Fashion Wizard

(Similar Cloth Search using a Image Key)

(Image can be a pure cloth image or Human/Object wearing clothes)

Mahbubur Rahman

Matrikel Number: 914626

Course: Learning from Images, M.Sc in Data Science

Berliner Hochschule für Technik Berlin, Germany

Introduction: 2

Use Case Study: 2

Application Area: 3

Related Dataset and Paper: 3

Application Workflow: 4
- Pipeline 1: 4
  - Segmentation: 4
  - Filter Segmentation: 4
  - **Feature Extraction:** 4
- **Pipeline 2:** 5
  - **Search Engine:** 5
  - **Image Lookup:** 5

Project Goal: 6

Difference between Paper Implementation: 6

Evaluation: 7

Challenges: 7

Future Work: 8

Conclusion: 8

References: 8

Appendix A: List of Figures 9

Appendix B: Color Mapping Labels 10
- ROI using segmented color mapping 11
  - Initial Version 11
  - Version 1.1 12

Appendix C: Small and Not Considerable ROIs 12

Appendix D: Application Output 13
- Sanity Check: 13
- Search Result: 14

Appendix E: Error on FashionNet Color Mapping 15

Introduction:

Clothes are one of the most important needs of people. People usually go to the shopping mall to buy clothes. Moreover people buy clothes from online stores and it becomes popular day by day. There is a generation growing rapidly who loves new cutting edge technologies. These generations are the major consumers of different online stores. Also because of the Covid 19 pandemic situation, the online shopping trends and behaviours get the extreme boost. People are enjoying buying their regular products from online stores and the shopping clothes from online stores, has higher frequency than others.

According to the statistics of online shopping, worldwide the growth of online marketing is rapidly rising and it will touch 4 trillion in 2020. Among them, the US has achieved the highest online shoppers. The e-commerce business and market analyst predicts that this will reach 300 million in 2023[1].

Moreover, the online buy will be predicted to increase from 14.1% to 22% in the next year 2023. Another important point, women are the majority in online shopping and their main focus is on fashion related items, for example: clothes etc. The visualisation of the above mentioned statistics of men and women is illustrated in the figure 1 on the appendix section no A.

Either in online or offline shopping, the consumer wants to receive a quick response from the seller, for example: to lookup clothes in the store etc. Statistics show that telephone conversation and prior waiting time make customers more frustrated, as a result, chatbots gain the customer's major attraction and 60% of customers experienced the usage of chatbots. They get responses more quickly than other mediums of conversations [2].

The use of voice assistants is another important reason of getting higher popularity in chatbots usage. According to Google, 20% of searches are voice searches using voice assistants. Moreover, 27% of people around the world use voice chat.

Use Case Study:

In real life, when people want to buy specific or similar clothes from a clothing store(for example: H&M), people show a sample picture to the sales person. Then, the sales person displays the exact cloth or some clothes of similar patterns and textures.

Taking the idea into image processing and deep learning, my proposed application will take an image as input for the search key and give the exact or similar clothes(similar patterns and textures) as output.

---

1 https://optinmonster.com/online-shopping-statistics/#:~:text=That's%2091%25%20of%20the%20country's,their%20first%20item%20on%20Amazon.

[2]https://optinmonster.com/online-shopping-statistics/#:~:text=That's%2091%25%20of%20the%20country's,their%20first%20item%20on%20Amazon.

Application Area:

This application could be used in the online store or chatbot. In the chatbot, the customer can give a picture of a cloth. The chatbot can quickly search that cloth in the store and display the exact same or some similar clothes to the customer. Here similarity means the similar patterns and textures in the clothes. It is a more time saving and effective way of cloth search in the online store using the chatbots. This also helps customers to make quick and correct decisions through visualisation of exact or similar clothes which have different colours and textures.

Related Dataset and Paper:

For the implementation, I use the Deepfashion[3] dataset which is generated as a benchmark dataset for the fashion industry. This dataset contains a large collection of clothes data in different poses. This dataset is generated during the research and publication of the following paper:

Title: DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations.

