Artificial Intelligence Shoe Cabinet Using Deep Learning for Smart Home

Jun-Ho Huh and Kyungryong Seo

Abstract The recent development of smart home technology is making people's residential life more affluent. Numerous advanced technologies are being applied to various home appliances and furniture including smart boiler, smart refrigerator and smart bed changing people's everyday life gradually. The shoe cabinet is the furniture people pass through first when entering a house, and it was thought that if IoT function is attached to a shoe cabinet it will give much convenience to people as a component of the smart home just like smart boiler and smart refrigerator. On such a thought and trend, a small processor like Raspberry Pi was attached to the shoe cabinet at home to see the list of shoes, store shoes automatically and recommend right shoes for occasions. The shoes that the user wants to put into the shoe cabinet are stored automatically in the empty space of the shoe cabinet using x-y floater, and the shoe cabinet stores shoes by dividing the kinds and colors of shoes. And it is possible to see the status of the shoes in the shoe cabinet remotely using a mobile application. Moreover, the most appropriate shoes will be recommended when information on type of clothes worn and the destination are put in. The automatic storage of shoes was realized by controlling the input sensor and x-y floater with the Raspberry Pi attached to the shoe cabinet. Classification of shoe images was realized by putting in already classified shoe image data called 'UT Zappos50K' in the Deep Learning model made with Keras framework. Shoe recommendation service is made possible by entering the info on condition of user's clothes and destination and recommending the shoes which score the highest among the shoes in the shoe cabinet using the prepared score chart.

Keywords Artificial intelligence · Cluster computing · Shoe cabinet
Deep learning · Smart home

J.-H. Huh
Department of Software, Catholic University of Pusan, Busan, Republic of Korea
e-mail: 72networks@cup.ac.kr

K. Seo (✉)
Department of Computer Engineering, Pukyong National University, Daeyeon, Busan, Republic of Korea
e-mail: krseo@pknu.ac.kr

J. J. Park et al. (eds.), *Advanced Multimedia and Ubiquitous Engineering*, Lecture Notes in Electrical Engineering 518,
https://doi.org/10.1007/978-981-13-1328-8_108

1 Introduction

The recent development of smart home technology enriches people's residential life [1, 2]. Various home appliances and furniture such as smart boiler, smart refrigerator and smart bed are introduced with the latest technologies which have changed the everyday life of people [3–5]. Shoe rack is the first furniture to encounter when a person enters home and its though if IoT is added to the shoe rack, it can be of great convenience to people as a component of smart home such as smart boiler and smart refrigerator [6–9]. Based on such thought and trend, the shoe rack is designed by attaching small process such as Raspberry Pi that serves to check shoe list, automatic storage and recommendation of shoes for situation. Meanwhile, automatic storage function is to be implemented to move the shoe to the empty unit of the shoe rack by controller pressure sensor and motor through Raspberry when the user puts the shoes on x-y floater. Shoe recommendation can be obtained through mobile application of smart shoe rack. With mobile application, the garment, top and bottom clothes of the user can be input by types through UI, and the destination such as workplace, downtown and travel site can be input. Then, it recommends the most suitable shoes to the user among stored shoes in the smart shoe rack.

To find the most suitable shoes in the shoe rack with the recommendation function, it needs to categorize color and type of shoes. When the shoes are on the x-y floater, a camera on the smart shoe rack takes a photo and that image classifies 7 kinds of shoes using a deep learning model realized with CNN (Convolutional Neural Network) layer. And, using OpenCV image processing library, it enables to classify colors of shoes. Such classified types and colors are used for data attribution of shoes, which can be checked through mobile application of smart shoe rack on real-time basis and which is used for shoe recommendation.

2 Automatic Storage Function

In implementing automatic storage function, it needs a machine that moves the pedestal to the X-Y axis to set the storage position. So, a simple machine, called X-Y machine or X-Y linear machine, is designed to move along the rail. Basically, it is similar to X-Y floater of 3D printer, and it does not need that fine accuracy.

