Proceedings of the 3rd International Conference on Computing Innovation and Applied Physics
DOI: 10.54254/2753-8818/31/20241144

The disease prevention and rescue system based on Lorentz-RR analysis technology for public welfare organizations

Sun Maoran[1, 4, 5], Song Qingqing[2, 6](Joint Work), Peng Shaohan[3, 7]

[1]School of law and politics, Zhejiang Sci-Tech University, Major: Master of Social Work, Bachelor of Electronic Information Engineering, research interests: smart aging, digital social work, social organizations, Contributions: subject selection, system design and planning
[2]Balerusian State University, Applied Mathematics and Informatics Major, research interests: computer data analysis techniques, high performance computing, distributed systems, Contributions: technical solution design, realization and validation of the technology
[3]Qixin School, Zhejiang Sci-Tech University, Major: Computer Science and Technology, research interest: computer vision, Contributions: data organization and summarization of documents
[4]Corresponding author

[5]sunmaoran0911@163.com
[6]fpm.sunC@bsu.by
[7]pshalearner@163.com

Abstract. Health concerns have become a significant focus in people's daily lives. Currently, with the increasing demand for monitoring human health, some social organizations need to strengthen health detection technology, and various related technologies are emerging. The aim of this study is to develop a social organization disease prevention and assistance system that integrates the advantages and resources of social organizations, social service institutions, social workers, caregivers, service recipients, and their families. The main system is designed based on a front-end and back-end separation architecture, using core technology: the Lorenz-RR scatterplot classification algorithm to achieve the selection of classification algorithms, the development of AlexNet algorithm, and the optimization of dataset expansion algorithm, Thus, the improvement of the Lorenz-RR scatterplot classification algorithm of the AlexNet model is achieved, and the development of a social organization disease prevention and assistance system is completed. It has high accuracy and sensitivity in the analysis of relevant indicators of service objects, and its application value is significant, with widespread promotion significance.

Keywords: social organization, Lorenz-RR, AlexNet algorithm, health

1. Introduction: Background and Significance of the Development

1.1. Background System Development

The field of human health monitoring are emerging rapidly in response to growing demands. Traditional health monitoring methods, such as single-parameter monitoring devices and non-intelligent multi-parameter monitoring systems, are no longer sufficient to meet the needs of individuals or others for health monitoring.

There is an urgent need for a multi-terminal health monitoring platform to simplify the configuration process of health monitoring devices, facilitate the sharing of health information, and provide real-time health monitoring and warning functions. Since the mid-20th century, health monitoring tools like electrocardiograms have gradually entered the information age of computers, with a significant increase in sampling rates, enabling further analysis of monitoring data and the possibility of health warnings. However, the development of the specific function of health warnings has been slow, hampered by factors such as accuracy, even though mature solutions or warning strategies in the market are currently scarce.

At present, most health warnings are triggered “during” or “after” an incident, typically providing warnings only after the wearer of the device has already exhibited evident issues such as abnormal heart rhythm, blood pressure, or oxygen levels. This approach falls short of effectively informing the wearer and preventing the deterioration of health problems.

1.2. Significance of System development

By integrating software technology, electronic information technology, cloud computing technology, and data analysis technology, we aim to provide a mature and promising solution for the field of smart health monitoring. This solution can cover mainstream functionalities available in the market and, furthermore, simplify the configuration process of health monitoring devices, facilitate health data sharing, enable real-time health monitoring, and implement health warning and health data analysis functionalities.

The completion and implementation of the system can effectively contribute to the development of social organizations, smart elderly care, and other industries. It ensures the safety of both modern young individuals and elderly populations, effectively prevents a return to poverty due to illness, promotes industrial development, and empowers the field of smart health [1].

2. The Significance of Lorenz-RR in Disease Prevention.

On one hand, Lorenz scatterplots can provide an intuitive and rapid preliminary screening of diseases, facilitating continuous detection and analysis. This makes them particularly suitable for medical-related institutions such as elderly care centers, children’s welfare homes, health foundations, etc. Lorenz-RR offers an efficient and convenient monitoring method, helping these institutions quickly identify patients who require focused attention and intervention. This, in turn, enables better health management and disease prevention efforts.

On the other hand, in the current social context of China, issues related to poverty caused by illness or a return to poverty due to illness are prevalent [2]. There is a need for in-depth analysis of the reasons behind poverty caused by illness and the implementation of effective measures for governance. Through the analytical capabilities of Lorenz-RR, associations and patterns between these diseases and other factors can be identified. This analysis serves as a scientific basis for formulating effective policies to address poverty caused by illness.

3. System Overall Design

3.1. Overall Framework Design

The system adopts an overall architecture with a separation between the front end and the back end, utilizing a non-equivalent distributed architecture for the back end [3].During the phases of requirement

DOI: 10.54254/2753-8818/31/20241144

analysis and preliminary design, user needs and usage scenarios have been clearly defined. Now, the overall architecture of the system will be designed.