Book Title: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Author Names: Liu, Ziwei and Luo, Ping and Qiu, Shi and Wang, Xiaogang and Tang, Xiaoou

Published On: June, 2016

I use the In-shop Clothes Retrieval Benchmark, this benchmark is used to evaluate the performance of in-shop Clothes Retrieval. Some key features of this dataset are given below [4]:

- This is a large subset of DeepFashion, containing large pose and scale variations.
- It also has large diversities, large quantities, and rich annotations, including 7,982 number of clothing items; 52,712 number of in-shop clothes images, and ~200,000 cross-pose/scale pairs;
- Each image is annotated by bounding box, clothing type and pose type.
- Contain JSON files for segmented annotation for different clothes types, accessories, body parts.

In figure 2 on the appendix section A, shows the frequency of different clothes categories and it is clearly seen that the upper body cloth's category "top" has highest number of

[3] https://liuziwei7.github.io/projects/DeepFashion.html
[4] http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion/InShopRetrieval.html

Application Workflow:

The fashion wizard consists of 2 pipelines. The input of the application is a query cloth's image which is used as a key to search images in the database. The output of the application is the exact or similar cloth's images. Here similarity means the similar patterns and textures in the clothes. So, the input goes into pipeline 1 and the output comes out from pipeline 2. There is the intermediate output from pipeline 1 and the intermediate inputs into pipeline 2 where both intermediate output and inputs are the same in the sense of the matrix type. These are feature matrices.

Pipeline 1:

Figure 3 illustrates the workflow of pipeline 1. The pipeline 1 consists of the following modules:

Segmentation:

This module takes image as input and then performs a segmentation using masked R-CNN. The outputs of this module are the segmented regions with bounding box's coordinates as top right corner's xy position, width and height. The segmented regions are mapped using color labels.

Filter Segmentation:

After the segmentation process, the segmented regions of each image are filtered by this module. The filtering criteria is the cloth's categories, for example: top, dress, jeans, pants, skirt, sweater, cardigan, jacket etc. The figure 2 in the appendix section no A visualises the frequencies of the top 19 cloth's categories in the Deepfashion In Shop benchmark dataset. The output of this module is the specific ROIs (Region of Interest).

However, because of time and resource constraints, I use the segmentation json files from the dataset for this module and skip the previous module. The segmented json files contain data which are already segmented using deep neural networks by FashionNet. In those segmented data, I find some problems on color labeling. The color labeling problem is discussed in the appendix section no E.

Feature Extraction:

This module performs the computation on detecting keypoints and descriptors, in this case they are known as landmarks or interesting points and descriptions respectively. Here I use SuperPoint [5] keypoints or landmarks detection and related descriptors detection. After computation, I calculate the minimum number of keypoints (k) from the keypoints of all images and make d dimensional array that contains all descriptors of k keypoints. The reason to do this is to make 2nd dimension 'd' equal with both the db matrix and the query matrix. Moreover this is the input requirement of pipeline 2.

[5] SuperPoint: Self-Supervised Interest Point Detection and Description (https://arxiv.org/abs/1712.07629)