2.1 Rail Operating Principle

Rail itself can be operated in various manners and it is divided into two. One is to move axis with tensile strength using a timing belt. The other is to move axis by rotating movement using a screw. The belt method is relatively cost effective and quick in operation. The screw method is relatively precise and has enough allowable load when operating Y axis.

2.2 Design of Rail Operating Part

To operate rail, the rail in Y axis is composed in a screw method as in Fig. 1. And, for the experiment, it developed a prototype that only considered operation. Thus, it attempts to improve considerably.

In the first place, it consisted of two profiles, but it was changed to 3 for stability on load. Materials used include aluminum profile 30 * 30 × 3 (1000 mm, 700 mm * 2), step motor × 5, ball bearing (thickness 7.2 mm, internal diameter 6 mm, external diameter 20 mm × 20, thickness 6 mm, internal diameter 5 mm, external diameter 14 mm × 6) and 300 * 300 mm wood panel. As in Fig. 2, when X axis is operated, it delivers power by combining to two profiles with a timing belt. In case of Y axis, while it is not shown in the design, the timing belt is fixed on one side in the empty square. As in Fig. 3, it works same force to both sides and moves the shoe pedestal to Y axis.

Storage operating part is designed with a focus on storage only and, it has pedestal where a person stands on it for shoe off. Belt is moved to input shoes. Figure 4 shows storage operating part.

