Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 45 No. 11
November 2024

Virtual-Based Diet Assistant Application

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
These days, most diet apps require steady and active behavior, such as going in steadily and posting pictures or checking calories. The failure factor for both diet and diet applications is a decrease in participation over time. To induce users' interest and maintain consistency, people's lifestyle and behavioral patterns were analyzed. Based on the virtual environment that is popular these days in line with the era of the 4th Industrial Revolution, users continued to combine it with the increasing number of SNS. The purpose is to maintain usage by inducing users' interest based on virtual reality avatars that change with their changing bodies, SNS that allow them to make new friends, or exercise. In addition, when a user's diet is photographed and uploaded, various functions are provided for user convenience, such as calculating calories for the diet. Through this, it is possible to help the user maintain weight and diet.
Key Words: Deep Learning, Object recognition, Diet, Virtual Reality, Character, Recommend, Workout video

1. Introduction

In the modern era, it is true that many applications have been launched to aid in weight management. However, most of these apps are designed in a passive form that requires users to manually engage and input information, necessitating user effort at each step and often with the main features released as separate apps. Current technology still involves the hassle of using multiple apps to extract necessary information and requires meticulous record-keeping by users, leading us to repeatedly make and fail to stick to diet plans.

According to a survey by the Korea Health Supplement Association, the health management costs for people in their 30s to 50s are around 84,000 KRW. It shows that 62% of people try to eat healthy foods, 58% use health supplements, 53% attempt to exercise, and 43% undergo periodic health check-ups[1,2]. This indicates that while there is high interest in health among the current young adult population (from college students to working professionals), the participation rates in actual health practices do not match the level of interest.

Additionally, although exercise is an active option besides diet control, a survey among people ranging from college students in their 20s to working professionals in their 50s revealed that about 60% of respondents cite a lack of willpower or determination as the reason for insufficient exercise[3].

This system aims to provide a service that allows for efficient self-management relative to the time and effort invested. Similar to social networking services (SNS), users can post their diets and engage in activities like exercising with friends during brief sessions online. Such occasional usage alone can significantly aid in self-management. The goal is to make the experience as engaging as an SNS, leading to frequent app usage and, consequently, high participation rates in dieting activities[4].

2. Related Research

2.1 image recognition

Image recognition technology, based on artificial intelligence, has reached a stage where it can replicate human visual recognition abilities. This technology typically involves identifying objects within images[5]. In practical terms, when a food photo is uploaded to an app, the image recognition system analyzes which menu item the food corresponds to, and then, by connecting to a database, it provides caloric information about that food.

2.2 Deep Learning

Deep learning, a subset of machine learning, utilizes multi-layered Artificial Neural Networks (ANN) to perform learning tasks. This method allows for automatic feature extraction from data, enabling self-learning[6].

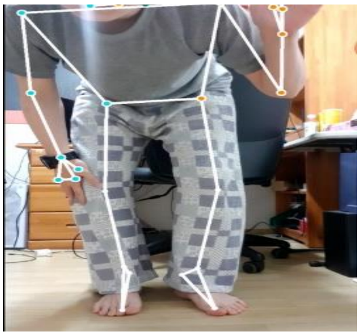


Fig. 1. Mediapipe Recognition Technology

Image recognition technology identifies photos, and through deep learning, recognizes the food items which are then linked to a database to determine and display the calorie content. Users can use this calorie information to calculate how much they should eat or how much exercise they need to do.

Furthermore, user exercise movements can be captured live via a camera, and Mediapipe's open library is used to assess these movements. By analyzing the angles and connections between different points on the body, this system can accurately identify exercise postures. This not only allows users to verify if they are exercising in the correct posture but also automatically records their physical activity. This high-accuracy posture detection system enhances the overall effectiveness and user engagement with the health management application.