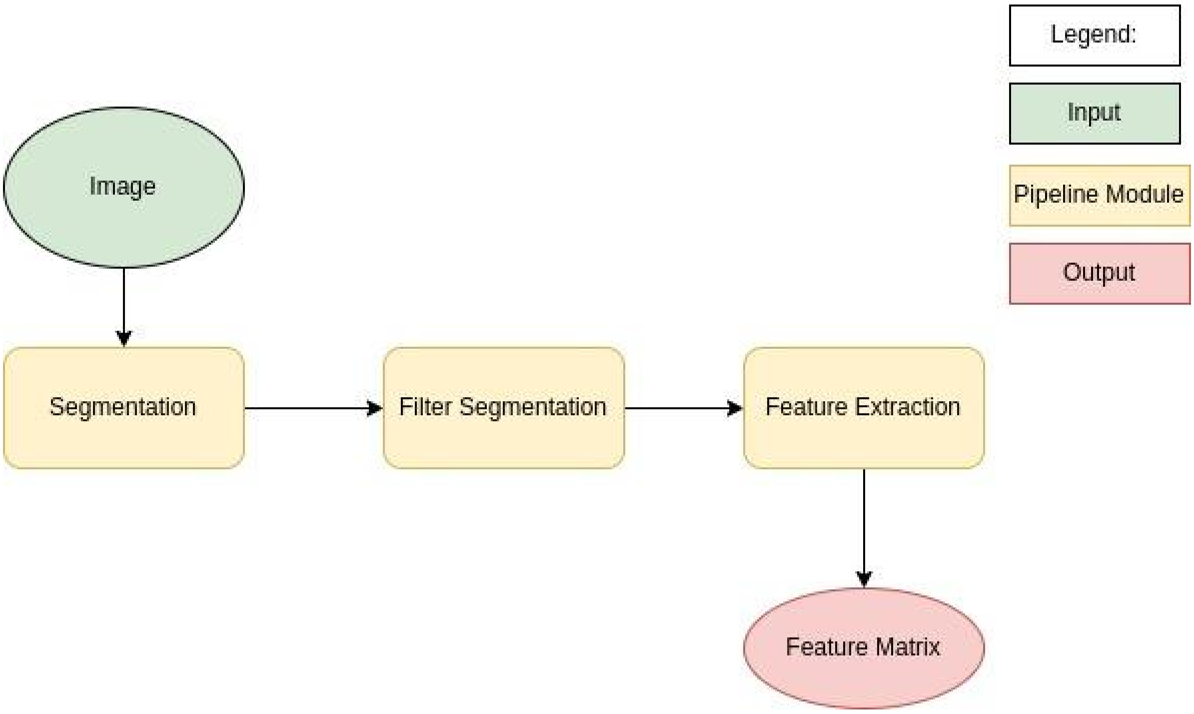


Figure 3: Pipeline 1, image to feature matrix conversion using artificial intelligence algorithms

Pipeline 2:

Figure 4 illustrates the workflow of pipeline 2. The pipeline 2 consists of the following modules:

Search Engine:

I implement this module using the FAISS [6] library that is developed by the Facebook AI Research team. The input is the db matrix(feature matrix of nb x d) that is the vector representation of images and the query matrix(feature matrix of nq x d) is also the vector representation of the query image which is used as a key to search in the database. In simple words, search operation is done by calculating the distance between two vectors in the d dimensional vector space.

Image Lookup:

After search operation, the search engine module gives the index matrix(I) and the distance matrix(D) as the output. The index matrix(I) contains nq rows and k columns where k is the number of nearest or similar images to the query image. Each row is arranged in the ascending order of distance between query and db images. This lookup module performs lookup of those indexes in the files and displays the result of similar images as output of the Fashion Wizard.

