Bachelor of Technology

Project Presentation

Department of Computer Engineering
Sardar Patel Institute of Technology
(Autonomous Institute Affiliated to University of Mumbai)
Munshi Nagar, Andheri(W), Mumbai-400058
2020-21

A PRESENTATION ON

“Pocket Fashionista - A Complexion based Outfit Color Advisor using Neural Networks”

By

Tejashri Wagh - 2018230077
Anisha Gharat - 2018230071
Siddesh Sonawane - 2017130059

Under the guidance of

Prof. Reeta Koshy

Problem Statement

- People usually find it difficult to get the best clothing color combinations that suit their skin tone well and go well with the existing fashion trends.
- There exists a lot of confusion among the users as what outfits to buy that will suit their personality, be a good fit for a certain weather and so on.
- So the aim is to develop a complexion based clothing color recommendation system that will help to choose the best possible clothes color combinations.
- The system will should recommend the most suitable outfits and their color combinations.

Problem Statement

- It will also allow the users to virtually visualize how they will look in the recommended color combinations.
- Also recommendations can be made based on the ongoing weather conditions or seasons as well.
- The application will allow users to make best choices with their clothes color combinations and thus saving their time and energy in even trying out the clothes.
- Easy for merchants to master the real-time demand of consumers.

Objectives

- To study current consumer trends and identify target demographics.
- To provide the most suitable color combination.
- To provide personalized outfit recommendations.
- To minimise time and energy required to select among options and try it on mobile screen.
- To provide a personalised experience in relation to various events and variable factors like weather.

Proposed Solution

1. **Skin Tone Detection**

- **HSV and YCbCr Color Space -** These are two different color model used to determine the image pixel and separate defined threshold pixel. This we used for skin detection. It will separate skin and non-skin pixels in a given image.
- **K-Means Clustering Algorithm -** This is unsupervised learning algorithm used to cluster group of data points with similar features in the given data set. We used it to cluster pixel data based on their threshold values.
- **openCV -** for image processing.

Proposed Solution

2. Similar Outfits Recommendations

- **Feidegger dataset -** composed of dress images and related textual descriptions of 8732 high-resolution images.
- Transfer Learning is used for content based recommendations.
- We utilize the pre-trained VGG16 model to extract relevant features from our dress images and build a similarity score on them.
- We pass a dress image to our system, we compute the similarity with all our dresses stored in ‘train’ and then we select the most similar (with the highest similarity scores).

Proposed Solution

3. Virtual - Try On

- The live video stream is captured on the user side using OpenCV.
- Then all clothes to be tried on are read.
- The person in the video stream is detected using Object detection.
- Then the HoughCircles formula is used for estimating the size of the clothes for further resizing.
- The user can move back or forward to check out different clothes.
- If the user wants to save a screenshot then he/she can also do so.

Proposed Solution

4. Weather based recommendations

- **DeepFashion dataset -** 280K fashion images across 46 categories.
- Seasons considered - Winter, Summer, Spring, Autumn.
- For recommendations, Convolutional Neural Networks(CNN) with transfer learning from ResNet and approximate nearest neighbors algorithm is used.
- We then pass a set of season-wise images to the model for generating similar recommendations.
- Centroid embeddings will then merge all the seed images into one representation by averaging the values across all dimensions.
- Finally, Approximate nearest neighbors is applied to return outfits closest to the representation.

Methodology Adapted

1. **Skin Tone Detection**

- Implemented using Python, numpy, matplotlib, sklearn, etc.
- openCV is used to read the given images.
- First we detect skin by using HSV & YCbCr color space.
- RGB image is converted into HSV and YCbCr value
- HSV (Hue, Saturation, Value) and YCbCr (Luminance, Chrominance) color models.
- We use this two models combined to improve recognition of skin pixel.
- The HSV and YCbCr value of each pixel compare to standard values of skin pixel.

Methodology Adapted

1. **Skin Tone Detection**

- The ranges for a skin pixel used in this algorithm are as follows:

