Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 45 No. 11
November 2024

AI Style Recommendation Application based on User Data

Seok-WooJang[1], Jeong-Joon Kim[2*]
[1]Dept. of Software, Anyang University, Anyang-si, Gyeonggi-do, Republic of Korea
[2,*] Corresponding Author, Dept. of Software, Anyang University, Anyang-si, Gyeonggi-do, Republic of Korea

Abstract
Depending on the style, it provides the function to recommend based on the information you entered, but the function of the system considering the user's clothes and coordination style, and the function of the style evaluation are insufficient. In this paper, we recommend the coordination image of the expert based on the styling registered by the user and recommend the coordination combination based on the clothes registered by the user. Moreover, it constitutes with the function of evaluating its own coordination image. We develop Androidbased style recommendation applications to solve user's troubles immediately.
This application collects and analyzes data of clothes and coordination information registered by the user and provides recommendation services and evaluation services using deep learning based on this. It also offers a variety of additional features such as calendar, YouTube recommendation, and My Page. The user can solve the problem immediately through the application and save time and implement the coordination with high completeness.
Keywords: AI, Style Recommendation, Deep Learning, Data Analysis, User Data

1. Introduction

Among the basic elements of human life, food, clothing, and shelter, fashion is a field where one can express oneself and communicate with the outside world. Recently, individual individuality has become more important, especially among the MZ generation, and with this change, interest in fashion is naturally increasing. These days, people worry about clothes every day to express their personality and charm through fashion. However, for modern people living in the ever-changing 21st century, it is not easy to choose clothes to wear every day and organize the wardrobe.According to a survey by British fashion distribution company ‘Marks & Spencer’, adult women spend on average, six months in their lifetime choosing clothes. In fact, there are 152 pieces of clothing in an adult woman's closet on average, but only 44% of the clothes are worn. Among the clothes in the closet, 57 were left unworn, and about 16 were worn only once and never worn. Additionally, according to the survey results, the stress experienced by adult men and women regarding clothes was significant. One in 20 people said they complained about not having anything to wear once a week, and about 15% said they felt bad all day if they didn't like what they wore in the morning. Additionally, about 10% of respondents said that the reason they were late for work or appointments was because they wasted too much time choosing clothes in the morning[1-3].

Recently, as interest in deep learning, a method of machine learning, has increased, the number of clothing recommendation services utilizing it has increased. Currently, most of these services recommend outfits based on the clothes they are selling. In other words, a similarity-based search is performed using the design and category of clothes sold that are like the product owned by the user. As interest in fashion increases, there are more means and methods to communicate, such as sharing one's OOTD (outfit of the day) through various community sites and SNS, but it is still important to evaluate the coordination of one's own styling. Results cannot be known immediately. It takes a lot of time to post a photo on a community or SNS and look at the reactions, and it is difficult to check the completeness of the actual coordination because you must make vague judgments based on the reactions of other users[4].

This system was created to relieve men and women's stress about fashion, which has become a part of expressing individuality and charm in modern society. It is efficient and high-quality through recommendation and evaluation services using mobile wardrobes and deep learning. The purpose is to help you with styling. Through deep learning, we aim to provide a personalized coordination recommendation service by conducting a similarity search based on user data rather than products sold on the market.

This paper is structured as follows. Chapter 2 describes related research, and Chapter 3 describes system design and implementation. Chapter 4 concludes and explains the expected effects of this study.

2. Related research

2.1 Image classification

Image classification involves inputting an image into an algorithm and telling it which class label the image belongs to. For example, assuming that an image recognition algorithm is trained with images that have a class label of one out of three, such as {top, bottom, outerwear}, the algorithm will classify this image as one of the three every time it receives an image. You can classify which class label it belongs to. Even if the image you entered is none of the three, it will give you the closest answer.

Please note that image classification and image recognition are often used interchangeably. If you can not only classify clothes but also classify which brand and model they are, it can be considered image recognition.