[6] https://github.com/facebookresearch/faiss

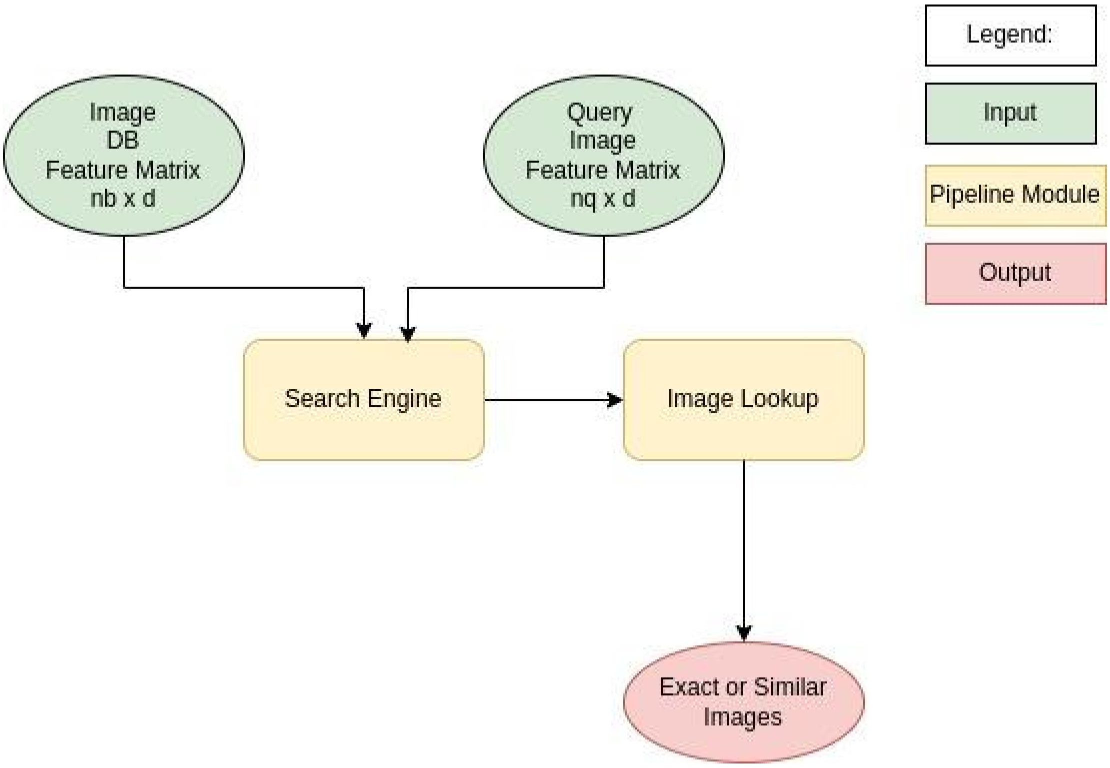


Figure 4: Pipeline 2, image retrieval using artificial intelligence algorithms

Project Goal:

- Understand the dataset and related information inside the dataset.
- Implement the whole application workflow with a very small amount of data from the large dataset. For this purpose, I took the gallery dataset from the segmentation directory because this file contains a smaller volume of data than other datasets. From this dataset, I select the cloth's category top which is the upper body type of clothes because this category has the highest frequency in the dataset among other cloth's categories. After that, I sample those small volumes of data.
- Implement pipeline 1 and test the pipeline 1 whether it is working or not without any error.
- Implement pipeline 2 and test the pipeline 2 whether it is working or not without any error.
- Connect pipeline 1 and 2 properly without any problems.
- Test the connection of two pipelines and the whole workflow.

Difference between Paper Implementation:

FashionNet	Fashion Wizard
Use a fixed number of landmarks for feature extraction or descriptors. Here landmark means keypoints (Upper body clothes uses 6 landmarks, Lower body clothes uses 4 landmarks, Whole body clothes uses 8 landmarks)	Use a dynamic number of landmarks or keypoints based on the variety of dataset and it's volume.
Define landmarks of ROI(for example: Top or Skirt etc) using the bounding boxes, so there are overlapping bounding boxes from others(for example: hand, other type of clothes(coat over shirt etc))	Define landmarks of ROI using segmented color mapping and bounding boxes to get the optimal keypoints and descriptors
The Pipeline of FashionNet consists of a deep convolutional and fully connected neural network.	2 pipelines used in the Fashion Wizard to fulfil different tasks using different artificial intelligence algorithms. For example: for segmentation R-CNN, for keypoint, descriptors SuperPoint, for search FAISS etc.

Evaluation:

Here because of time constraints, I can't implement any evaluation metrics and related visualisations. However, the whole application workflow generates the output successfully without any errors and I successfully accomplished the project goals. The outputs of the Fashion Wizard are nicely generated in the google drive which are separated by db, query and result [7]. In addition, in the appendix section no D contains the sanity check result and some results of few query images.