0 <= H <= 17 and 15 <= S <= 170 and 0 <= V <= 255

and

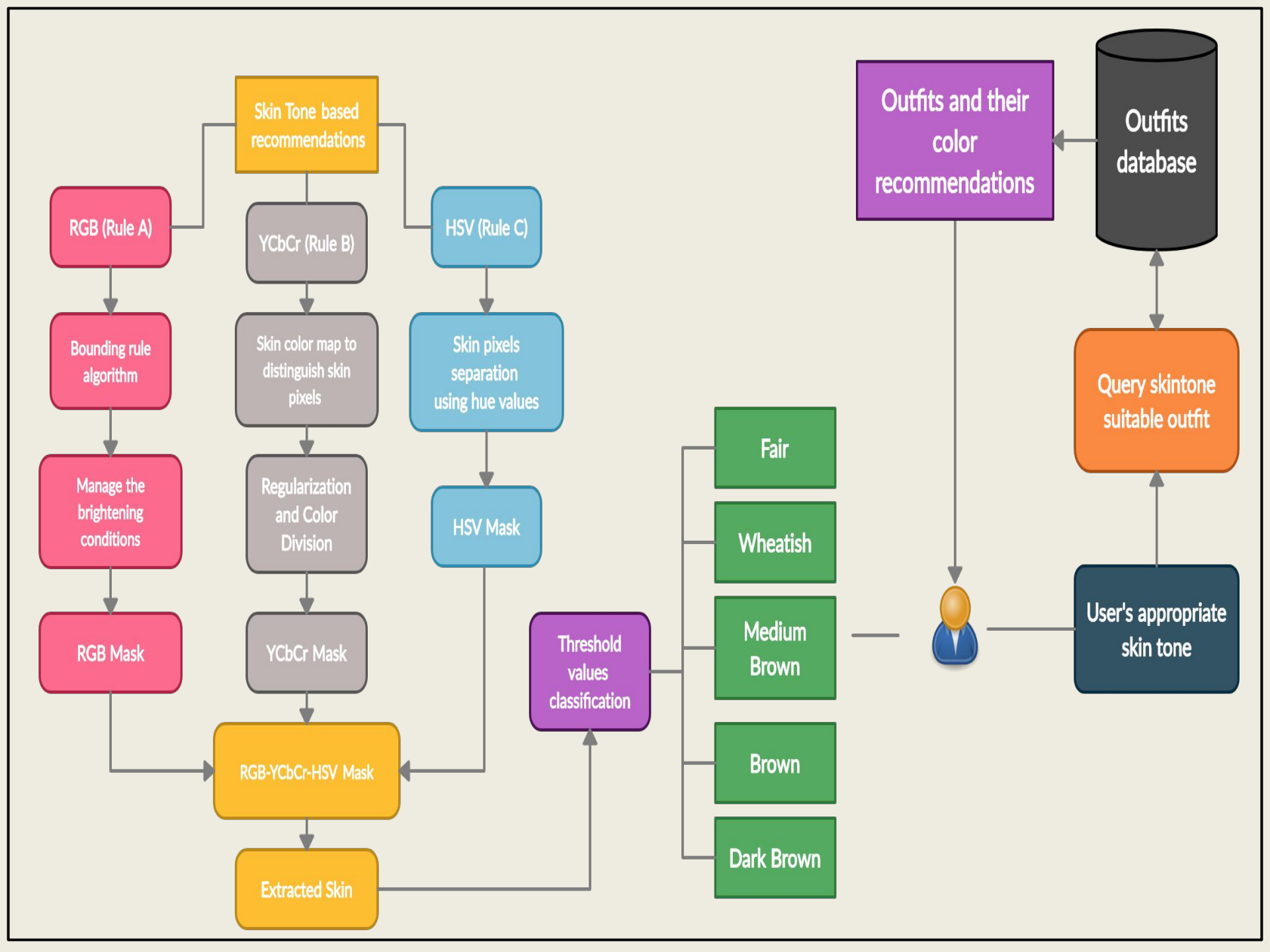
0 <= Y <= 255 and 135 <= Cr <= 180 and 85 <= Cb <= 135

- Find the Dominant Colors - We used the K-Mean Clustering Algorithm. ***KMeans Clustering*** is used to cluster the pixel data based on their threshold values.
- Further skin tone will be classified by particular threshold values.
- Defined five skin tones - Fair, Wheatish, Medium Brown, Brown, Dark Brown.

Methodology Adapted

2. Outfit Color Recommendation

- Skin tone classification module pass the tone to this module.
- Based on that we defined specific outfit colors to particular skin tone.
- Used **“men-formal-shirts.csv”** dataset.
- In the dataset we considered attributes like “Description & Color” to get the color name from this column.
- Accordingly outfits will be shown to the user by getting images from the dataset and using matplotlib images will be plotted.


Skin Tone based recommendations
RGB (Rule A)
YCbCr (Rule B)
HSV (Rule C)
Bounding rule algorithm
Skin color map to distinguish skin pixels
Skin pixels separation using hue values
Manage the brightening conditions
Regularization and Color Division
HSV Mask
RGB Mask
YCbCr Mask
RGB-YCbCr-HSV Mask
Extracted Skin
Threshold values classification
Fair
Wheatish
Medium Brown
Brown
Dark Brown
Outfits and their color recommendations
Outfits database
Query skintone suitable outfit
User's appropriate skin tone

Methodology Adapted

3. Similar Outfits Recommendations

- This is a Content based recommendation system wherein we use Transfer learning.

- The pretrained VGG16 model is used here to extract the relevant features from our dress images and build a similarity score on them.

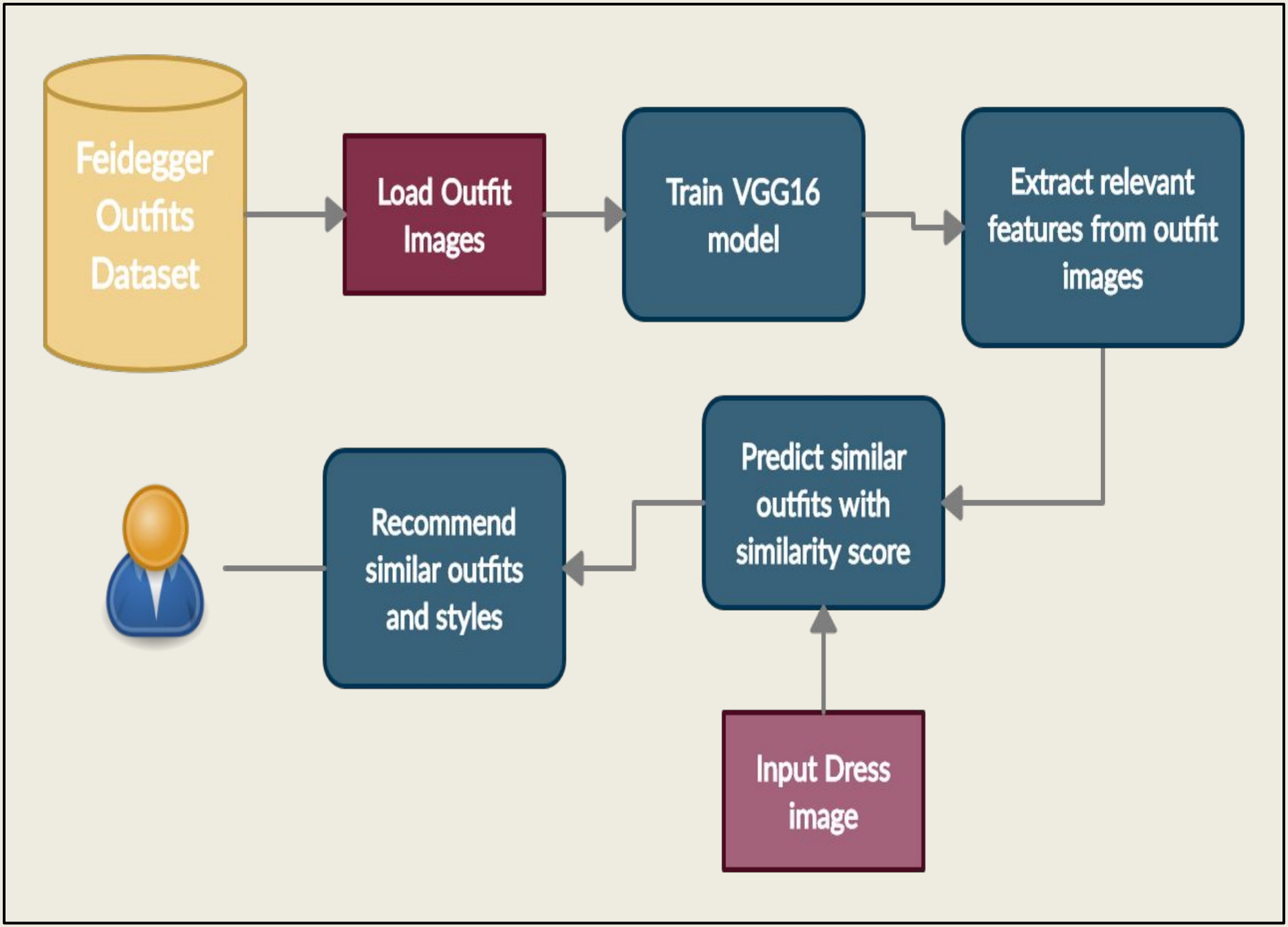
- We ‘cut’ the VGG at the second-last layer, so we obtain for every single image a vector of dimension 1x4096.