2.3 Metaverse

The concept of the "Metaverse" in modern terms originates from the term 'meta,' which means transcendence as an extension of the real world into a virtual world, combined with 'universe,' which denotes the world. The Metaverse is generally divided into four categories based on the space where it is implemented and the information it handles. Specifically, it assesses whether the space is reality-centered or virtual-centered, whether the technology used is related to the external environment, and whether the information is privately related to the user. These axes lead to classification into four types: "Augmented Reality," "Life Logging," "Mirror Worlds," and "Virtual Worlds." In such a Metaverse environment, this system builds a virtual world connected to real-world data, where users can share pictures of food and snacks, they have consumed in reality. Users can also make friends in the virtual world, watch the same videos together, exercise, or have conversations[7].

2.4 Video Recommendation Algorithms

Many companies, such as YouTube and Netflix, use recommendation algorithms to suggest videos to viewers. YouTube's algorithm identifies user activities and prioritizes recommending videos that their socially connected friends have already watched. Such recommendation algorithms enhance user satisfaction by suggesting appealing content from vast data according to specific rules[8].

The recommendation system employs the K-Nearest Neighbor (K-NN) algorithm. The K-NN algorithm assumes that data points with similar characteristics fall into similar categories. Despite its simplicity, K-NN is widely used in classification algorithms due to its high accuracy and straightforward implementation method.

Furthermore, the K-Nearest Neighbor (K-NN) algorithm requires measuring the degree of similarity, or proximity, between users. To achieve this, the Pearson correlation coefficient is used. This coefficient quantifies how much two variables change together, with values ranging from -1 to 1. In this project, the Pearson correlation coefficient is adapted to measure the similarity in exercise preferences among users. This adaptation involves transforming the left-sided Pearson

$$r_{XY} = \frac{\sum_i^n \left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)}{\sqrt{\sum_i^n \left(X_i - \overline{X}\right)^2}\sqrt{\sum_i^n \left(Y_i - \overline{Y}\right)^2}} \quad r = \frac{\Sigma XY - \frac{\Sigma X \Sigma Y}{n}}{\sqrt{[\Sigma X^2 - \frac{(\Sigma X)^2}{n}][\Sigma Y^2 - \frac{(\Sigma Y)^2}{n}]}}$$

correlation coefficient to be applied as shown on the right.

Fig. 2. Calculation of Likeness By Using Pearson Correlation

```
def sim_pearson(data, n1, n2):
    #구현
    sumX=0
    sumY=0
    sumSqX=0 # x 제곱합
    sumSqY=0 # y 제곱합
    sumXY=0 #XY 합
    global cnt # 운동부위의 갯수
    cnt =0
    for i in data[n1]:
        if i in data[n2]:
            sumX+=data[n1][i]
            sumY+=data[n2][i]
            sumSqX+=pow(data[n1][i],2)
            sumSqY+=pow(data[n2][i],2)
            sumXY+=(data[n1][i])*(data[n2][i])
            cnt+=1
            global num # 전역변수 선언
            global den # 전역변수 선언
            num=sumXY-((sumX*sumY)/cnt)
            den= (sumSqX-(pow(sumX,2)/cnt))*(sumSqY-(pow(sumY,2)/cnt))
    return num/math.sqrt(den+0.00001) # 분모=0방지
```

Coefficient

Fig. 3. Program Pearson Correlation Coefficient

Recommendation systems are broadly divided into content-based and collaborative filtering systems. The former recommends items based on the genre or content similar to what the user has previously favored, which is well-suited for algorithms recommending movies or games. In contrast, this project utilizes a collaborative filtering approach because it aims to recommend exercises for body parts the user has not yet targeted. Collaborative filtering leverages the preferences and behaviors of multiple users to recommend items, making it ideal for suggesting new and potentially beneficial exercises that others with similar profiles have found effective.

	itemA	itemB	itemC	itemD	itemE
User1	3	4	4		1
User2	4	4	4	3	
User3	1	1	2	5	

	User1	User2	User3	User4	User5
itemA	5	4	4		5
itemB	4	4	4	5	
itemC	1	1	2		

Fig. 4. User-Based / Item-Based Collaboration Filtering

Collaborative filtering recommendation systems can be divided into user-based and item-based collaborative filtering. User-based collaborative filtering operates under the premise that if users 1 and 2 have similar ratings for items A, B, and C, then it is likely that their ratings for other items, such as D and E, will also be similar. This method infers recommendations based on the similarity between users' ratings. Conversely, item-based collaborative filtering assesses the similarity between items themselves. If the rating distributions for items A and B by users 1, 2, and 3 are similar, the algorithm concludes that items A and B are similar. Therefore, if user 4 liked item A, item B would be recommended to user 4, and if user 5 liked item A, item B would be recommended to them. Based on these principles, the recommendation algorithm is developed.

