IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AI-POWERED SMART WARDROBE AND PERSONALIZED OUTFIT RECOMMENDATION SYSTEM USING DEEP LEARNING

Mr. Akashraj S, Mr. Hariprasath S, Mr. Nithin S, Mr. Mohamed Aazil S M, Ms.Akileshwari

[1,2,3,4] B.Tech AI&DS Final Year, [5]Associate Professor

[1,2,3,4,5] Department of Artificial Intelligence and Data Science,

[1,2,3,4,5] Rathinam Technical Campus, Coimbatore, India

Abstract

The growth of digital fashion platforms has increased the need for intelligent and personalized styling systems. This paper presents an AI-powered Smart Wardrobe that combines deep learning-based clothing classification with a hybrid recommendation engine. Users can upload garment images and receive outfit suggestions based on body type, skin tone, and wear history. A transfer learning approach using ResNet50 is applied for clothing classification. The model achieved 70.2% validation accuracy and 68.7% real-world test accuracy. The system enhances wardrobe utilization, minimizes outfit repetition, and promotes sustainable fashion through automated personalization.

Keywords: *Smart Wardrobe, Deep Learning, ResNet50, CLIP, Visual Embeddings, Personalized Recommendation, Computer Vision*

1. Introduction

Artificial Intelligence has significantly transformed various industries, including healthcare, finance, and e-commerce. The fashion industry has recently adopted AI-driven approaches to enhance customer experience and personalization. Despite the presence of online recommendation systems, most platforms provide suggestions based primarily on browsing history or purchasing trends rather than individual wardrobe data.

Users often struggle with daily outfit selection due to lack of styling knowledge, poor color coordination, and underutilization of existing clothing. Moreover, conventional wardrobe applications require manual organization and lack intelligent automation. Therefore, there is a need for an AI-based system that can digitally manage wardrobe collections and generate personalized outfit recommendations using image-based analysis.

This research proposes a Smart Wardrobe System that integrates deep learning-based clothing classification with a rule-enhanced recommendation engine to provide context-aware outfit suggestions.

2. Literature Review

Previous research has explored various fashion recommendation techniques.

Liu Z et al. introduced the DeepFashion dataset, enabling large-scale clothing recognition using CNN architectures.

He R et al. proposed FashionNet, which focused on personalized outfit compatibility modelling.

Factorization-based recommendation approaches were introduced by Steffen Rendle using factorization machines to model user-item interactions.

Context-aware recommendation systems were discussed by Gediminas Adomavicius, emphasizing contextual personalization.

However, most existing systems:

- Do not analyse personal wardrobe images
- Lack repetition control
- Ignore body metrics and skin tone
- Focus primarily on trend-based suggestions

The proposed system addresses these limitations through image classification, compatibility filtering, and adaptive ranking.

Table 1: Literature Comparison

Author	Year	Focus Area	Limitations	Our Improvement
Liu Z et al.	2016	CNN-based clothing classification	No outfit recommendation	Adds personalized recommendation engine
X.Hang	2017	Outfit compatibility modelling	No wardrobe digitization	Includes digital wardrobe & wear tracking
Steffen Rendle	2010	Collaborative recommendation	No image-based analysis	Uses CNN-based visual feature extraction
Gediminas Adomavicius	2022	Context-aware recommendation	Not fashion-specific	Applies body type & skin tone context

3. Problem Statement

Existing wardrobe management systems lack intelligent automation and deep personalization. They depend on manual tagging and do not integrate visual similarity detection or body-type-based recommendation logic. Furthermore, outfit repetition is not controlled effectively, leading to inefficient wardrobe utilization. This research aims to design an AI-driven system capable of automatically classifying clothing items and generating optimized outfit combinations tailored to user attributes.

4. Proposed System

The proposed AI-Powered Smart Wardrobe System is designed to provide intelligent wardrobe digitization and personalized outfit recommendations by integrating deep learning, computer vision, and rule-based filtering mechanisms. Unlike traditional wardrobe applications that rely on manual tagging, the proposed system automates clothing classification and recommendation generation through AI-driven analysis.