- The train test split is of 80-20%. The training data is used for building a similarity score matrix.

Methodology Adapted

3. Similar Outfits Recommendations

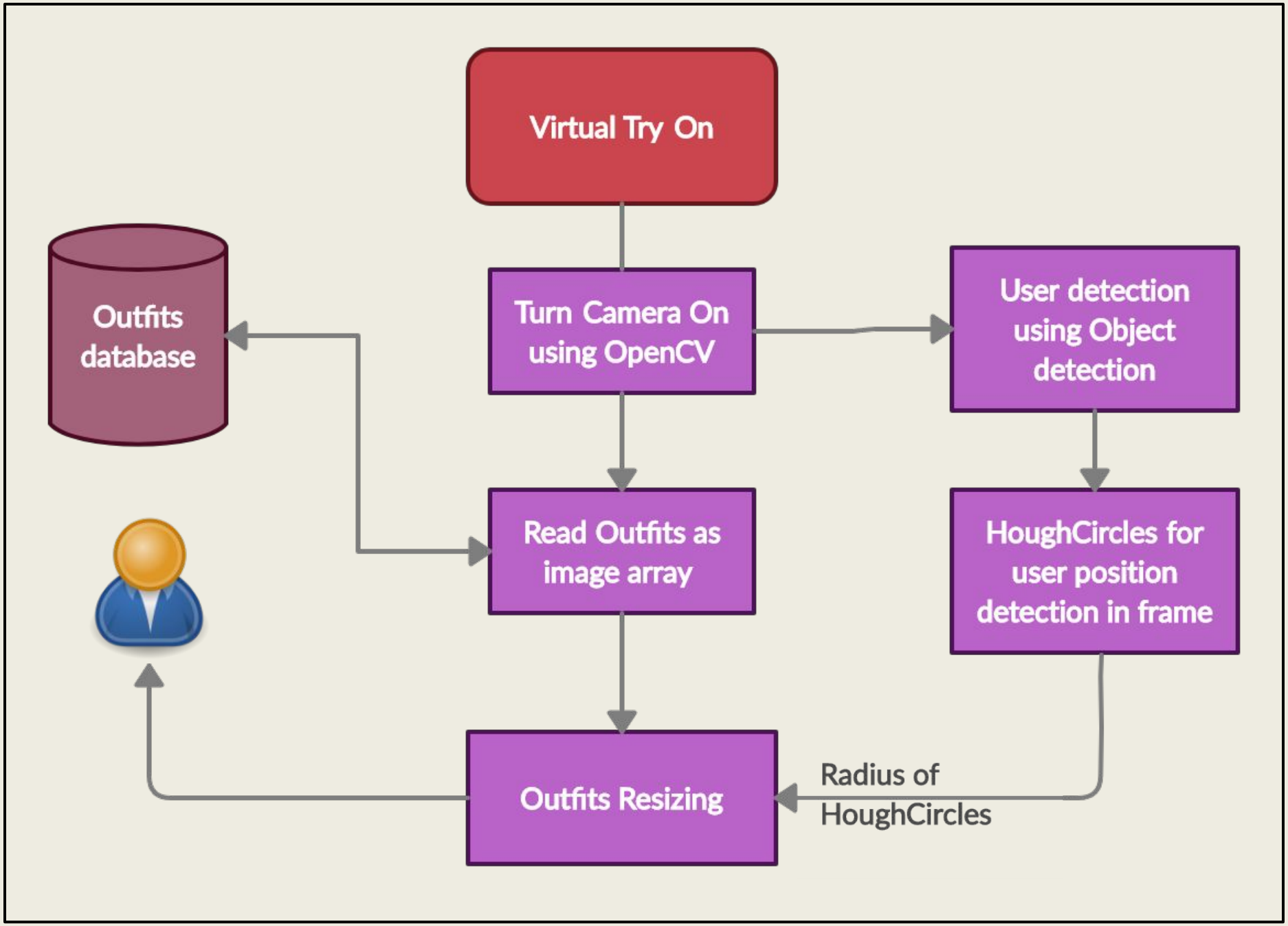
- Then we make the prediction on data by using the CNN models.
- Then we stored the model in a pickle format.
- We then compute the similarity matrix for the other clothes in the dataset.
- Finally we plot the most similar outfits as per the user's input of clothes as per his choice.