There is also an image classification algorithm that adds a location measurement function to a simple image recognition algorithm. These algorithms not only classify what the class label of the image is, but also tell us where the object is within the image using a bounding box[5].

2.2. AI recommendation algorithm

Companies such as Amazon, Netflix, and Google consider recommendation services important because 35%, or more than two-thirds, of sales are generated through recommendations[6].

It appears that the impact of personalized services on the market will continue to grow. This is because more than 80% of customers prefer shopping malls that provide personalized services.

Currently, it is important to consider the user and provide services from the perspective of a selection designer, rather than from the perspective of a product information provider.

The way to implement this is to find and recommend similar products based on the user's past product preference history, and to find people with similar preferences to the user's past product preference history and recommend products they liked, a mixture of two methodologies. So, there is a recommended method. Accordingly, deep learning technology is contributing to enriching data not only for the purpose of algorithm advancement, but also through user profile inference and automatic extraction of product information (image recognition).

2.3. PCCS(Practical Color Coordinate System)

The Japanese Color Research Institute announced in 1964 under the name of the Practical Color Coordinate System, it is a color system with the main purpose of color harmony. This system introduces the concept of tone and has a structure suitable for comprehensive understanding of tone and hue. Not only is it easy to achieve color harmony, but it can also be matched with systematic colornames[7].

Looking at the characteristics of PCCS, it is a mixture of various color systems, including the colors of the Ostwald color system, the sensory system of Munsell's value, Pyrene's concept of harmony, and a description method that modified the tone description code of the American ISCC - NBS.

2.4. K-Means Clustering Algorithm

The k-means algorithm is an algorithm that groups given data into K clusters and operates by minimizing the variance of the distance difference between each cluster. A cluster refers to data divided into several similar classes. Clustering is the process of grouping things into groups with similar characteristics. First, decide how many classes to cluster[8]. Next, determine the center point according to the number of classes. Determine the nearest center point for each point. Now, move the center point based on each mapped point. The newly created center point becomes the average of the points mapped to the same class. In this way, clustering is performed based on the average of the number of classes. The center point is moved repeatedly until it is no longer mapped, and each point finds the closest center point.

2.5. GrabCutAlghorithm

An algorithm for extracting the foreground from an image, excluding the background, was proposed in the paper GrabCut by Carsten Rother, Vladimir Kolmogorov, and Andrew Blake, researchers at Microsoft Research Cambridge in the UK[9].

The Grab Cut algorithm grabs (marks as a border) the area of an object in an image. Afterwards, the data sampling distribution of the background area, not the object, is analyzed to generate a Gaussian Maxture Matrix (GMM) for the background. The created Grab Cut algorithm has the advantage of being able to separate images by specifying a simple area for the object without the user having to set initial points for the object and background twice.

However, if there is a lot of area occupied by objects in the input image and the background area is insufficient, the background GMM is not properly generated, which reduces overall performance.

Also, by repetition, calculation is performed one pixel at a time from the border of the grabbed area. Therefore, there is a disadvantage that the calculation must be repeated as much as the difference in the number of pixels from the border of the maximum area to the object area.

3. System design and implementation

To implement this system, Kotlin (ver 1.4.32), Python (ver 3.8.5), and PHP (ver 1.4.3) are used, and the IDEs are Android Studio (ver 4.1.3) and PyCharm (ver 2021.01). I used it. The server used AWS (Amazon Web Services) EC2 to build Nginx (1.18.0) on Ubuntu server (ver 20.04). The database used mysql (ver 8.0.25).

3.1. System Design

Users consist of experts and the public. The experts perform the evaluation function of this system and indicate when a score exceeds a certain standard is received and provide coordination to the general public. The public can receive coordination from experts, register a photo of the coordination, receive likes, and receive approval from the administrator before being registered as an expert. The application consists of membership registration, evaluation function, clothing and coordination registration, recommendation function, and additional functions. The web server system uses the REST (Representational State Transfer) service method to share data between application users and the database server. In other words, when an application requests a service through a URL, the server returns the data in the database in JSON format.