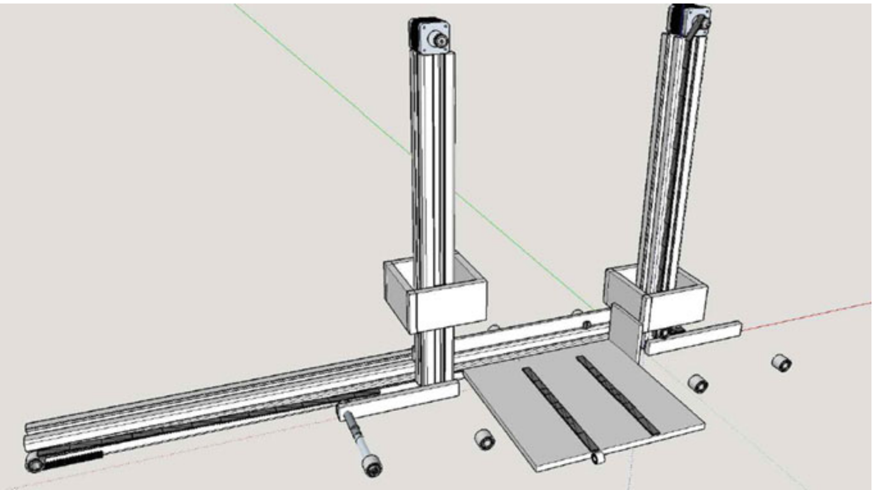

Fig. 1 Design of rail operating part

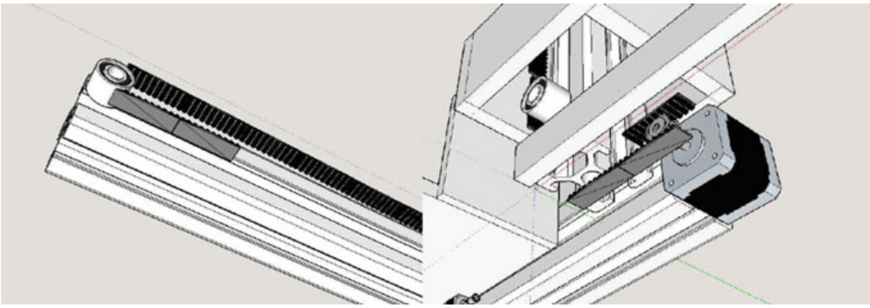

Fig. 2 X axis operating method

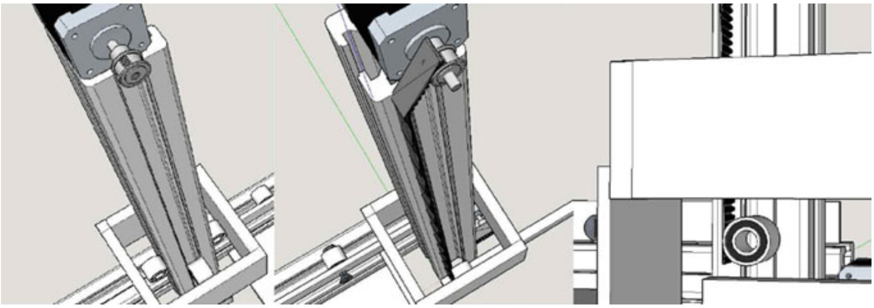

Fig. 3 Y axis operating method

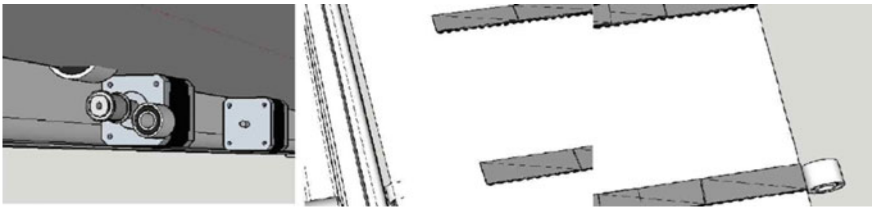

Fig. 4 Storage operating part

3 Control System: Artificial Intelligence Shoe Cabinet

If it is constructed as above design followed by wiring work, the control part, a key of automatic storage system is constructed. The control device for operation consists of Raspberry Pi3, Arduino UNO and a4998 driver chip. Arduino UNO checks whether the shoes are on the plate using a strain sensor and sends a signal to Raspberry Pi in a serial port. Raspberry Pi3 controls the motor in connection with a4998 chip.

As it is 1.8 per 1 pulse, one cycle is rotated per 200 pulse. In this experiment, as 24 sawtooth pulley is used, it moves 24 sawtooth per cycle. The pitch of the timing belt is 2 mm. When rotating 1 cycle of 24 sawtooth pulley, it is total 48 mm. As such, the moving distance is defined. Total design length of horizontal length is 900 mm in this experiment. Among them, 300 (plate) + 30 + 30 (profile) mm are excluded, the moving section is about 520 mm. It arrives to the end by moving 520 mm. If composition of shoe rack is 2 * 2, it is the first row point. Second row point is set 260 mm which is half. If it is converted to pulse, it is 260/48 * 200 = about 1083 pulse. Vertical movement is set in the same manner. The plate part motor, which is operating part of storage is made to process all kinds of shoes by giving a sufficient rotating time.

As such, the movement is composed and each stack is saved in matrix to figure out the stack of each shoe. So, it fills the least number of stack (if it is same such as 1.2 or 2.1 point, smaller one in the first number) to complete the prototype.

3.1 Training Data Collection

The Deep Learning model in this paper is Supervised Learning. To learn image classification model, multiple shoe image files are required as in Fig. 5 and each shoe image files must be categorized by types. Dataset called 'UT Zappos50K' [1] is found and it could collect 50,025 shoes which are already categorized.

Figure 6 shows image file collection with search results. Besides 'UT Zappos50K' dataset, to obtain more diverse images in diverse situations, additional images were obtained through Google image search. It created a Python script that collects images by a specific search word. 'Selenium', a web application test framework, and 'beautiful soup' a html parsing library are used for implementation. Google's image search UI does not show all images at once. It additionally shows a certain number of image files along with the scroll down and DOM is converted dynamically through JavaScript. To download search images, Selenium was used. It was available to access finally rendered DOM through JavaScript after scrolling