4.1 Image Classification Module

The proposed Smart Wardrobe System consists of four primary layers:

- **Frontend Layer:** User interaction and image upload interface
- **Backend API Layer:** FastAPI-based communication and processing
- **AI Processing Layer:** CNN-based clothing classification
- **Database Layer:** Structured storage using Supabase PostgreSQL

This modular design ensures scalability, maintainability, and efficient real-time inference.

4.2 Clothing Classification Module

The classification module employs a transfer learning approach using a pre-trained ResNet50 architecture. The final fully connected layer is fine-tuned to classify garments into predefined categories such as tops, bottoms, jackets, and dresses.

Input images are:

- Resized to 224 × 224 pixels
- Normalized according to ImageNet standards
- Passed through convolutional layers for feature extraction

The extracted feature representations are then processed through a softmax layer for category prediction.

4.3 Visual Embedding and Similarity Analysis

In addition to classification, the system utilizes CLIP-based embeddings to capture semantic relationships between clothing items. These embeddings allow the system to:

- Measure similarity between garments
- Detect repetitive outfit patterns
- Improve compatibility scoring

This hybrid approach enhances recommendation accuracy beyond basic category matching.

4.4 Personalized Recommendation Engine

The recommendation engine generates outfit combinations using:

- Classified wardrobe items
- User profile attributes (body type, skin tone)
- Wear frequency history
- Color harmony rules

Each possible outfit pair is assigned a compatibility score based on rule-based filtering and embedding similarity. The highest-ranked combinations are presented to the user.

4.5 Workflow Process

The operational workflow includes:

1. User uploads clothing image
2. Image is preprocessed and classified
3. Metadata is stored in database
4. Recommendation engine retrieves wardrobe data
5. Compatibility scoring is applied

6. Top-ranked outfits are displayed

5. Methodology

The proposed system combines deep learning-based image classification with embedding-based similarity analysis and rule-driven recommendation logic. The methodology consists of dataset preparation, model training, feature extraction, scoring, and evaluation.

5.1 Dataset Preparation

The system is trained using the DeepFashion dataset, which contains 2,90,000 annotated clothing images across multiple categories.

Data Preprocessing Steps:

- Image resizing to 224 × 224 pixels
- Pixel normalization using ImageNet mean and standard deviation
- Data augmentation (rotation, flipping, brightness variation)
- Train–Validation–Test split (70:15:15)

Data augmentation improves model generalization and reduces overfitting.

5.2 Model Architecture

The classification module is built using transfer learning with ResNet50.

Architecture Flow:

Input Image → Convolution Layers → Feature Maps → Global Average Pooling → Fully Connected Layer → Softmax Output

Let:

- X= Input image
- $f(X)$= Extracted feature vector
- W= Learned weights
- y= Predicted class

Classification function:

$$\boldsymbol{y = Softmax(W \cdot f(X))}$$

Cross-Entropy Loss is used as the objective function:

$$\boldsymbol{L = -\sum y_i \log(\hat{y}_i)}$$

The model is optimized using the Adam optimizer.

5.3 CLIP-Based Embedding Extraction

To enhance recommendation quality, the system uses CLIP embeddings to capture semantic similarity between clothing items.

Let:

- E_i= Embedding of item i
- E_j= Embedding of item j

Cosine similarity is calculated as:

$$Similarity(i,j) = \frac{E_i \cdot E_j}{\| E_i \| \| E_j \|}$$

Higher similarity values indicate visually compatible items.

This mechanism helps in:

- Avoiding repetitive outfit combinations
- Improving compatibility ranking
- Enhancing personalization

5.4 Recommendation Scoring Function

Each outfit combination is assigned a compatibility score based on:

- Category compatibility
- Color harmony
- Embedding similarity
- Wear history penalty

Final scoring equation:

$$Score = \alpha C + \beta S - \gamma R$$

Where:

- C= Category compatibility score
- S= Similarity score
- R= Repetition penalty
- α, β, γ= Weight parameters

The outfit with the highest score is recommended.