3.2 Menu Design

The menu of this system is largely divided into five categories. First, the home menu includes current weather, tag-based coordination style recommendations, color analysis coordination recommendations, coordination combination recommendations by TPO, expert recommendations, AI coordination evaluation, and style calendar functions. Considering user convenience, the current weather was placed at the top, and the main functions of the application were exposed on the home menu to increase accessibility and visibility. In the Fashionista menu, you can check the expert's feed photos and ID through the expert list, and bookmark them.Additionally, you can go to the expert feed to see photos of the expert's coordination. The menu focuses on increasing consistency by consisting of functions related to experts. Next, the wardrobe menu includes functions for users to register clothes and combine coordination. To make operation easier, the wardrobe menu was divided into a screen for registering clothes and a screen for registering coordination. Photos of registered clothes and coordination are displayed on the screen to clearly provide users with the results of the function. The feed menu is a space for interaction between users, where you can check published coordination photos and weekly coordination rankings. The My Page menu displays user information and consists of a graph that analyzes not only user information but also level information and the color composition of the user's wardrobe.

3.3 System Implementation

This system, implemented to reduce time wasted by providing style recommendations based on user data to busy modern people, uses Android SDK (ver 4.1.3), YouTube Data API and OpenWeather in Kotlin (ver 1.4.32) environment. API was used. A REST-based web service method was used to transmit and receive data to the mysql database system.Data is returned in JSON format using Volley, an Android HTTP library. A web server program consists of database connections, data structures, and data search statements to insert and retrieve data from each module of the application into the database. Database connection connects to the database server by setting the database server driver and server URL information. Data structure is the part that defines the data structure in the database table. The data search statement is the part where you write query statements to enter, modify, delete, and search values in the table.

When users first sign up, they are evaluated by AI coordinators and classified into experts and ordinary people. Take a photo of the outfit you want to evaluate with a camera or import it from the gallery. You will receive results for your coordination photos through evaluation.

First, import Selenium's web driver. Selenium allows you to control browsers such as Firefox, Internet Explorer, Chrome, etc. and can dynamically crawl web pages using web drivers. It is possible to generate a mouse click on a specific HTML element in the crawl result or fill text into an input element.

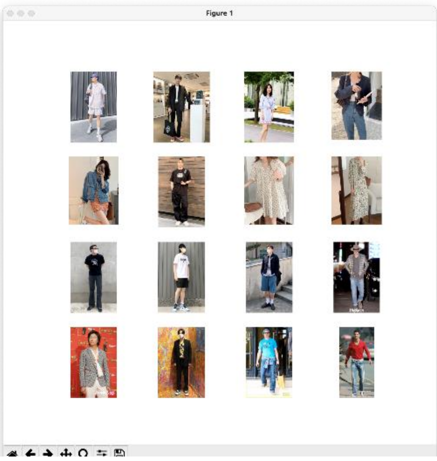

Fig. 1. Crawling Result

Approximately 1,800 Styling photos were crawled using the web crawler shown in Figure 1 are the results of dividing the crawled data set. For a total of 6 months, from March 2021 to September 2021, a total of 63 questionnaires evaluated 100 coordination photos each. It was evaluated using two indicators, best and worst, and the image was classified into the one that received the majority of votes. We obtained a data set for best and worst coordination and preprocessed the image data by adjusting the angle and size through rotation_range and rescale of the ImageDataGenerator object. Through binary classification processing, the label classification result that accounts for the higher percentage of best or worst is output. In order to improve the accuracy of the evaluation model, the number of likes and dislikes evaluated by users in the feed function within the app is counted every month and the top 50 coordination photos with the most are applied to the model. A high level of evaluation functionality will be achieved by continuously adding data to the dataset with user data.