Fig. 5 UT Zappos50 K data set

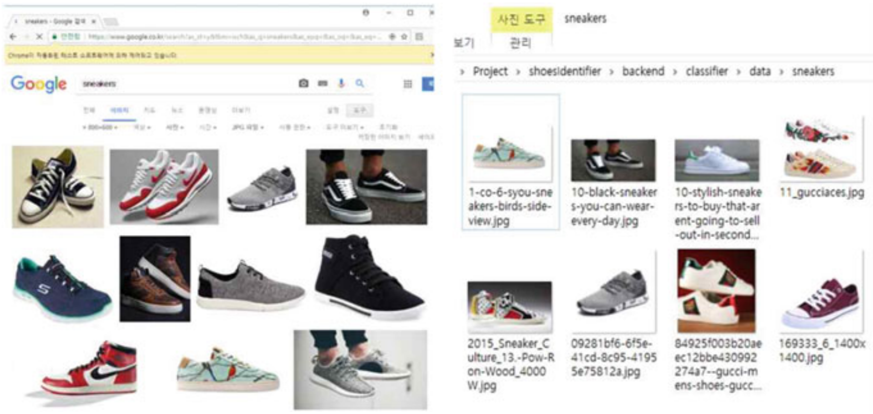

Fig. 6 Image file collection with search results

down by controlling Chrome browser. Then, parsing on html was made using 'beautiful soup' and image files could be downloaded by accessing URL of image files. As a result, additional 300–1,000 image files by type could be obtained.

3.2 Shoe Color Classification

In this paper, to classify shoe color, pixel value of shoe image is considered. Generally, pixel of image file has RGB value, but it has possibility of distortion of color by light if a photo is taken with a camera. Thus, RGB value is converted to HSV color space value. HSV has color (Hue), chroma (Saturation) and brightness (Value). The color value H refers to the relative display angle when red, the longest wavelength, as 0° in the hue circle in a ring shape of visible ray spectrum. Chroma value S means the level of thickness when the thickest of a specific color is set 100%. Brightness value V refers to the level of brightness when white and red are set 100% and black is set as 0%.

3.3 Shoe Color Classification

In this paper, to classify shoe color, pixel value of shoe image is considered. Generally, pixel of image file has RGB value, but it has possibility of distortion of color by light if a photo is taken with a camera. Thus, RGB value is converted to HSV color space value. HSV has color (Hue), chroma (Saturation) and brightness (Value). The color value H refers to the relative display angle when red, the longest wavelength, as 0° in the hue circle in a ring shape of visible ray spectrum. Chroma

value S means the level of thickness when the thickest of a specific color is set 100%. Brightness value V refers to the level of brightness when white and red are set 100% and black is set as 0%. Another thing to consider in the classification of color of shoe image is the background other than shoes. If all pixels of image file are considered for color classification, the background color appears when taking a photo on shoes. To solve such problem, as in Fig. 7, use grab cut algorithm provided by OpenCV library. Grab cut algorithm receives the location information that is considered as landscape in image data and image as a parameter. When carrying out the grab cut algorithm, see the pixel color in the mage and group clustering with similar values. Then, use the border between landscape and background pixel to decide the division. It is followed by connecting with similar labeled pixels for optimization.

When taking a photo on smart shoe rack, the camera direction and pedestal position attached to x-y floater are always constant. Thus, fix the location information that is designated as landscape. The photo taken is as in Fig. 8.

Mobile client provides services such as shoe status view and shoe recommendation system interlocking with the smart shoe rack. It is implemented in Java and works in Android OS. Shoe status view function shows the show status stored in the smart shoe rack in a list view format. Image, color type and location information are shown.

Shoe recommendation function provides interface in which the user enters the destination while wearing clothes (outer, top and bottom) and shows as in Fig. 9. The input information is delivered to Main Server and it is determined according to the highest score among the shoes in storage. The list of recommendation shows is delivered to the mobile client for the user to view. The suggested UML in this paper in relation to the smart shoe rack is as in Fig. 10.

