the AI inference layer, and the data management layer.

The mobile application enables users to capture or upload garment images through a simple interface. Once an image is selected, it is transmitted to the backend server where image preprocessing and prediction tasks are executed. A trained deep learning model analyzes the image and generates predictions related to garment style and fabric composition. The prediction results are then returned to the mobile application and displayed to the user in real time.

Software Architecture

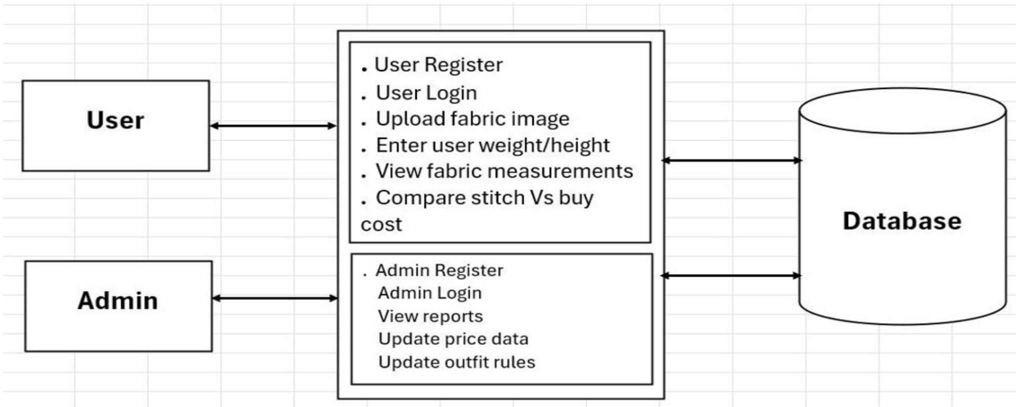


Fig 1 Software architecture

The software architecture follows a client-server model. The frontend mobile application manages image acquisition, user interaction, and result visualization. The backend server, developed using FastAPI, handles request processing, image preprocessing, and communication with the trained machine learning model. The prediction engine utilizes a multitask learning framework capable of simultaneously identifying garment and fabric categories.

Technical Architecture

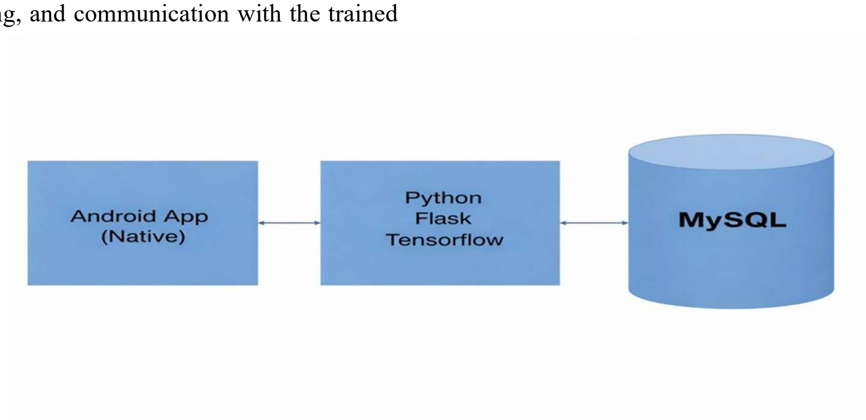


Fig 2 Technical architecture

The technical architecture integrates multiple technologies to support efficient system operation. The mobile application communicates with the FastAPI server through RESTful APIs. The server hosts a TensorFlow-based deep learning model trained using MobileNetV2. User information and system records are maintained in a MySQL database, while image processing is performed using Python image-processing libraries. This layered architecture ensures scalability, reliability, and efficient response times.

Implementation

The FabricFit platform was implemented using modern web and machine learning technologies to provide accurate garment analysis and an intuitive user experience. The system integrates frontend technologies, backend services, database management, and deep learning frameworks.

Frontend Development

The user interface was developed using HTML, CSS, and JavaScript. These technologies provide responsive layouts, interactive controls, and seamless navigation across different devices.

HTML

HTML serves as the structural foundation of the application. It organizes interface components such as image upload forms, prediction displays, navigation menus, and user interaction elements. Semantic HTML elements were utilized to improve accessibility and maintainability.

CSS

CSS was employed to create a visually consistent and responsive interface. Layout design,

ISSN 2277-2685
IJESR/April-June. 2026/ Vol-16/Issue-2/1081-1086
V Bhavana *et. al.,* / International Journal of Engineering & Science Research

typography, spacing, and adaptive screen rendering were managed through cascading style sheets, ensuring usability across mobile and desktop environments.

JavaScript

JavaScript enables dynamic interaction within the application. Client-side validation, image preview functionality, asynchronous communication with backend services, and real-time result updates were implemented using JavaScript to enhance user experience.

AI Model Implementation

The core intelligence of the FabricFit platform is provided by a deep learning model developed using TensorFlow and Keras. MobileNetV2 was selected as the base architecture due to its lightweight structure and strong performance in image classification tasks.