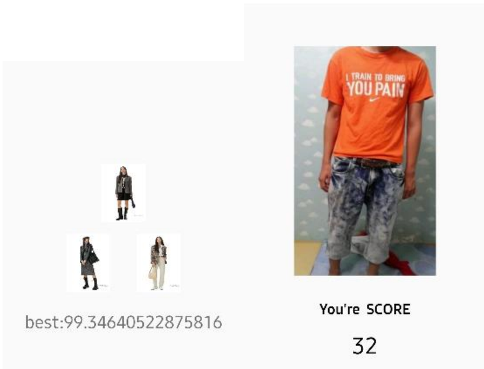

Fig. 2. Resgister Styling and Home Styling Evaluation

Figure 2 shows that when registering as a member, the user receives three coordination photos, runs the evaluation model for each photo, calculates the average score, and displays the evaluation results to the user on the screen. At this time, if the evaluation score is 96 points or higher, you can sign up as an expert. If you sign up as an expert, you can designate one of your own style tags when signing up. You can recommend experts to general users using the style tag. If your score is low, you can sign up as a general user. In this system, users are divided into general users and expert users.

When registering clothes, you must enter information about the clothes. The entered information is stored in the Clothes table. Among the saved data, detailed categories are automatically entered and saved through deep learning. For example, when registering a top, there are various types such as blouses, knits, and T-shirts. The corresponding detailed categories are stored in the database.

Layer (type)	Output Shape	Param #
conv2d_15 (Conv2D)	(None, 28, 28, 32)	896
activation_20 (Activation)	(None, 28, 28, 32)	0
max_pooling2d_10 (MaxPooling	(None, 14, 14, 32)	0
dropout_15 (Dropout)	(None, 14, 14, 32)	0
conv2d_16 (Conv2D)	(None, 14, 14, 64)	18496
activation_21 (Activation)	(None, 14, 14, 64)	0
conv2d_17 (Conv2D)	(None, 12, 12, 64)	36928
max_pooling2d_11 (MaxPooling	(None, 6, 6, 64)	0
dropout_16 (Dropout)	(None, 6, 6, 64)	0
flatten_5 (Flatten)	(None, 2304)	0
dense_10 (Dense)	(None, 512)	1180160
activation_22 (Activation)	(None, 512)	0
dropout_17 (Dropout)	(None, 512)	0
dense_11 (Dense)	(None, 27)	13851
activation_23 (Activation)	(None, 27)	0

Total params: 1,250,331
Trainable params: 1,250,331
Non-trainable params: 0

Fig. 3. Detailed Category Model composition

Stack the layers of the neural network using the Sequential class of the Keras module. Construct the convolutional multilayer Conv2D and Max Poolin2D repeatedly through Add. Among the convolutional layers provided in Keras, Conv2D is used to create a layer. Create a kernel of 3*3 with the number of filters being 32. Set the output image size and input image size to SAME, and specify the input shape excluding the number of samples as the shape of the learning data with input_shape.

The activation function uses 'relu', and to prevent overfitting of the convolution layer, MaxPoolin and dropout are used to extract only the maximum value of the corresponding part. The last output layer uses the softmax activation function to output the category labels as probability values.

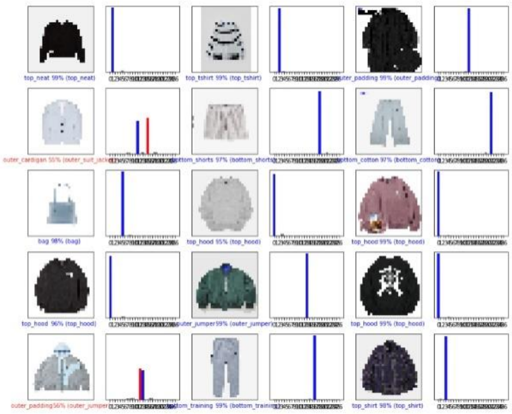

Fig. 4. Detailed category prediction data result

Figure 4 shows correctly predicted labels in blue and incorrectly predicted labels in red.