For the scenario targeting family members, The design described in this paper employs an interaction platform based on UniAPP technology as a WeChat mini-program. Utilizing the WeChat mini-program platform allows for cost-effective adaptation to a wide range of user devices. In the scenario for caregivers, the interaction platform also uses a WeChat mini-program based on UniAPP technology. For the administrator scenario, the design described in this paper utilizes a web front-end program based on Vue.JS. In the case of the monitoring scenario, a web front-end program using HTML5+JavaScript+CSS3 technology is employed for interaction [4]. Finally, for the scenario involving device wearers, the design described in this paper is divided into hardware and software components. The hardware part uses a sensor platform based on the MAX86176 chip, while the software part is implemented using an Android app. The software portion of the hardware is based on an open-source embedded intelligent wristband system.

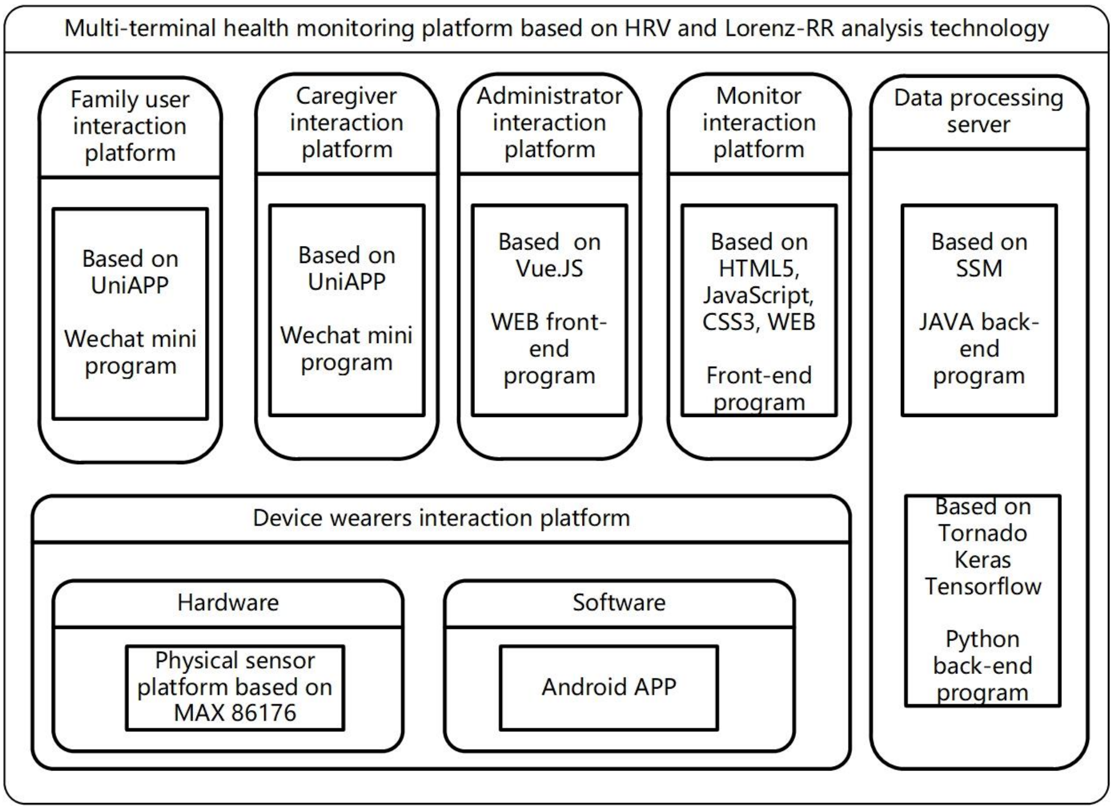

Figure 1. Overall Architecture of the System

All user interaction platforms in the system are interconnected with the data processing server. The data processing server is a distributed computing architecture consisting of the main business server and the health warning server. The main business server is a JAVA server based on the SSM framework, directly providing services to all interaction platforms. The health data analysis server is a Python server specifically designed for Lorenz scatterplot classification, based on Tornado, Keras, and Tensorflow technologies.

3.2. Division of Subsystems

After designing the overall architecture of the system, a thorough analysis has allowed for the clear identification of the system consisting of six scenarios and eight components. In this section, each part of the system will be named and systematically delineated.