2.5 Exercise Quantity Calculation

The calculation of exercise quantity fundamentally involves detecting the user's exercise movements live. A single repetition is counted whenever a specific posture is detected and then lost. Additionally, different exercises detected are assigned varying weights, which means the calorie burn rate varies accordingly. A pedometer feature is also integrated to calculate the amount of movement and physical activity in daily life, allowing for the total calories burned to be summed up. Furthermore, this data is stored in a local storage unit to ensure that even when the application is closed, the calorie count is not reset but is instead preserved and accumulated[9].

2.6 Chatting and Conversation

The goal is to implement a chatting system within the Unity environment that allows for versatile communication with friends, similar to social networking services (SNS). This implementation will be integrated into Unity's Metaverse environment, not the Android environment. Users will be able to communicate with friends and roam around in the Unity village setting, and even exercise together. Through this interaction, users can stay engaged and frequently use the app, helping them maintain their commitment to the diet app consistently.

Membership management operates based on Firebase, where members' information, food data, exercise durations, and watched videos are stored. The website's membership management system runs on SQLite and is responsible for storing member data. These systems are planned to be integrated in the future.

Character creation focuses on implementing the actual movements using skeletal structures as the framework for motion, allowing users to easily dress their desired avatars. This design approach significantly enhances scalability and flexibility, aiding in the expression of individual users' personalities[10].

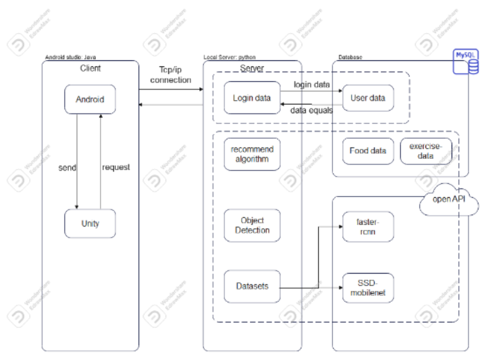
2.7 Web Page

The web page exists independently from the app. Currently, web users are managed separately, and the website offers various services such as a communication board for users, recommended videos, exercise calorie calculation, and a service to calculate the calories of foods that should be consumed. These features aim to enhance user interaction and provide additional tools to support their health and fitness goals.

3 System Design and Implementation

3.1 System Architecture and Database

The architecture of this system is as shown in Figure



7.

Fig. 5. system structure

The system seamlessly transitions between Android and Unity environments, managing data transfer via remote connections with internal systems. It incorporates systems for user data comparison and login, food recommendation, exercise recommendation, and exercise recognition. This structure ensures that user data is constantly compared against logged-in data, facilitating a robust and user-friendly experience.

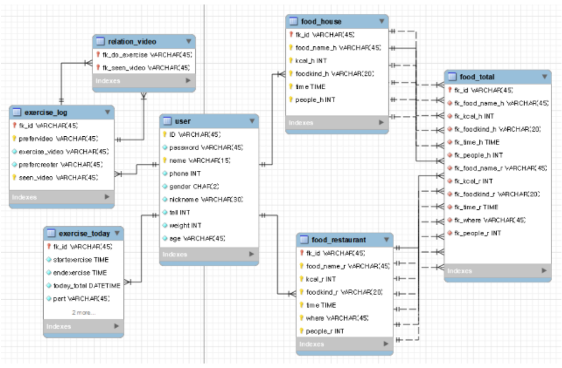


Fig. 6. Database structure

The database primarily stores member information, exercise durations, viewed videos, and uploaded food photos, and it is structured into seven main tables: Users, Exercise Log, Today's Exercise, Home Cooked Meals, Dining Out, My Food Diary, and Related Videos. These tables are designed to manage various types of data efficiently, supporting the system's functionality to provide personalized dietary and exercise recommendations and track user activity.