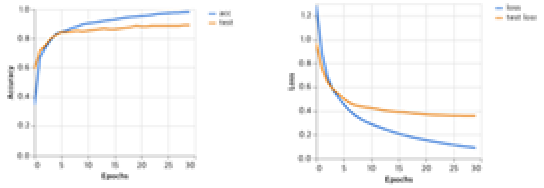

Fig. 5. Model accuracy & Loss function

In Figure 5, 1) accuracy was 0.8774, or 87%, and 2) loss was 0.3805, or 38%. Check the images in the test set using the trained model as follows.

A total of 1,020 image data sets were constructed using web crawling. The background of all images is removed using the grabcut algorithm.

To classify by similar saturation and brightness, the RGB color code of the most widely distributed color, excluding black and white, is changed to HSV color code. Among the HSV color codes, S and V, which are the saturation and brightness values, are matched with each clustered image and saved as annpy file.

Fig. 6. First Clustered Image Data

Figure 6 is the data set that underwent first clustering. Re-clustering is performed using this data set. The data is classified into 13 groups based on the PCCS color system.

1	num_cluster = 13 min_max = [np.min(data, axis=0), np.max(data, axis=0)] quantile = (min_max[1] - min_max[0])/num_cluster/2 clst_mu = np.array([quantile*(2*i+1) + min_max[0] for i in range(num_cluster)])
2	distance = (data - clst_mu.reshape(num_cluster,1,dimension)) distance = distance*distance distance = distance.sum(axis=2).T cluster = np.where(distance==distance.min(axis=1).reshape((len(data),1)))[1] clst_mu_old = clst_mu.copy() for i in range(num_cluster): clst_mu[i] = np.mean(data[np.where(cluster==i)[0]], axis=0) clst_mu = clst_mu_old*np.isnan(clst_mu) + np.nan_to_num(clst_mu)

3	maxEpoch = 100
	for epoch in range(maxEpoch):
	distance = (data - clst_mu.reshape(num_cluster,1,dimension))
	distance = distance*distance
	distance = distance.sum(axis=2).T
	cluster = np.where(distance==distance.min(axis=1).reshape((len(data),1)))[1]
	clst_mu_old = clst_mu.copy()
	for i in range(num_cluster):
	clst_mu[i] = np.mean(data[np.where(cluster==i)[0]], axis=0)

4	clst_mu = clst_mu_old*np.isnan(clst_mu) + np.nan_to_num(clst_mu)

Fig. 7. K-means Clustering

Figure 7 shows the process of re-clustering with a data set that underwent primary clustering. 1) is the process of setting the number of clusters to 13 and determining the center point of each cluster. Find the minimum and maximum values for each dimension. 2) is the process of finding the closest cluster in each data and matching the data. 3) is the process of recalculating the center point of the cluster using the matching results. If there is no factor belonging to a specific cluster, nan is received as the result of np.mean and the previous average is used as is. 4) is a code that continuously repeats the processes of 2) and 3). When the process is converted to a model and image data is input, the name of the class appears as the result.

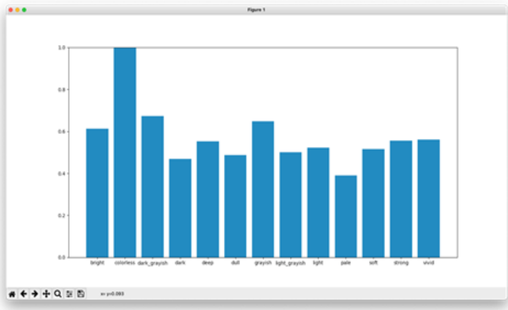

Fig. 8. Similarity Model Result

Figure 8 is the result when inputting an achromatic image using the model. When a user registers a coordination in the application, the coordination image is sent to the input data, the result is obtained and stored along with the coordination image in the database.