DOI: 10.54254/2753-8818/31/20241144

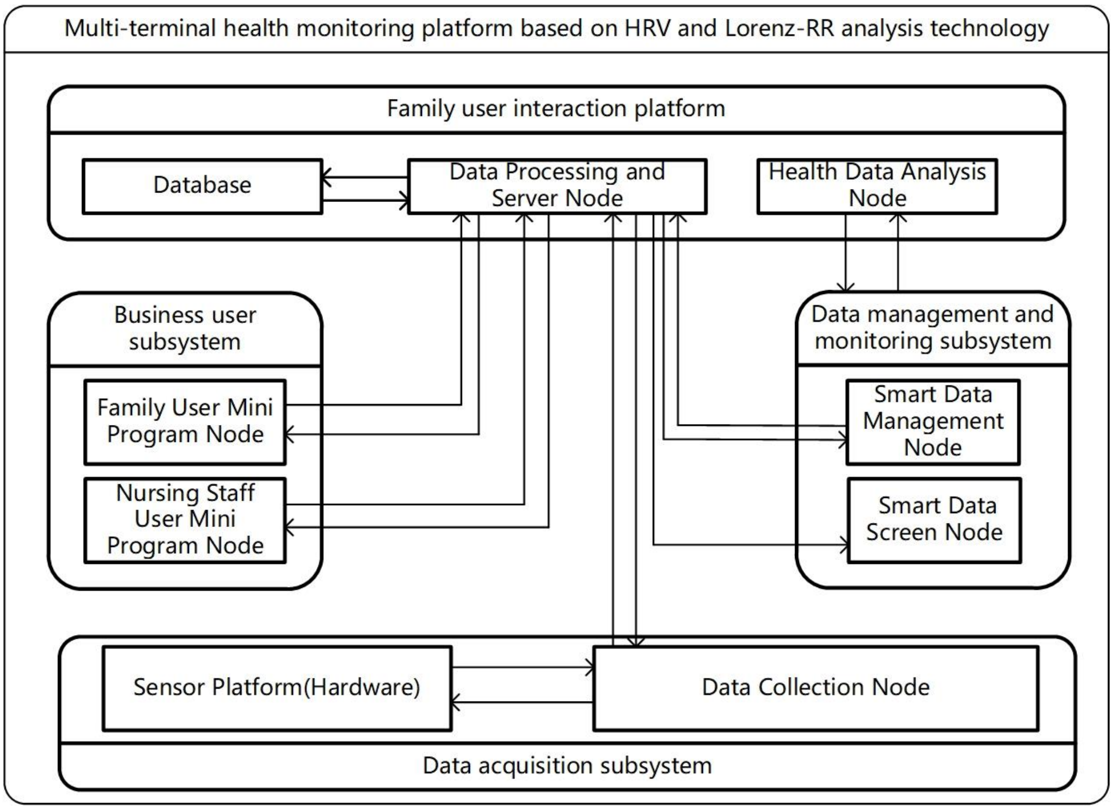

Figure 2. Diagram of Communication Directions Among System Components

For the convenience of future development and understanding, the WeChat mini-program in the family member user interaction platform is hereby named as the Family User Mini-Program Client. The WeChat mini-program in the caregiver user interaction platform is named as the Caregiver User Mini-Program Client. The web front-end program in the administrator user interaction platform is named as the System Data Management Client. The web front-end program in the monitor user interaction platform is named as the Smart Data Dashboard Client. The Java backend program and Python backend program in the data processing server are named as the Data Processing & Service Server and Health Data Analysis Server, respectively. The hardware part and Android app program in the device wearer user interaction platform are named as the Sensor Platform and Data Collection Client, respectively.

4. Lorenz-RR ScatterPlot Classification Algorithm

4.1. Lorenz ScatterPlot and Its Common Types

Lorenz-RR usually refers to Lorenz scatterplot, in this paper, Lorenz-RR refers to the coordinate data of each point in the Lorenz scatterplot. HRV is the full name of "Heart Rate Variability" in Chinese, which refers to the variation of the difference between each heartbeat cycle, in simple terms, it is the variation of the heartbeat speed and slowness. Each point in the Lorenz-RR consists of two RR intervals, and the absolute value of the interpolated values of two adjacent RR intervals is HRV, where RR interval is the time interval between two heartbeats. the horizontal coordinate (X-axis) of each point in the Lorenz scatterplot is the value of the first RR interval, and the vertical coordinate (Y-axis) is the data of the second RR interval [5].

Based on the concept of HRV and RR interval, we can derive the HRV data and Lorenz-RR data from the RR interval data, where the relationship between HRV and RR interval data is as follows.

$$HRV_n = |RR_{n+1} - RR_n|$$

DOI: 10.54254/2753-8818/31/20241144

In the above formula, n represents the number of monitoring times. HRV_n represents the heart rate variability obtained after the nth monitoring, and RR_n represents the interbeat interval data obtained after the nth detection.