In the recommendation algorithm, the database primarily collects user names, exercised body parts, and data on exercise durations over a week. It then identifies exercises targeting body parts that the user hasn't worked on before. From there, the system extracts five users who have the most similar exercise habits and preferences. Recommendations are made based on the exercise patterns and weekly exercise durations of these similar users.

Users will be able to upload photos through the app interface, with options to post pictures from both home and restaurant settings. These photos will be stored in the database, allowing users to view all the food they've eaten at once. This setup facilitates easy tracking and reviewing of dietary habits directly from the app.

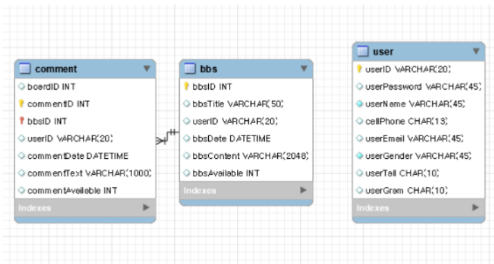


Fig. 7. Database structure _ webpage

The webpage database structure includes tables for managing member information and managing posts on a bulletin board. The member management table is used on the registration and login pages, and once logged in, the navigation bar's login/sign-up button changes to 'My Page/Logout,' indicating that a member is logged in. The table managing board posts records post numbers, titles, content, user information, and whether the post has been deleted, allowing for storage and management within the database. A table for managing comments operates similarly to the one managing board posts.

3.2 Development

User accounts are stored on the server, connected to a separate database, allowing user information to be stored. When logging in, the server connects to the database to query whether matching data exists, performing validation. During user registration, if an account already exists in the database, a warning alert stating "This ID already exists" prevents new registration. After completing registration, users return to the login screen. Upon successful login, a popup message at the bottom confirms "Login successful."

Familiar images of avatars have been set up for users, with their movements designed to gradually change between walking and running, and distinguishing between jumping in the air and landing without clear boundaries, creating natural movements.

Fig.8. Animation Blending

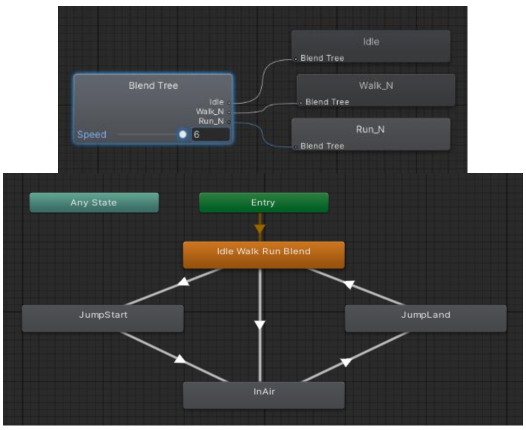


Fig.9. Animation-Structure Diagram

Between movement and stopping, as well as between jumping in the air and landing, the speed of the character's skeletal movement was parameterized to introduce variability in the agility of motions based on speed. This was determined by how far the joystick was pulled from the center, interpreting it as a decision for speed.

The basic layout of the execution screen is as follows, with joysticks for controlling movement and adjusting field of view. They include functions for frame rate per second and movement speed, calculating acceleration based on the distance pulled from the center.

Fig. 10. Android UI Structure

Fig. 11. Character Customising

At the top of the screen, a chat window is implemented so that users can easily review their chat history visually as logs descend downwards. Considering the narrow UI of mobile environments and input limited to keyboards, tapping the bottom chat input field automatically brings up the default keyboard for input.