The model employs a multitask learning strategy where shared feature representations extracted from garment images are simultaneously used for clothing type classification and fabric type identification. Transfer learning was applied by utilizing pretrained ImageNet weights, reducing training time while improving generalization performance.

During inference, uploaded images undergo preprocessing, including resizing, normalization, and batch dimension generation. The processed image is then fed into the trained model, which outputs probability distributions for both garment and fabric classes. The categories with the highest confidence scores are returned as prediction results.

Backend Implementation

The backend system was developed using FastAPI, a high-performance Python web framework. The server exposes RESTful endpoints for image upload and prediction requests. Incoming images are processed using image manipulation libraries before being passed to the TensorFlow model. Prediction results are formatted as JSON responses and transmitted back to the mobile application.

Database Management

MySQL was used to store user information, prediction records, and application-related data. The database design supports efficient retrieval and management of historical prediction results, enabling future analytics and personalization features.

Testing

Testing was conducted to evaluate system functionality, reliability, and prediction accuracy. Multiple testing levels were performed to ensure the successful operation of all modules.

Unit Testing

Unit testing focused on individual components such as image upload functionality, preprocessing operations, API request handling, database connectivity, and prediction result generation. Each module was tested independently to verify correct behavior under normal and exceptional conditions.

Integration Testing

Integration testing examined interactions between the frontend application, backend server, database system, and machine learning model. The objective was to ensure seamless communication and accurate data exchange across all system components.

System Testing

System testing evaluated the complete FabricFit platform as an integrated solution. Real-world garment images were used to assess overall functionality, response time, prediction consistency, and user experience. The system successfully processed image uploads and generated accurate classification results within acceptable execution times.

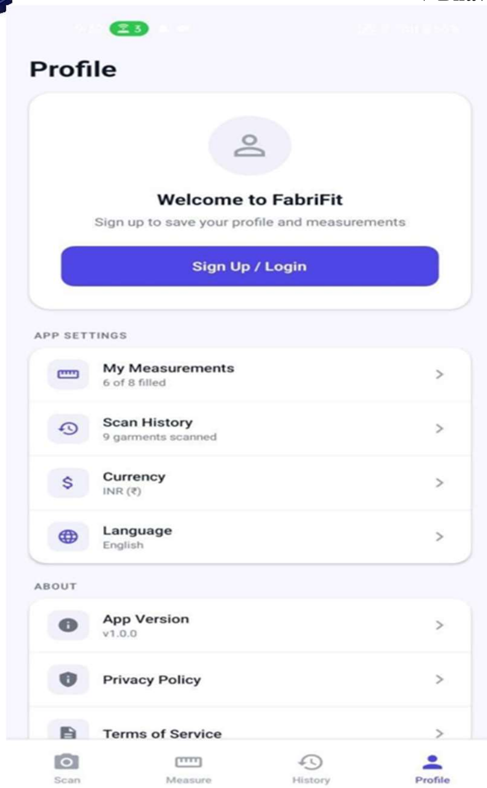
User Acceptance Testing

User acceptance testing was conducted to determine whether the application satisfied end-user expectations. Participants evaluated system usability, interface design, and prediction quality. Feedback indicated that the platform was intuitive to use and provided meaningful clothing and fabric identification results.

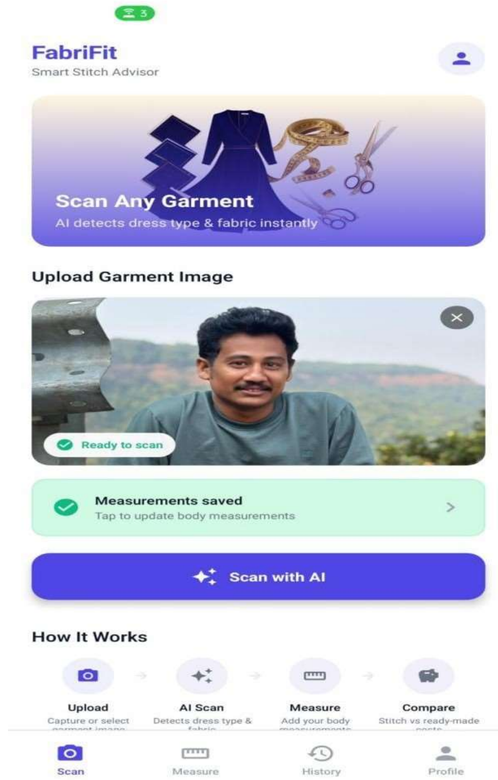
Testing Outcomes

The testing process confirmed that the FabricFit platform operates reliably under expected usage conditions. Functional tests verified successful user registration, authentication, image upload, prediction generation, result display, and administrative operations. Validation tests also confirmed the system's ability to handle incorrect inputs and missing data without failure.