6. Implementation

6.1 Hardware

- Laptop (i5/i7, 8GB RAM minimum)
- Optional: NVIDIA GPU (4GB+ VRAM) for training

6.2 Software

- Python 3.11
- PyTorch (for ResNet50 model training)
- OpenAI CLIP (for visual embeddings)
- FastAPI (backend API framework)
- PostgreSQL / Supabase (database)
- Supabase Storage (image storage)
- NumPy, Pandas, Matplotlib

- OpenCV (image preprocessing)
- React /ShadCN (frontend)

6.3 Development Steps

1. Collect dataset (DeepFashion + custom test images).
2. Preprocess data (resize, normalize, augmentation).
3. Train ResNet50 model (GPU-based training).
4. Extract CLIP embeddings for similarity analysis.
5. Build recommendation scoring logic.
6. Develop FastAPI backend for inference.
7. Integrate database and cloud storage.
8. Test system with multiple user scenarios.
9. Deploy frontend interface for user interaction.

7.Results and Analysis

The performance of the proposed Smart Wardrobe System was evaluated in two stages: clothing classification accuracy and overall recommendation effectiveness.

7.1 Classification Results

The ResNet50-based classification model was trained using the DeepFashion dataset and evaluated using validation and test sets.

Performance Metrics:

- Training Accuracy: 78.5%
- Validation Accuracy: **70.3%**
- Test Accuracy: 68–71%

The model demonstrates moderate performance in multi-class clothing classification. The reduced accuracy compared to benchmark datasets is primarily due to:

- Visual similarity between clothing categories
- Complex backgrounds in user-uploaded images
- Lighting variations
- Real-world data noise

Despite moderate classification accuracy, the model successfully identifies major garment categories such as tops, bottoms, jackets, and dresses with acceptable reliability.

7.2 Recommendation Performance

The recommendation engine combines:

- Classification output
- CLIP-based similarity embeddings
- Body-type compatibility rules
- Wear-history filtering

Experimental testing across multiple user scenarios showed:

- Reduced repetitive outfit suggestions
- Improved outfit coordination
- Better personalization compared to manual selection

The integration of embedding similarity significantly enhanced outfit compatibility ranking.

Advantages:

- Reduces daily outfit decision time.
- Minimizes wardrobe underutilization.
- Prevents repetitive dressing patterns.

Limitations:

- Limited Context Awareness
- No Virtual Try-On Capability
- Dataset Dependency

8. Conclusion

This research presented an AI-driven Smart Wardrobe and Personalized Outfit Recommendation System that integrates deep learning and compatibility-based ranking. The proposed approach automates wardrobe management and enhances daily outfit decision-making. The experimental results demonstrate stable classification performance suitable for real-world wardrobe applications. The system contributes to sustainable fashion practices by improving wardrobe utilization and reducing unnecessary purchases.

9. Future Scope

- Virtual Try-On System using AR visualization.
- Reinforcement Learning Personalization for adaptive styling.
- Weather & Calendar Integration for contextual outfit planning.
- E-commerce API Integration for intelligent purchase suggestions.
- Social Fashion Community Platform for outfit sharing and feedback.
- Mobile Application Deployment for real-time usage

10. References (IEEE Format)

[1] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, “DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 1096–1104.

[2] R. He, C. Packer, and J. McAuley, “Learning Compatibility Across Categories for Heterogeneous Item Recommendation,” in *Proc. ACM Int. Conf. Multimedia (ACM MM)*, 2017, pp. 937–945.

[3] S. Rendle, “Factorization Machines,” in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, 2010, pp. 995–1000.

[4] G. Adomavicius and A. Tuzhilin, “Context-Aware Recommender Systems,” *ACM Trans. Inf. Syst.*, vol. 23, no. 1, pp. 103–145, 2011.

[5] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.

[6] A. Radford et al., “Learning Transferable Visual Models From Natural Language Supervision,” in *Proc. Int. Conf. Machine Learning (ICML)*, 2021, pp. 8748–8763.

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.

[8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 4510–4520.

[9] M. Tan and Q. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in *Proc. Int. Conf. Machine Learning (ICML)*, 2019, pp. 6105–6114.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification With Deep Convolutional Neural Networks,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2012, pp. 1097–1105.

[12] D. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” in *Proc. Int. Conf. Learning Representations (ICLR)*, 2015.

[13] Y. LeCun, Y. Bengio, and G. Hinton, “Deep Learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[14] X. Han, Z. Wu, Y.-G. Jiang, and L. S. Davis, “Learning Fashion Compatibility With Bidirectional LSTMs,” in *Proc. ACM Int. Conf. Multimedia*, 2017, pp. 1078–1086.