Fig. 7 Grab cut algorithm provided by OpenCV library

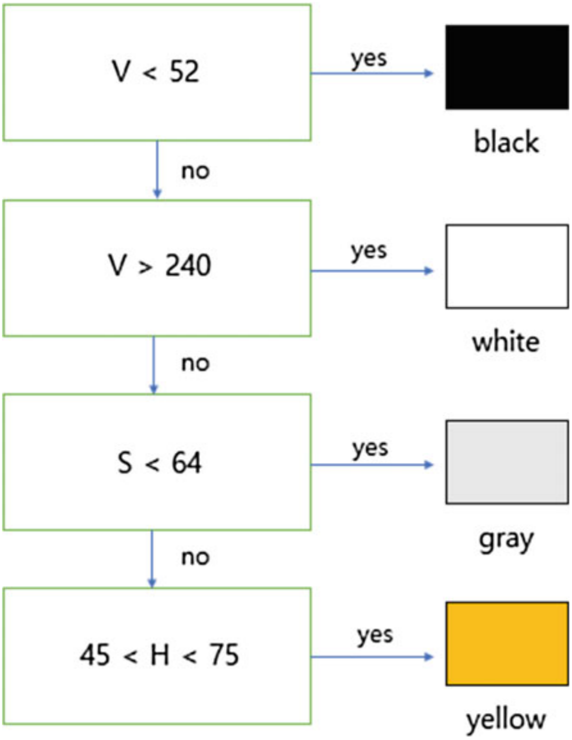

Fig. 8 Photo taken from smart shoe rack

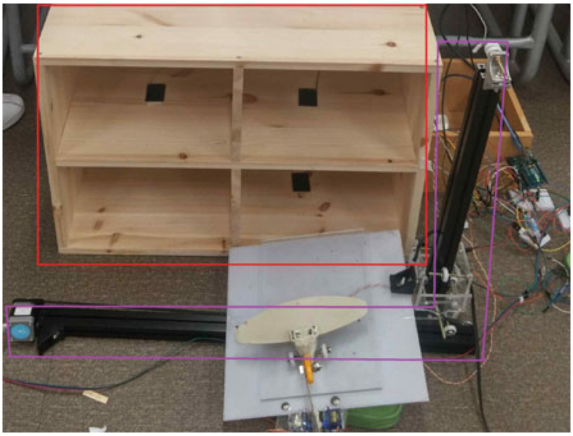

Fig. 9 Overall structure of smart shoe rack (lift)

Fig. 10 Interface to input destination while wearing shoes (right)

4 Conclusion

In this paper, the shoe rack is implemented that provides automatic storage, shoe type classification and shoe recommendation. With Raspberry Pi, pressure sensor and x-y floater were controlled to have an experience of embedded programming. And, it attempted to classify shoes by using the Deep Learning. And, efforts were made to enhance accuracy of classification of type of shoes by learning shoe images to various CNN models. In doing so, it was possible to make a model that serves shoe classification even though it is not perfect and it was an opportunity to enhance knowledge on the Deep Learning. If more shoes could be obtained, higher accuracy would be acquired. The expectation effect of this paper is to provide more diverse functions to people with shoe rack as part of home IoT.

References

1. Huh J-H, Otgonchimeg S, Seo K (2016) Advanced metering infrastructure design and test bed experiment using intelligent agents: focusing on the PLC network base technology for Smart Grid system. J Supercomput 72(5):1862–1877
2. Huh J-H (2017) PLC-based design of monitoring system for ICT-integrated vertical fish farm. Human-centric Comput Inf Sci 7(20):1–19
3. Vosoughi Soroush, Roy Deb, Aral Sinan (2018) The spread of true and false news online. Sci AAAS 359(6380):1146–1151
4. Jordan MI, Mitchell TM (2015) Machine learning: Trends, perspectives, and prospects. Sci AAAS 263(349):255–260
5. Hastie T, Tibshirani R, Friedman J (2008) The elements of statistical learning; data mining, inference and prediction
6. Murphy KP (2012) Machine learning: a probabilistic perspective. MIT press, Cambridge

7. Doan, T.; Kalita, J.; “Predicting run time of classification algorithms using meta-learning,” International Journal of Machine Learning and Cybernetics, Springer, 2016, 1–15
8. Huh J-H, Kim T-J (2018) A location-based mobile health care facility search system for senior citizens. J Supercomput 1–18
9. Garcia Lopez P et al (2015) Edge-centric computing: vision and challenges. SIGCOMM Comput Commun Rev 45:37–42