Feidegger Outfits Dataset
Load Outfit Images
Train VGG16 model
Extract relevant features from outfit images
Predict similar outfits with similarity score
Recommend similar outfits and styles
Input Dress image

Methodology Adapted

4. Virtual - Try On

- We have used the OpenCV library of Python for capturing the live video stream.
- From the images that the user can try on, we are resizing them to fit on the user's body.
- We have used HoughCircles() to determine the area in which the user is standing in front of the frame.
- According to the obtained measurements of the radius, we then use the resize() function of the imutils library to resize the clothes.
- Resizing takes place by increase/decrease of either the height/width of the clothes.


Virtual Try On
Outfits database
Turn Camera On using OpenCV
User detection using Object detection
Read Outfits as image array
HoughCircles for user position detection in frame
Outfits Resizing
Radius of HoughCircles

Methodology Adapted

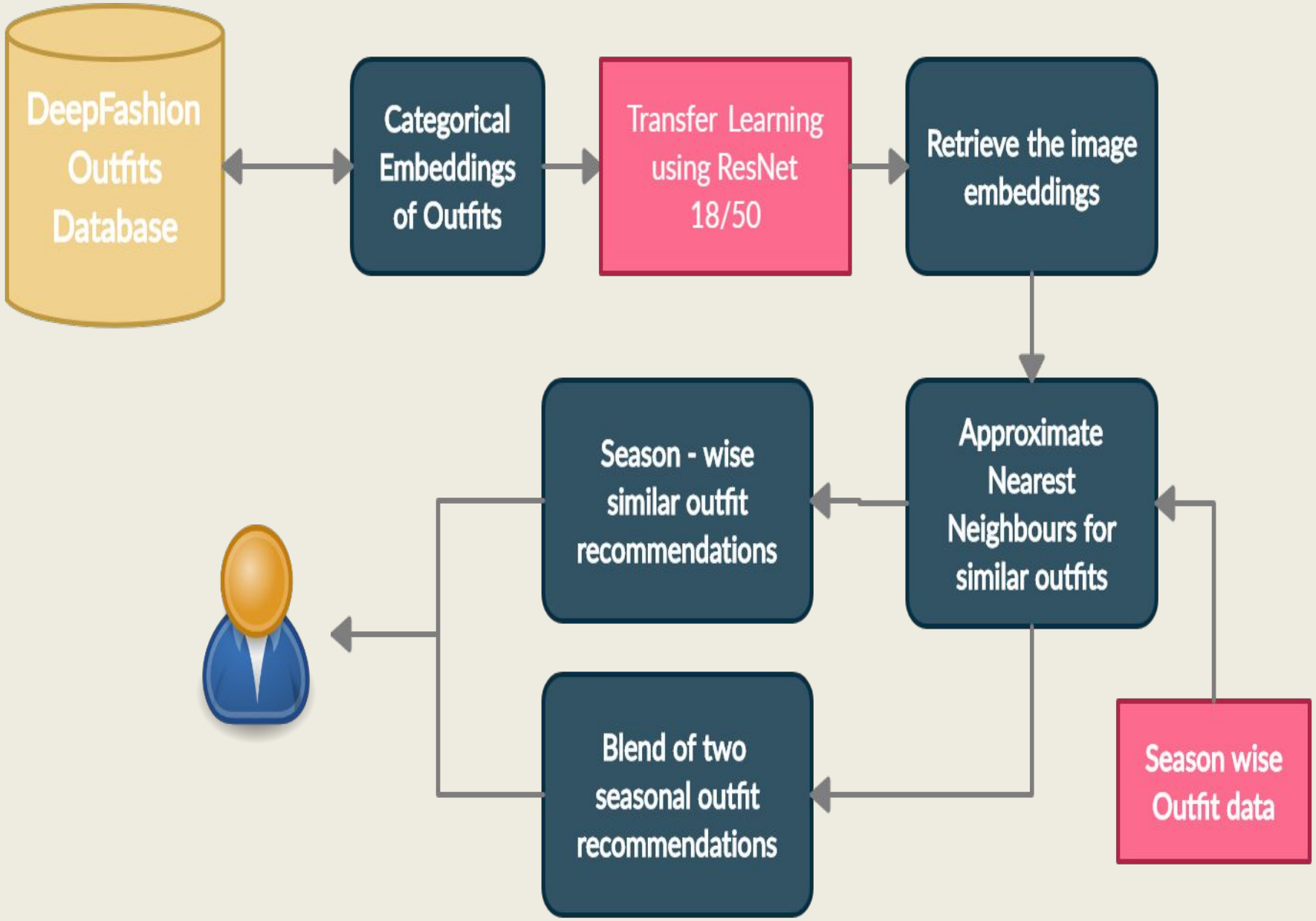
5. Weather based recommendations

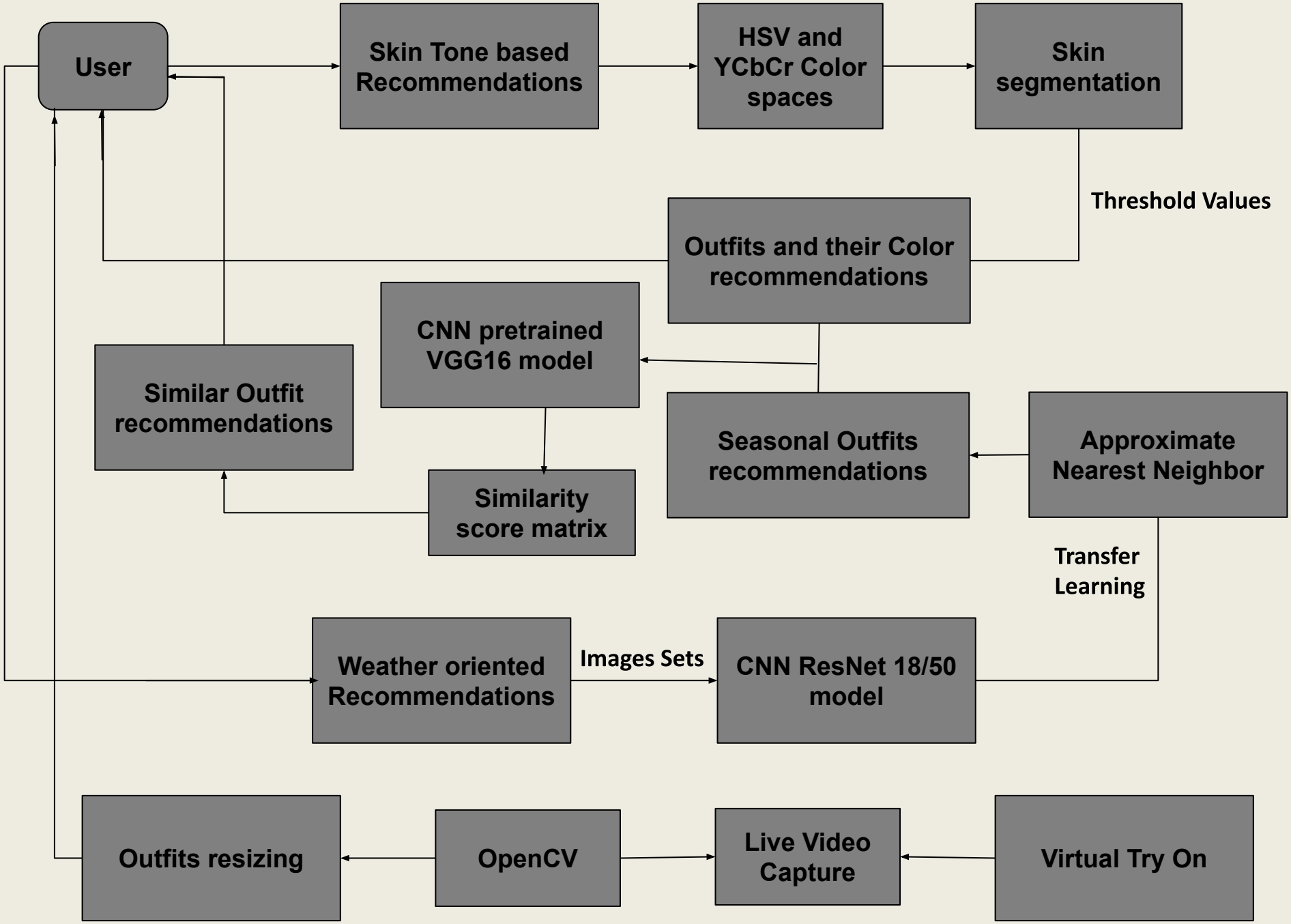
- We have used the DeepFashion dataset for giving recommendations for four seasons - Spring, Autumn, Summer, Winter.
