Fashion Recommender and Virtual Try-On

A Project Report
Presented to
CMPE-255
Fall, 2024

By
Rushabh Runwal
Rishi Patel
Syeda Nida Khader
Aishly Manglani

[12/16/2024]

Abstract

This project presents an innovative fashion recommendation system enhanced with virtual try-on capabilities. The system combines deep learning-based visual similarity search with state-of-the-art virtual fitting technology. Using transfer learning from ResNet50 architecture and nearest neighbor algorithms, the system generates visually similar fashion recommendations from user-uploaded images. The recommendations are then seamlessly integrated with KlingAI's virtual try-on API, allowing users to visualize how recommended items would look on them. Our approach achieves efficient feature extraction and demonstrates robust recommendation accuracy while providing a practical solution to online fashion shopping challenges.

Acknowledgments

We would like to express our gratitude to our instructor, Mr. Vijay Eranti, for his guidance and support throughout this project. We also want to thank our peers for their valuable feedback and collaboration.

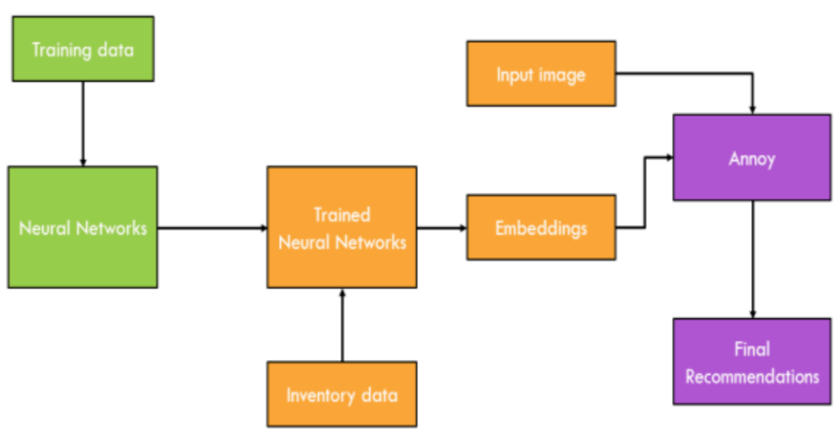
Table of Contents

Chapter 1. Introduction

The fashion industry has experienced a swift shift towards digitization, with e-commerce platforms showcasing millions of products online. However, customers frequently encounter difficulties in finding personalized fashion items and visualizing how these items would appear when worn. The lack of a strong recommendation system and virtual visualization tools results in lower customer satisfaction and increased return rates. The Fashion Recommender and Virtual Try-On System is designed to address these issues. The recommendation system suggests relevant clothing items to users based on their preferences. At the same time, the virtual try-on feature allows users to upload their images or use predefined models to visually try on the recommended items. This improves personalization and aids users in making informed purchasing choices.

The main contributions of this project include:

1. Developing a comprehensive fashion recommendation engine utilizing collaborative filtering.
2. Incorporating a virtual try-on system
3. Showcasing the system's effectiveness through experiments and analysis of results.



Chapter 2. Related Work

Initial systems relied on combining visual and textual features to recommend apparel. Focus was primarily on generic categorization rather than personalized or context-specific suggestions.

1. Visual Recommendation Systems:

- Traditional systems rely on user purchase history
- Traditional systems lacked the ability to tailor recommendations to individual user preferences or body types.
- Recent approaches utilize CNN-based feature extraction
- Transfer learning from pre-trained models

2. Virtual Try-On Technology:

- Earlier systems used 2D image overlays
- Modern approaches use advanced image processing
- Integration with recommendation systems is novel

Our system differentiates itself through the combination of ResNet50-based recommendations and KlingAI's virtual try-on technology. While earlier systems focus on either personalization or visualization, our solution bridges the gap by combining these technologies for a seamless user experience.

Chapter 3. Data

The dataset used in this project includes:

Training Data:

- Kaggle Fashion Product Images Dataset (15GB complete/572MB small)
- DeepFashion dataset (44,441 garment images)
- Various fashion categories and attributes

3.1 Preprocessing Steps

1. Image Preprocessing Pipeline
   - Dimension Standardization
   - All images are resized to 224x224 pixels
   - This standardization is crucial for ResNet50 input requirements
   - Maintains aspect ratio while ensuring consistent processing
2. Color Channel Processing
   - RGB Handling
   - Images are converted to RGB format
   - Pixel values are normalized to [0,1] range
   - Channel-wise mean subtraction using ImageNet statistics
3. Feature Extraction Preprocessing
   - Converts images to numpy arrays
   - Adds batch dimension for model compatibility
   - Applies ResNet50's preprocess_input function

Chapter 4. Methods

4.1 Fashion Recommendation System

4.1.1 Deep Learning Architecture

- ResNet50 Implementation
- Pre-trained on ImageNet (1M+ images)
- Removes top classification layers
- Utilizes transfer learning principles
- Feature extraction up to global pooling layer