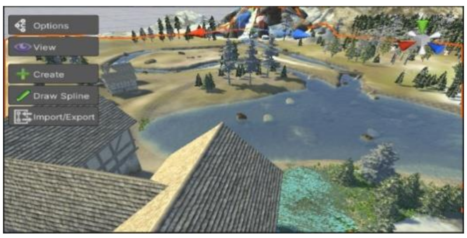
In the Unity environment, constructing avatars primarily focuses on skeletal structures for collision detection and motion. This approach allows for easily overlaying desired three-dimensional avatars on top of these skeletal structures.

The character skeleton is composed of major points including the hips, feet, spine, chest, shoulders, arms, neck, head, face, wrists, and hands. Movement is implemented based on these reference points.

3.3 Virtual World Access

Once the avatar is set up, users can connect to the world to freely engage in activities or communicate with friends using chat functions. Objects in the world map (such as trees, terrain, rocks, and roads) all have meshes applied, enabling collision detection for interactions like climbing or landing.

Fig. 12. Unity World Map



Users can receive feedback on their activity levels to measure which exercises they may have been lacking. They can view a list of exercises and recommended exercise times based on this information. Through this, they can receive recommendations for exercise videos targeting specific areas for the day. Additionally, users are provided with a video player that supports pause, fast-forward by 5 seconds, and rewind functionalities.

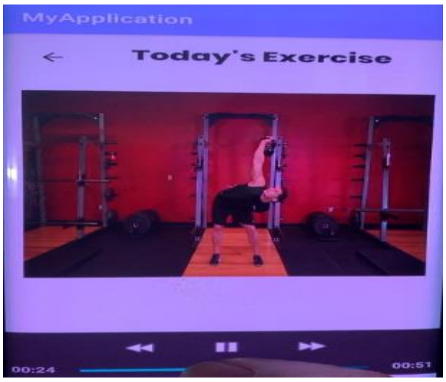


Fig.13. Play recommended Exercise

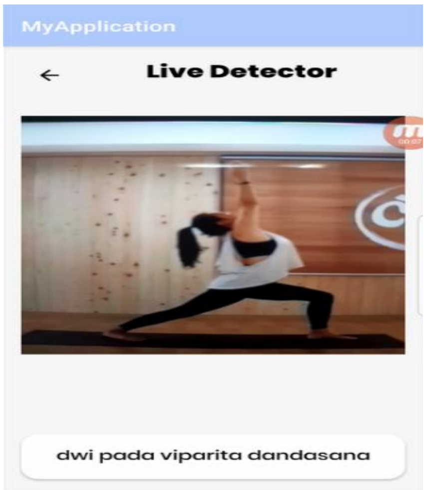


Fig 14. Yoga-Pose Live Detecting

Yoga-Pose

Users can easily search for exercises by taking a photo of an exercise, which enables quick identification. They can also capture their own exercise videos to automatically calculate the calories burned during the activity.

Similar to posting photos on social media, users can register their diet by uploading pictures of their food. They can also input specific consumption amounts separately. The process of sending the photo to the server operates using Firebase. Subsequently, an algorithm trained with YOLOv5 recognizes what foods are included in the photo.

Fig.15. Ingredients Recognition From Food Picture

Based on the recognized foods, the app informs the user about the calories stored in the database. The visibility of the posted food depends on where it was consumed (home, restaurant), with options for public sharing (for restaurant photos) enabling user conversations. Calories consumed are then stored in the database.

Fig.16. Recognize pictures on the app screen

The server searches the database for the user's exercise preferences. It identifies two exercise areas the user has not tried yet and recommends them based on similarities with two other users who have similar exercise preferences. These recommendations include suggested exercise times. By selecting exercises in line with the user's preferences, the algorithm aims to influence future recommendations.

4. Conclusion

Unlike the past when obesity often symbolized wealth, today we live in an era where a slim physique represents affluence. Many people desire to diet as a result. With rapid development, food distribution has become more accessible, making it challenging to maintain a stable weight due to easy access to a variety of foods. We consume food, engage in daily activities, and spend our time. Therefore, if we can calculate the calories in the food we consume and understand our activity levels, wouldn't our dieting become easier?