- We selected 24 outfits for each season. They will act as the seed images for Seasonal Collections.
- We only pass images into the model; the model does not ingest additional attributes/descriptions/product details; this is because we want the model to automatically learn and detect the style of fashion images passed into it without further human/machine labeling.
- Seasonal recommendations have been built on Convolutional Neural Networks, with transfer learning from ResNet and approximate nearest neighbors.

Methodology Adapted

5. Weather based recommendations

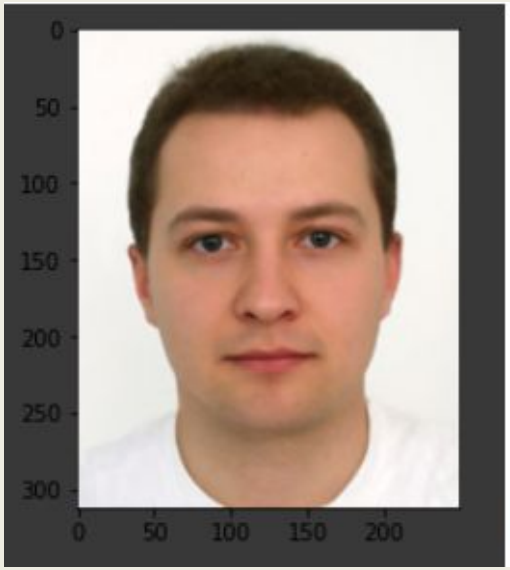
- After the train-test split, we extracted the different categories of clothes that the dataset consists of.
- Later, we converted the images into these categorical embeddings.
- We used the Resnet 18/50 layers for training the CNN model.
- Then we saved the model to import it conveniently later.
- We finally used Approximate Nearest Neighbors algorithm for generating the similar clothing recommendations.
- Centroid embeddings in ANN will then merge all the seed images into one representation by averaging the values across all dimensions.
- Thus the group of similar images as per the season passed are recommended to the user.