Challenges:

- I faced challenges during json to dataframe conversion because of few data descriptions in the segmentation labels dataset. For example: in one page of readme file contains description of BBOX Labels where bbox is defined as top left corner coordinates(x1, y1) and bottom right corner coordinates(x2, y2) but in the segmentation json file they have different meaning of 4 values under bbox array. In the segmentation json file they mean top left corner coordinates(x1, y1), width and height of the bounding box.

- In the filter segmentation module of the pipeline 1, some segmented regions contain color values of specific ROI, for example the top has color mapping r-g-b =

[7] https://drive.google.com/drive/folders/1SkQKsmuvgNPzpTrvI3Gcmon2uZKUIQGB?usp=sharing

255-250-250, so during the filtering, the blue color values are filtered correctly using color map but for green and red, some other regions are highlighted or on or enabled instead of disabled or off or dark.

- There are also some specific ROIs which are very small and has less impact of keypoints detection. So those ROIs create problems in overall image retrieval performance. The idea is to filter them and drop them out from the dataset during filtering the segmented regions in the filter segmentation module of the pipeline 1.

Future Work:

- Implement R-CNN in pipeline 1, at the moment it is not possible because of time and resource constraints.
- Improve the segmentation performance so that ROI selection will be more precise using color mapping and bounding box because at present, there are some errors in the color mapping in the given segmentation json file in the dataset.
- Add more configuration on FAISS to improve vector search, for example: add cosine similarity, because at present I just configure FAISS search using L2 distance metric .
- Remove useless ROIs from the dataset in the filter segmentation module of the pipeline 1.

Conclusion:

Fashion wizards will help people to search the desired clothes in the online store and also help to make quick decisions on buying them. From this project and course, I learned a lot about image retrieval techniques and methodologies and I implemented my learned knowledge in the project implementation.

References:

1. http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion/FashionSynthesis.html
2. https://liuziwei7.github.io/projects/DeepFashion.html
3. SuperPoint: Self-Supervised Interest Point Detection and Description(https://arxiv.org/abs/1712.07629)
4. https://github.com/facebookresearch/faiss

Appendix A: List of Figures

Figure 1: Statistics of Men vs Women's online shopping POI(Point of Interest)