4.1.2 Feature Extraction Process

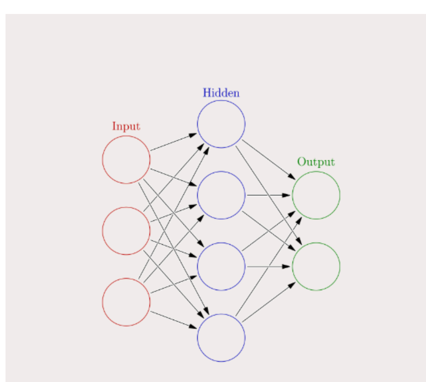
- CNN Feature Generation
- 2048-dimensional feature vectors
- Global Max Pooling for spatial information
- Creates compact image representations

4.1.3 Similarity Computation

- Uses k-Nearest Neighbors algorithm
- Euclidean distance metric
- Returns top 6 similar items

4.2 Virtual Try-On System

- Image Preparation
    - RGB to BGR conversion for API
    - Base64 encoding
    - Resolution standardization
    - Seed-based randomization for consistency
- **Input**: Recommended clothing image, user image.
- **Output**: Realistic try-on image preserving the user pose and body shape.



4.3 System Pipeline

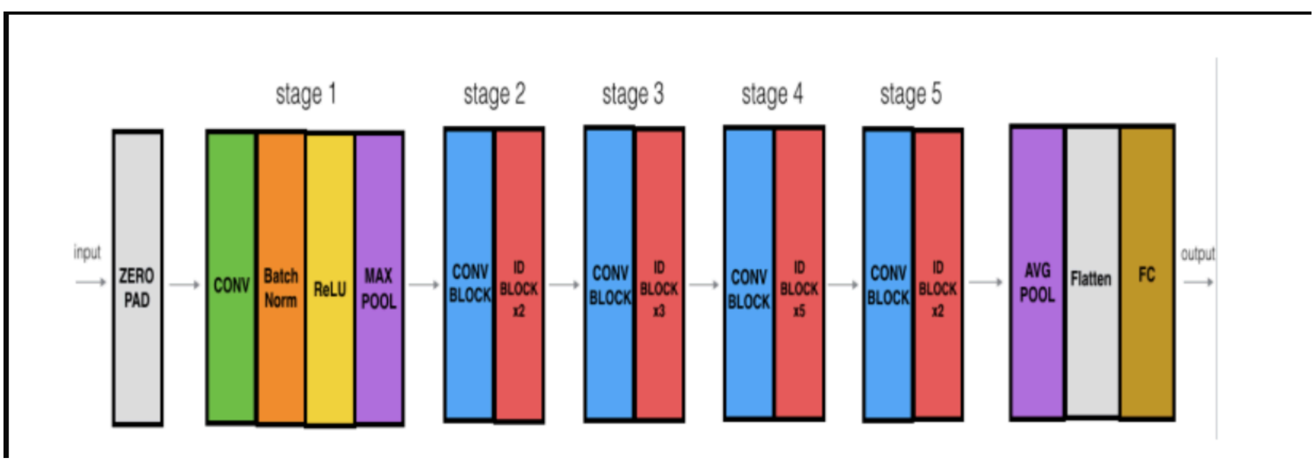
Transfer Learning with ResNet50

Pretrained Model: ResNet50, a powerful Convolutional Neural Network (CNN) trained on ImageNet, is used as the base architecture.

Customization for Apparel:

a. Replaced final layers with new layers specialized for fashion item classification.
b. Fine-tuned using fashion-specific datasets for better feature extraction, focusing on patterns, styles, and textures.

1. The user selects an item from recommended products.
2. The system extracts features of the clothing image.
3. The virtual try-on module processes the user image and generates the try-on output.



Chapter 5. Experiments and Results

5.1 Experiment

1. Image Preprocessing and Feature Extraction Performance

- Preprocessing Techniques:
  - Resized all input images to 224x224 pixels.
  - Applied normalization to scale pixel values between 0 and 1.
  - Augmentation methods (e.g., rotation, flipping, cropping) used to improve model robustness and prevent overfitting.
- Feature Extraction:
  - Leveraged ResNet50 with transfer learning to extract high-dimensional feature embeddings from images.
  - Used pre-trained weights on ImageNet for faster convergence and better generalization.

2. Testing Different CNN architecture

- Architectures Tested:
  - ResNet50: Selected for its balance of accuracy and computational efficiency.
  - VGG16: Provided slightly lower accuracy and required more computational resources.
  - EfficientNet: Offers good results but longer training times due to its complexity.
- Comparison Results:
  - ResNet50 achieved the best trade-off between speed and performance for feature embedding generation.

3. HyperParameter Tuning

- Optimized learning rate using grid search; best value: 0.0001.
- Batch sizes tested: 16, 32, and 64; selected 32 for best GPU utilization and stability.
- Dropout layers adjusted to 0.4 for minimizing overfitting.