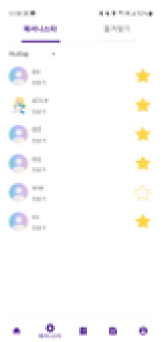

Fig. 9. Expert List Screen

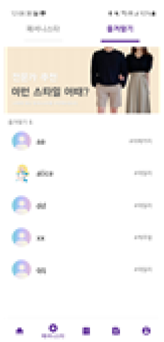

Fig. 10. Expert Favorites Screen

Fig. 11. Expert feed screen

Figure 9 is a screen that receives the profile image, ID, and level information of a user whose level is expert from the user table. The received expert information data is displayed by adding it to the ItemView of RecyclerView. If there is an expert you would like to add to your favorites, click the star button. When you click the empty star button, it changes to a filled star button and is also added to the favorites list. Additionally, the user ID, the expert ID to be favorited, and the time the favorite was pressed are stored together in the favorite table. When you click on a filled star button, it changes to an empty star button and is deleted from the favorites list.

When you click on the item of the expert whose feed you want to see, the expert's ID information is transmitted to the server and moves to the expert feed screen.

Figure 10 is a screen showing a list of experts that the user has bookmarked. The number of favorite experts is indicated at the top.

Figure 11 is the expert feed screen, which shows images released by the expert when registering a coordinator. The expert feed screen shows the expert profile image and ID, the number of registered posts, the number of favorites and followings, and the expert's coordination photos. The photo of the expert coordinator in the profile is used in the coordination recommendation function. The + button at the top right of the screen is a button that appears only when an expert moves to his or her own feed. Even if you are an expert, it is not visible when you move to an expert feed of another ID.

Fig. 12. Custom Youtube Video Recommendations

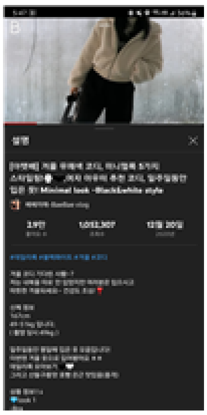

Fig. 13. Moved Youtube Screen

Figure 12 is a screen that recommends fashion-related YouTube videos according to the user's gender and season. Using the Youtube Data API, we search for videos on YouTube by combining the user's gender and season and recommend the top five most relevant videos. Take the videoID and title of the video and put them into an array. Afterwards, useYouTubeThumbnailView to display the thumbnail image on the screen through videoID. When you click on the thumbnail image, the video ID of the video is connected through Intent and moves to the YouTube app. Figure 13 is the screen that moves to the YouTube app when clicking the thumbnail image.

Fig. 14. Home Screen

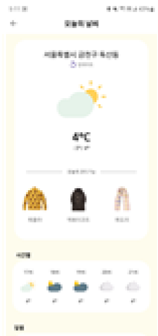

Fig. 15. Recommended clothes for each weather

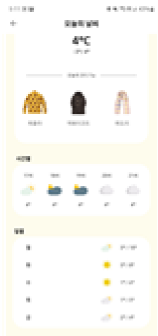

Fig. 16. Weekly Weather

Figure 14 is the home screen, showing the current weather, temperature, and location at the top of the screen. When you click the refresh button, your current location is retrieved and the weather is updated. If you click on attire for each weather, you will be taken to a screen that recommends attire appropriate for today's weather. Figure 15 is today's weather screen, which shows the temperature with intuitive icons and shows the current temperature and today's highest and lowest temperatures. Today's coordination tip is the recommended outfit according to today's temperature. Additionally, as shown in Figure 16, in addition to today's weather, you can see today's hourly weather and 5-day weather at a glance.

Find your location based on the device's GPS and receive your latitude and longitude. Obtain a Korean address using Geocoder, which converts latitude and longitude coordinates into an address, and use the OpenWeather API to request weather information and retrieve it in JSON format.