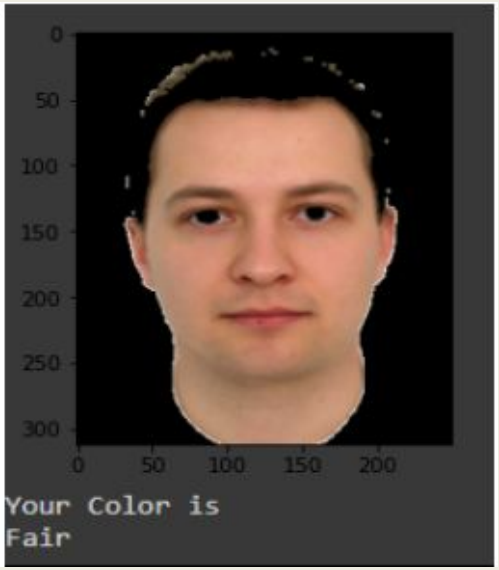

DeepFashion Outfits Database
Categorical Embeddings of Outfits
Transfer Learning using ResNet 18/50
Retrieve the image embeddings
Approximate Nearest Neighbours for similar outfits
Season - wise similar outfit recommendations
Blend of two seasonal outfit recommendations
Season wise Outfit data

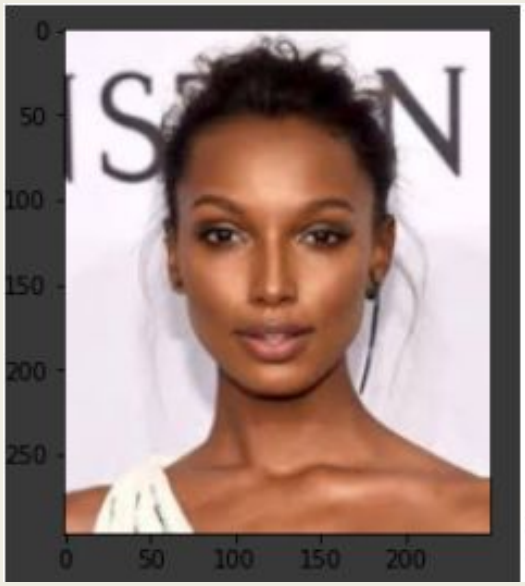

User
Skin Tone based Recommendations
HSV and YCbCr Color spaces
Skin segmentation
Threshold Values
Outfits and their Color recommendations
CNN pretrained VGG16 model
Similar Outfit recommendations
Seasonal Outfits recommendations
Approximate Nearest Neighbor
Similarity score matrix
Transfer Learning
Weather oriented Recommendations
Images Sets
CNN ResNet 18/50 model
Outfits resizing
OpenCV
Live Video Capture
Virtual Try On