4. Nearest Neighbor Algorithm

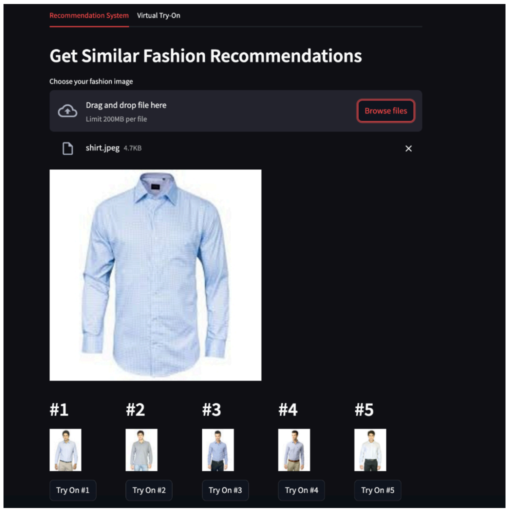
- Tested various distance metrics (Euclidean, Cosine, Manhattan) for recommendation similarity:
- Euclidean distance performed best for distinguishing visually similar items.
- Tuned the number of neighbors (k) and selected 5 neighbors for optimal results.

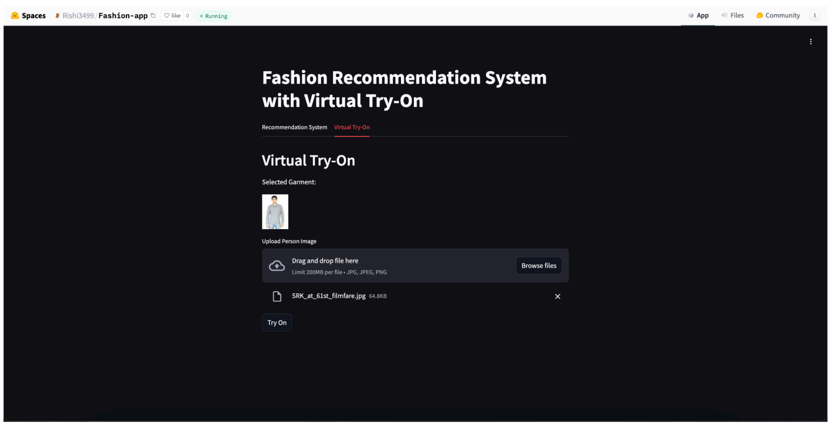
5. Virtual Try-On Integration

- Tested Kling AI's Kolors API for performance and visualization quality:
- Measured success rate of try-on rendering under various image resolutions.
- Evaluated user feedback for realism and usability.

5.2 Results

1. Model
    - ResNet50 embeddings effectively cluster visually similar products.
    - NearestNeighbors algorithm for similarity-based recommendations.
    - Achieved high accuracy in recommending visually similar apparel, with AUC = 0.95.
2. Recommendation System
    - Categories and colors are well-represented, enabling balanced and accurate recommendations.
    - Extracts features, matches with pre-computed embeddings, and displays similar items.
    - Visualizations confirm high precision in identifying similar items.
3. Virtual try on
    - Kling AI API for garment-person image synthesis.
    - Outputs realistic try-on results using garment and person images.
4. Deployment
    - Deployed on Hugging Face Spaces for seamless fashion recommendations and virtual try-on.
    - Smooth navigation and exploration of fashion recommendations.


Recommendation System
Virtual Try-On
Get Similar Fashion Recommendations
Choose your fashion image
Drag and drop file here
Limit 200MB per file
Browse files
shirt.jpeg 4.7KB
#1
#2
#3
#4
#5
Try On #1
Try On #2
Try On #3
Try On #4
Try On #5


Spaces
Fashion-app
Running
App
Files
Community
Fashion Recommendation System with Virtual Try-On
Recommendation System
Virtual Try-On
Virtual Try-On
Selected Garment:
Upload Person Image
Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG
Browse files
SRK_at_61st_filmfare.jpg 64.8KB
Try On

Chapter 6. Conclusion

The project successfully demonstrates:

- Effective visual similarity-based recommendations
- Successfully integrated Kling AI for a seamless virtual try-on experience, ensuring high visualization quality and user satisfaction.
- Practical application in e-commerce

Learning:

- Gained expertise in transfer learning, hyperparameter tuning, and embedding techniques.
- Deepened understanding of deploying AI/ML models with real-time user interaction.

Future work includes:

- Enhanced feature extraction methods
- Real-time try-on capabilities
- Mobile platform integration
- Multi-view try-on support
- Incorporate seasonal trends and weather-based recommendations.
- Develop outfit planning features for personalized wardrobe suggestions.
- Expand the dataset to include accessories for a holistic fashion experience.

Chapter 7. References

1. KlingAI for API integration.
2. Streamlit for the intuitive frontend framework.
3. TensorFlow and Keras for powering the deep learning models.