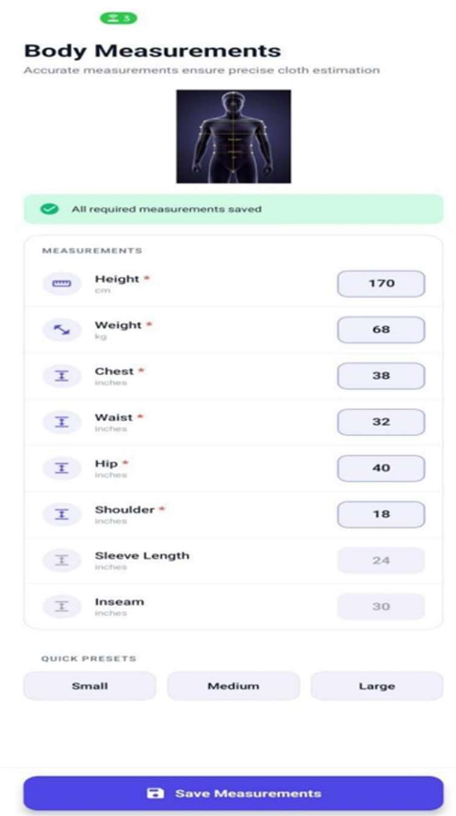
Screenshots



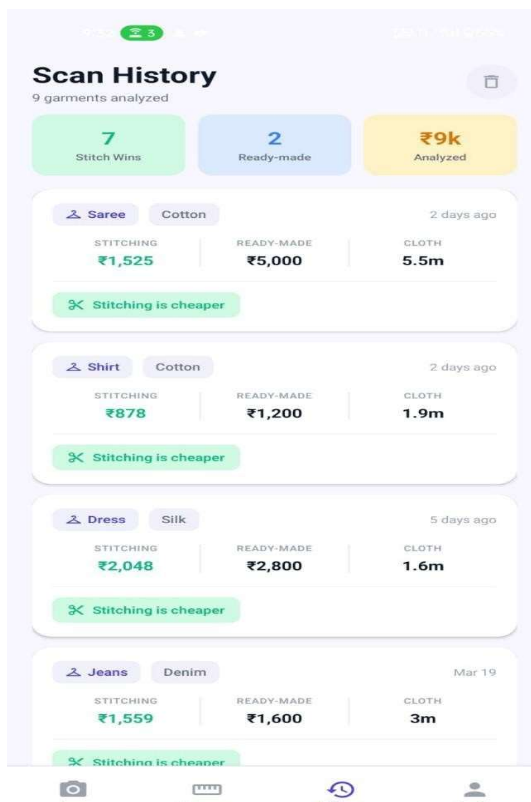
Screenshot 1 : Login page



Screenshot 2 : Upload outfit



Screenshot 3 User measurements



Screenshot 4 : Scan history

Conclusion

The FabricFit system demonstrates the effective integration of artificial intelligence and web technologies for fabric identification and clothing cost analysis. The developed solution utilizes a Convolutional Neural Network (CNN) model to classify fabric types from user-uploaded images and provides automated cost estimation based on fabric price and garment measurements. By comparing tailoring expenses with ready-made garment prices, the system assists users in making informed and cost-effective purchasing decisions. The implementation confirms the feasibility of applying machine learning techniques to practical fashion-related applications while maintaining a user-friendly and efficient workflow.

Future Scope

Several enhancements can further improve the performance and usability of the FabricFit platform. Expanding the training dataset with a wider variety of fabric samples would improve classification accuracy and model robustness. Future versions may incorporate real-time fabric recognition through smartphone cameras, enabling instant analysis without image uploads. The system can also be extended to provide personalized clothing suggestions based on user preferences, fabric characteristics, and current fashion trends. Integration with e-commerce platforms would allow users to explore and purchase recommended fabrics or garments directly through the application. Furthermore, the adoption of advanced deep learning architectures and intelligent recommendation techniques could enhance fabric quality assessment, prediction reliability, and cost estimation accuracy.

References

[1] M. A. I. Hussain, B. Khan, Z. Wang, and S. Ding, "Woven Fabric Pattern Recognition and Classification Based on Deep Convolutional Neural Networks," *Electronics*, vol. 9, no. 6, pp. 1–18, 2020.

[2] S. Das, S. Sundaramurthy, and S. Jayaram, "Deep Learning Convolutional Neural Network for Defect Identification and Classification in Woven Fabric," *Indian Journal of Artificial Intelligence and Neural Networking*, vol. 4, no. 2, pp. 45–53, 2020.

[3] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[4] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2019.

[5] A. Q. Ohi, M. F. Mridha, M. A. Hamid, and M. M. Monowar, "FabricNet: A Fiber Recognition Architecture Using Ensemble Convolutional Neural Networks," *IEEE Access*, vol. 9, pp. 128654–128666, 2021.

[6] Kaggle, "Convolutional Neural Networks with TensorFlow," 2023. [Online]. Available: https://www.kaggle.com

[7] TensorFlow, "Image Classification Using Convolutional Neural Networks," 2024. [Online]. Available: https://www.tensorflow.org/tutorials/images/classification