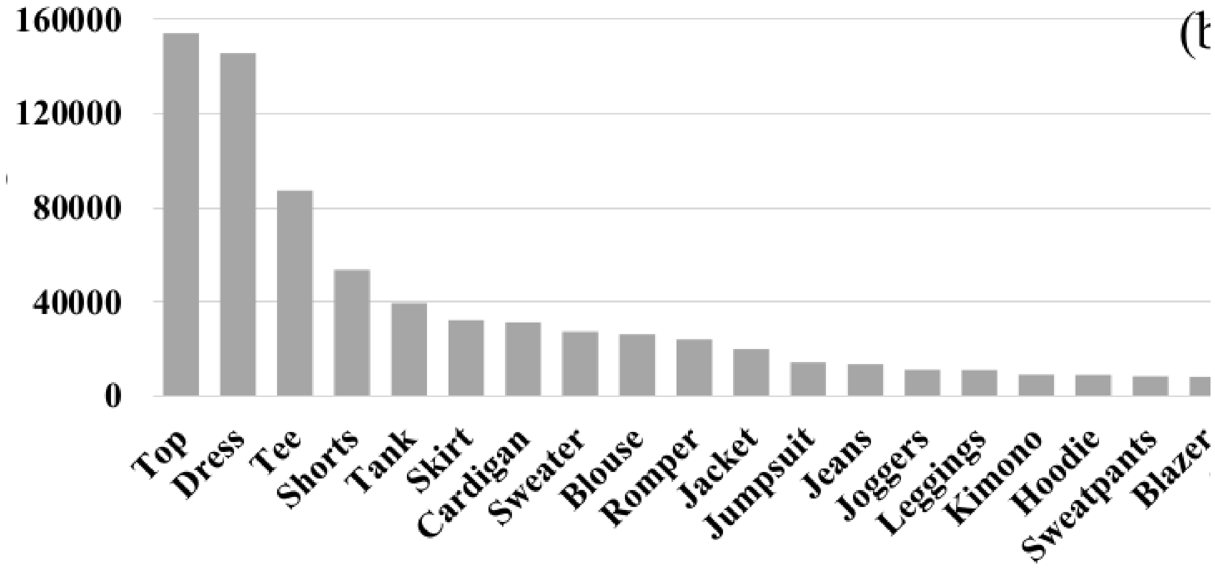


Figure 2: Image number of the top-19 categories [8]

Appendix B: Color Mapping Labels

The following color mapping labels are used for segmenting different categories(for example: top, skirt, belt, eyeglass etc) and labelling them using specific color values.

Color(R-G-B	Label
0-0-0	background
255-250-250	top
250-235-215	skirt
70-130-180	leggings
16-78-139	dress
255-250-205	outer
255-140-0	pants
50-205-50	bag
220-220-220	neckwear
255-0-0	headwear

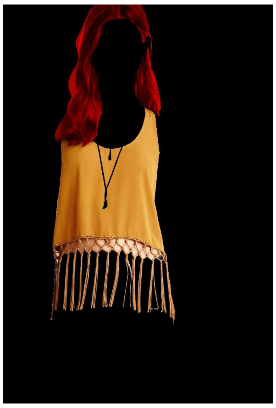
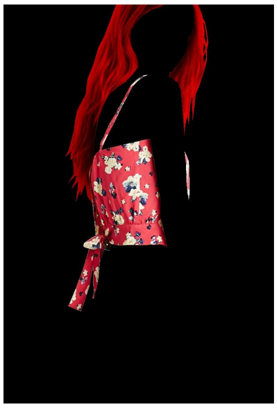
[8] DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations

127-255-212	eyeglass
0-100-0	belt
255-255-0	footwear
211-211-211	hair
144-238-144	skin
245-222-179	face

ROI using segmented color mapping

The following ROIs are the upper body clothes top's ROIs and they are extracted from the original image by filtering using a color mapping label for the top(r-g-b = 255-250-250). In the initial version, the segmented image is filtered using segment category and color label. Then the filtered image is mapped to the original image to get the specific ROI of a specific category. In the next version 1.1, another parameter namely bounding box is added in the filter segmentation module. Some examples of specific ROI filtering are given below:

Initial Version

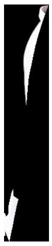
			

Appendix C: Small and Not Considerable ROIs

There are notable portions of ROIs in the dataset which are very small because of overlapping other objects, for example: other types of clothes(hand, other body parts, skin, coat over shirt, tops under the outer or cardigan etc), In some cases, the actual ROI area becomes too small that they become useless for feature extraction. The reason is if the actual ROI becomes too small then proper and as many as possible keypoints or landmarks will not be detected. Later, it will give lower accuracy in the image retrieval. Some examples of such small and not considerable ROIs for the upper body clothes type “top” are given below:

Original Top’s Image	Top’s ROI
	

Appendix D: Application Output

Sanity Check:

Query	Result 1	Result 2	Result 3	Result 4
				
				

Search Result:

Query	Result 1	Result 2	Result 3	Result 4
				
				
				
				

Appendix E: Error on FashionNet Color Mapping

In the FashionNet color mapping, I find some errors on different segment's color labels. For example: the dress type clothes have different color labels in segmented data and documentation. In the following, I mention some miss match between segmented data and color level documentation [9]

Label	Document's Color(R-G-B)	Segmented File's Color(R-G-B)
dress	16-78-139	245-222-179
outer	255-250-205	
pants	255-140-0	
bag	50-205-50	242-138-30
hair	211-211-211	236-82-30
face	245-222-179	16-78-139

[9] https://github.com/open-mmlab/mmfashion/blob/master/docs/dataset/IN_SHOP_DATASET.md#color-mapping-labels