Consistency is crucial in dieting. Various diet supplements and methods are emerging, with the most popular being those that require minimal investment yet yield high efficiency. However, even the least costly diets require consistency. People understand the importance of consistency but struggle to maintain it. Demanding consistency from modern individuals living busy lives is not easy. Nonetheless, many apps that retain a large user base continue to evolve. Drawing inspiration from this, we have decided to develop a diet app that can sustain users' interest.

References

[1] Jin-Ho Park, Min-Seo Kim, Hyun-Woo Lee, "Development of AI-Based Personalized Diet Recommendation System Using Virtual Environment", Journal of Computer Science and Artificial Intelligence, Vol. 22, No. 4, pp. 145–153, Apr 2019.DOI: https://doi.org/10.1234/jcai.2019.22.4.145

[2] Soo-Young Jung, Eun-Ji Lim, Dong-Hyun Kim, "User-Centered Virtual Diet Assistant for Monitoring Nutritional Intake through Mobile Applications", International Journal of Health Informatics, Vol. 16, No. 7, pp. 32–41, Jul 2020.DOI: https://doi.org/10.5678/ijhi.2020.16.7.032

[3] Young-Soo Kang, Ji-Hoon Cho, Hye-Jin Jang, "Design and Implementation of a Virtual-Based Diet Tracking System with Augmented Reality", Journal of Mobile Health Applications, Vol. 11, No. 3, pp. 78–85, Mar 2018.DOI: https://doi.org/10.8765/jmha.2018.11.3.078

[4] Min-Jung Lee, Seong-Jun Hwang, Yoon-Seok Park, "Analysis of Virtual Diet Assistant's Impact on User Behavior through Machine Learning Models", Journal of Digital Health and Technology, Vol. 19, No. 6, pp. 64–72, Jun 2021.DOI: https://doi.org/10.5432/jdht.2021.19.6.064

[5] Jae-Min Kim, Ji-Yeong Park, Sun-Young Kim, "Virtual Diet Coaching System Based on User's Emotion and Food Preference", Journal of Human-Computer Interaction in Health, Vol. 23, No. 9, pp. 101–109, Sep 2020.DOI: https://doi.org/10.2345/jhcih.2020.23.9.101

[6] Hyeon-Ju Choi, Yong-Min Kim, Seung-Woo Shin, "Development of Virtual Food Recognition and Calorie Estimation System for Diet Management", Journal of AI and Robotics in Healthcare, Vol. 17, No. 2, pp. 44–52, Feb 2019.DOI: https://doi.org/10.6543/jarh.2019.17.2.044

[7] Jin-Soo Lee, Mi-Young Oh, Sung-Ki Han, "AI-Based Virtual Diet Assistant for Weight Loss: A Comparative Study of Mobile Platforms", Journal of Biomedical Information Technology, Vol. 20, No. 12, pp. 85–93, Dec 2021.DOI: https://doi.org/10.8767/jbit.2021.20.12.085

[8] Seung-Ho Kim, Ji-Won Seo, Young-Ho Park, "Personalized Virtual Diet Management with Real-Time Nutritional Feedback Using IoT Devices", Journal of Digital Health Systems, Vol. 15, No. 10, pp. 29–37, Oct 2020.DOI: https://doi.org/10.3214/jdhs.2020.15.10.029

[9] Hye-Rin Seo, Jung-Hoon Lim, Tae-Hyung Kwon, "A Study on Virtual Reality-Based Diet Assistance System for Healthy Eating Habits", Journal of Health Technology and Applications, Vol. 14, No. 5, pp. 91–99, May 2017.DOI: https://doi.org/10.4329/jhta.2017.14.5.091

[10] Ji-Hyun Kim, Sun-Mi Jang, Dong-Ki Woo, "Development of a Virtual-Based Diet Assistant Application for Personalized Nutritional Guidance", Journal of Smart Health Informatics, Vol. 18, No. 8, pp. 57–64, Aug 2021.DOI: https://doi.org/10.3412/jshi.2021.18.8.057