Results

1. **Skin Tone Detection**

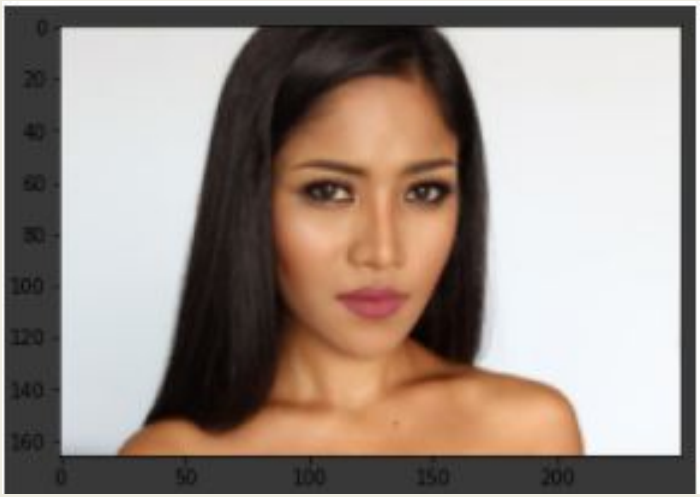


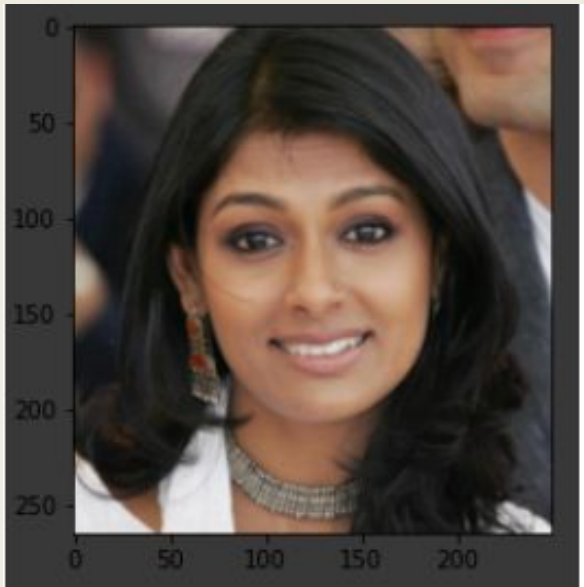


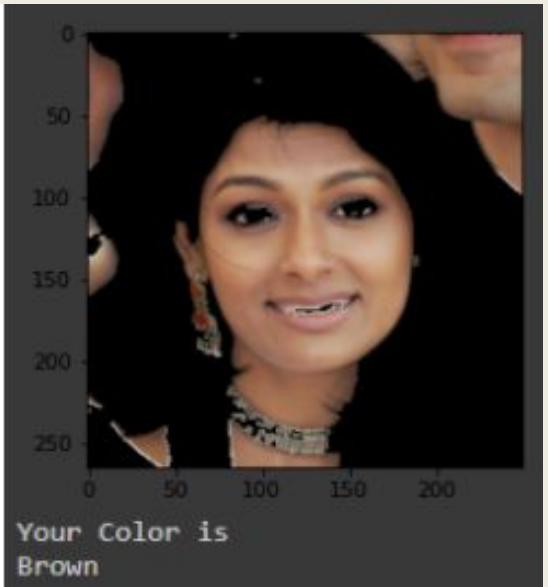


Results

1. Skin Tone Detection







Results

2. Outfit Color Recommendation

Results

3. Similar Outfit Recommendation

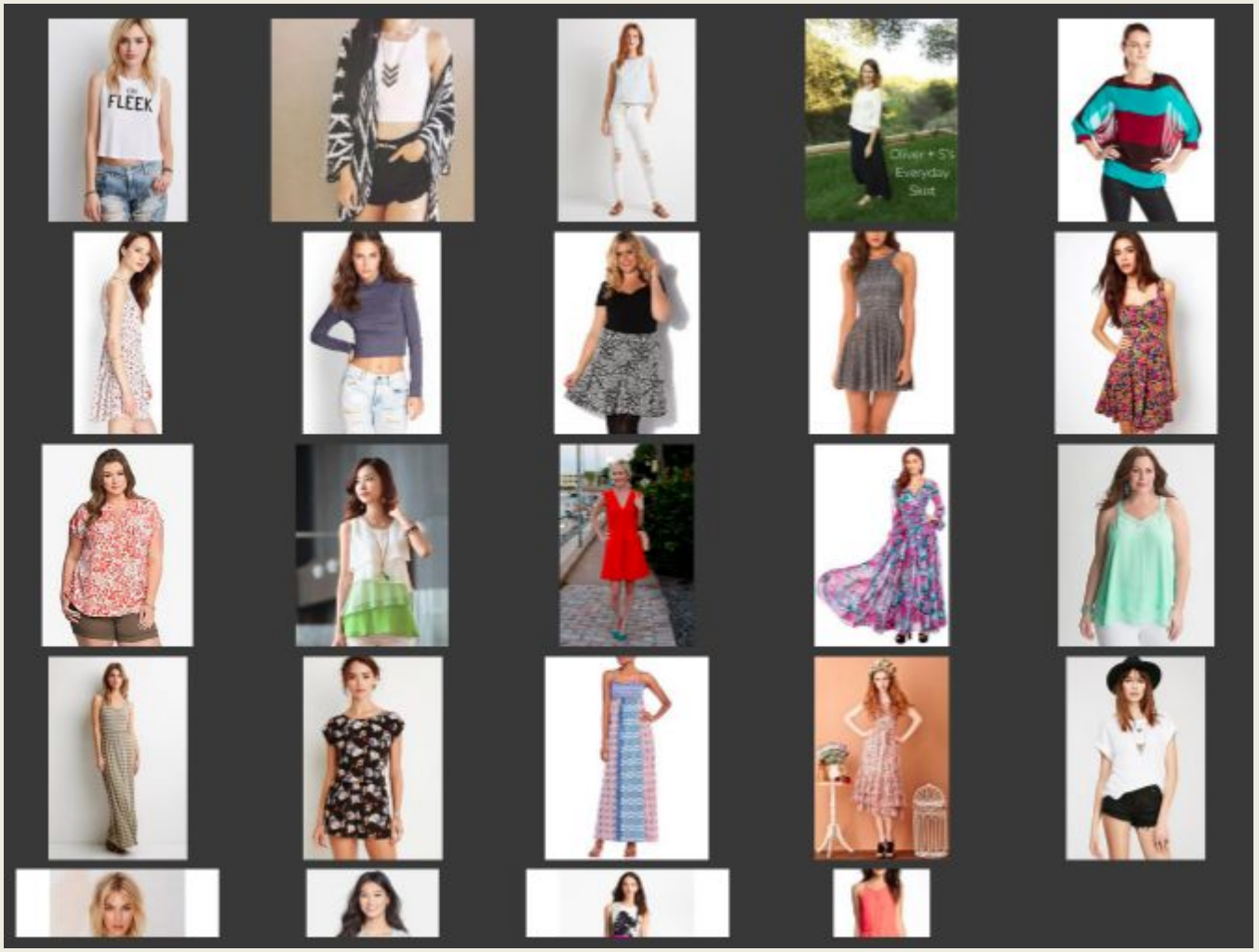
Results

4. Virtual Try-On

Results

5. Weather oriented outfit recommendations

- Spring season-

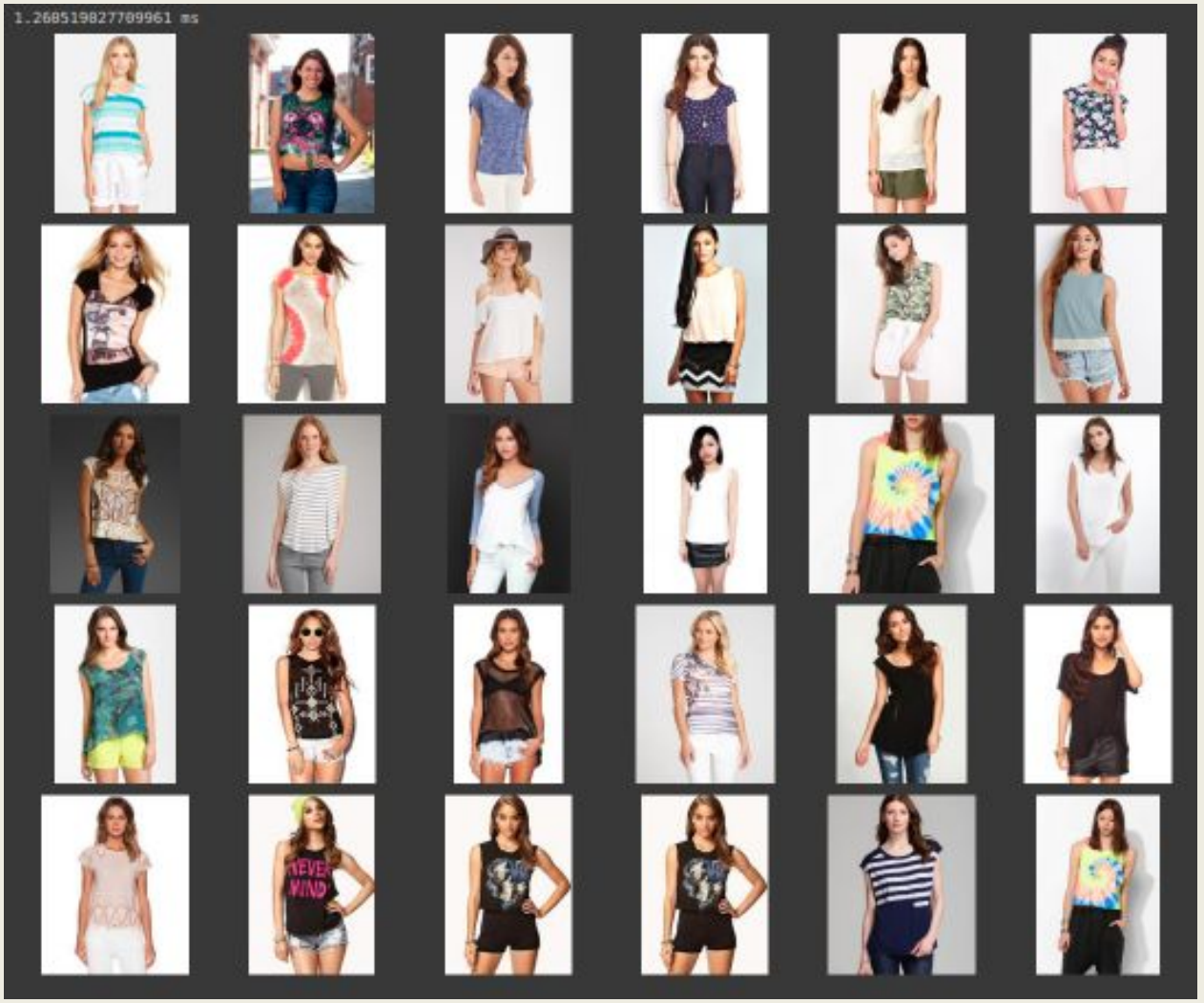


Results

5. Weather oriented outfit recommendations

- Summer &

  Autumn season-



Results

5. Weather oriented outfit recommendations

Accuracy of the model is 92% for prediction and 100%for classification.

80.00% [8/10 2:14:06<33:31]

epoch	train_loss	valid_loss	accuracy	top_k_accuracy	top_k_accuracy	time
0	1.471898	1.353760	0.602250	0.602250	0.897625	16:54
1	1.451260	1.388633	0.593200	0.593200	0.891875	16:19
2	1.440240	1.380244	0.594750	0.594750	0.894000	16:32
3	1.409721	1.315045	0.610425	0.610425	0.905450	16:39
4	1.335867	1.257601	0.634525	0.634525	0.911550	16:36
5	1.315253	1.238566	0.639000	0.639000	0.915575	16:48
6	1.275752	1.166933	0.658325	0.658325	0.923950	17:06
7	1.228926	1.135368	0.664400	0.664400	0.926100	17:09

20.75% [339/1634 03:15<12:28 1.1987]

Buffered data was truncated after reaching the output size limit.

Conclusion

- The proposed system recommends outfits and their color combination to users based on the skin tone of the user.
- The system also considers weather for best suited outfits recommendations.
- A virtual trial room is also provided for the user to try on the recommended outfits.
- Thus this system is a full proof “Fashion Advisor” for people who are worried about what to wear and lack fashion sense.
- This will serve as a real-time system that satisfies customer demands.

References

1. L. Hao and M. Hao, "Design of Intelligent Clothing Selection System Based on Neural Network," 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 2019, pp. 1789-1792, doi: 10.1109/ITNEC.2019.8729417.