4. Results

For modern people living in a rapidly changing modern society, wasting too much time and worrying about fashion is causing a lot of stress. Fashion, which has become one of the three basic elements of life, food, clothing and shelter, and a way to express oneself, is organically connected to life. As numerous clothing shopping malls emerge, it is difficult for users to find the style they want and to combine outfits that suit them. In particular, recommending clothes that are being sold can be repulsive to consumers because it has an advertising feel to it. Therefore, we developed an Android-based application with the function of recommending styles based on one's clothes and coordination. In this system, users directly register clothes, combine coordination, recommend expert coordination images based on the coordination they registered, and recommend coordination combinations based on the clothes they registered. It also includes a function that allows you to evaluate your own coordination image.

Existing clothing recommendation applications provide a function that makes recommendations based on the information you entered according to style, but it also provides a function that recommends products that are being sold by simply recommending based on the information you initially entered, such as skin tone and body type. to provide. This system was developed with a focus on users to solve immediate concerns. Through artificial intelligence, a personalized service is provided to users before they purchase a product, providing a guide to determine which combination of these types is suitable. It is possible to improve coordination through a post-purchase evaluation service. However, when combining outfits based on the clothes you have registered, it recommends them according to specific situations, but it has the limitation that the colors do not match harmoniously. Therefore, in the future, we will develop a recommendation algorithm that considers colors according to the user's situation and add a function that combines high-quality coordination.

References

[1] Ki-Tae Kim, “Feasibility of Deep Learning Algorithms for Binary Classification Problems”, Korea Intelligent Information System Society, Vol. 23, No. 1, pp. 95-108, Mar 2017.DOI:https://doi.org/10.13088/jiis.2017.23.1.095

[2] Yun-jin, Kim, “Analysis of image big data using deep learning”, of Statistics The Graduate School of Chung-Ang University, Feb 2017.

[3] Nae-Joung Kwak, Jae-Ho Hwang, “Splitting between Region of Chromatic and Achromatic by Brightness and Chroma”, The Korea Contents Association, Vol. 10, No. 7, pp. 107–114, Jul 2010.DOI: https://doi.org/10.5392/JKCA.2010.10.7.107

[4] Eun-Young Park, Yong-ik Yoon, Young-Ho Park, “Recommending Color Harmonies Based on User Preferences”, Korean Institute of Information Scientists and Engineers, Vol. 17, No. 8, pp. 469–473, Jan 2011.

[5] Dong-Jin Lim, Tae-Hong Kim, “Research on Deep Learning Performance Improvement for Similar Image Classification”, The Korea Contents Association, Vol. 21, No. 8, pp. 1–9, Aug 2021.DOI: http://doi.org/10.5392/JKCA.2021.21.08.001

[6] Ik-Ki Lee, Chang-Ha Lee, Jae-Hwa Park, “Automatic Color Palette Extraction for Paintings Using Color Grouping and Clustering”, The Korea Contents Association, Vol. 35, No. 7, pp. 340–353, Aug 2008.
http://www.mk.co.kr/news/society/view/2016/06/407663

[7] Hyo-Sun An, Sue-Hee Kwon, Min-Jung Park, “A Case Study on the Recommendation Services for Customized Fashion Styles based on Artificial Intelligence”, The Korean Society of Clothing and Textiles, Vol. 43, No. 3, pp. 349–360, Jun 2019.DOI:https://doi.org/10.5850/JKSCT.2019.43.3.349

[8] Nang-Hee Park, Yoon-Mi Choi, “Color& Texture Attribute Classification System of Fashion Item Image for Standardizing Learning Data in Fashion AI”, The Korean Society of Clothing and Textiles, Vol. 44, No. 2, pp. 354–368, Apr 2020.DOI:https://doi.org/10.5850/JKSCT.2020.44.2.354

[9] Seung-Min Sim, Ji-Yeon Lee, Yong-Ik Yoon, “Research and Development of Image Synthesis Model Based on Emotion for the Mobile Environment”, The Korean Society of Clothing and Textiles, Vol. 18, No. 11, pp. 51–58, Nov 2013.DOI: https://doi.org/10.9708/jksci.2013.18.11.051