1. K. Ayush, S. Jandial, A. Chopra and B. Krishnamurthy, "Powering Virtual Try-On via Auxiliary Human Segmentation Learning," 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Korea (South), 2019, pp. 3193-3196, doi: 10.1109/ICCVW.2019.00397.

1. N. Ramesh and T. Moh, "Outfit Recommender System," 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Barcelona, 2018, pp. 903-910, doi: 10.1109/ASONAM.2018.8508656.

References

4. X. Zeng, Y. Ding and S. Shao, "Applying Image Warping Technique to Implement Real-Time Virtual Try-On Based on Person's 2D Image," 2009 Second International Symposium on Information Science and Engineering, Shanghai, 2009, pp. 383-387, doi: 10.1109/ISISE.2009.9.

5. C. Limaksornkul, D. N. Nakorn, O. Rakmanee and W. Viriyasitavat, "Smart Closet: Statistical-based apparel recommendation system," 2014 Third ICT International Student Project Conference (ICT-ISPC), Nakhon Pathom, 2014, pp. 155-158, doi: 10.1109/ICT-ISPC.2014.6923240.

6. N. Dwina, F. Arnia and K. Munadi, "Skin segmentation based on improved thresholding method," 2018 International ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI-NCON), Chiang Rai, 2018, pp. 95-99, doi: 10.1109/ECTI-NCON.2018.8378289.

References

7. Y. Wen, X. Liu and B. Xu, "Personalized Clothing Recommendation Based on Knowledge Graph," 2018 International Conference on Audio, Language and Image Processing (ICALIP), Shanghai, 2018, pp. 1-5, doi: 10.1109/ICALIP.2018.8455311.

8. L. Yu-Chu, Y. Kawakita, E. Suzuki and H. Ichikawa, "Personalized Clothing-Recommendation System Based on a Modified Bayesian Network," 2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet, Izmir, 2012, pp. 414-417, doi: 10.1109/SAINT.2012.75.

9. Y. Liu, Y. Gao, S. Feng and Z. Li, "Weather-to-garment: Weather-oriented clothing recommendation," 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, 2017, pp. 181-186, doi: 10.1109/ICME.2017.8019476.

THANK YOU