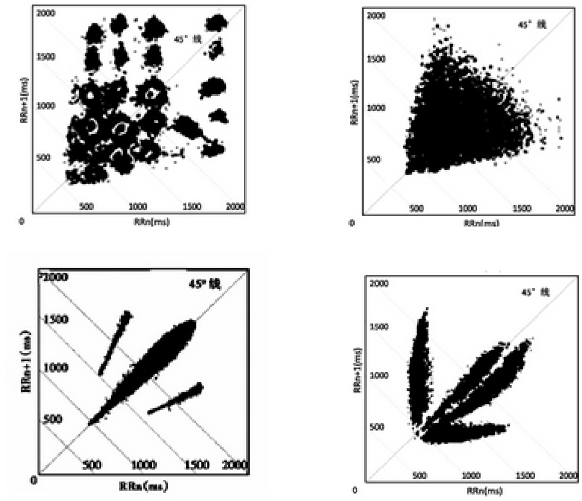

Figure 3. Sample Lorentz Scatterplot

From the n RR interval numerical type data, we can get to n-1 Lorenz-RR coordinate point data, where each Lorenz-RR coordinate point data we can get from the following equation:

$$\text{LorenzRR}_n = \{(X, Y) \mid X = RR_n, Y = RR_{n+1}\}$$

X and Y in the above equation represent the values of the horizontal and vertical coordinates of each point in the Loren scatterplot [6]. After summarizing the existing related studies, we classify the common Lorentz scatterplots into 13 categories, whose names are shown in the table below.

Table 1. 13 Types of Lorentz Scatterplots

Index	Name
1	Typical Network Distribution
2	Typical Sector Distribution
3	Typical Abnormal Three/Quad Distributions
4	Typical Normal Distribution
5	Typical Abnormal Distribution
6	Abnormal Distribution (Forked Shape)
7	Abnormal Distribution (Fusiform)
8	Abnormal Distribution (Grenade Shape)
9	Abnormal Distribution (Rocket Shape)
10	Abnormal Distribution (Short Rod Shape)
11	Abnormal Distribution (Rice Grain Shape)
12	Abnormal Distribution (Long Rod Shape)
13	Abnormal Distribution (Torpedo Shape)

Of these, 1-3 are collectively referred to as irregular distribution plots in the non-normal case, 5-13 are collectively referred to as separate distribution plots in the non-normal case, and only 4 is the distribution plot in the normal case.

4.2. Selection of Classification Algorithm

In the preliminary design of the Lorenz-RR scatterplot classification algorithm, we conducted experiments and comparisons involving machine learning algorithms such as decision trees, as well as deep learning models like LeNet, AlexNet, and VGG16. After thorough consideration, we ultimately decided to adopt the AlexNet deep learning network model for the classification of Lorenz scatterplots [7]. This section will provide detailed explanations of the advantages and disadvantages of each algorithm model in this design described in this paper, along with the specifics of the final Lorenz-RR scatterplot classification algorithm.

Concerning the decision tree algorithm, it requires manual classification of a large number of sample cases and summarization of medical rules. Additionally, at least 50 attributes need to be established during the feasibility analysis to achieve a 5-class classification. Faced with the requirement of a 13-class classification in our scenario, due to the massive quantity of sample cases needed and the difficulty in subsequent model improvements, we ultimately chose to abandon the decision tree algorithm.

Table 2. Experimental data of LeNet, AlexNet, and VGG16 algorithm models

Algorithm model	Number of samples	Training set accuracy	Verification set accuracy	Notes
LeNet	20-50	80%-95%	About 50%	30% of the sample is only used for testing
LeNet	500-1000	10%-30%	Less than 20%	30% of the sample is only used for testing
AlexNet	20-50	50%-70%	About 20%	30% of the sample is only used for testing
AlexNet	500-1000	99.5%-100%	99.5%-99.9%	30% of the sample is only used for testing
VGG16	20-50	30%-80%	About 10%	30% of the sample is only used for testing
VGG16	500-1000	95%-100%	90%-99%	30% of the sample is only used for testing

Regarding the deep learning models LeNet, AlexNet, and VGG16, we collected 50 original data images and conducted experiments by augmenting the original dataset to 550 images using the data augmentation algorithm described in section 4.4. The experimental results are shown in Table 2.

When the training dataset is relatively small, LeNet exhibits a stable and acceptable accuracy on the training set. However, its accuracy on the validation set is only 50%, which is not satisfactory. Moreover, it proves to be ineffective in distinguishing Lorenz scatterplots with similar symptoms. In the case of AlexNet and VGG16 models, the accuracy on the training set cannot stabilize and reaches a minimum of only 30%, indicating potential underfitting. The insufficient training set fails to effectively fill the network, preventing it from achieving meaningful learning. With a training dataset size ranging from 500 to 1000, LeNet demonstrates overfitting even after just one training epoch. Furthermore, it struggles to differentiate Lorenz-RR scatterplots with similar symptoms, making LeNet unsuitable for situations with a large training dataset.

In contrast, AlexNet and VGG16 models perform well when the training dataset is substantial. After 30 training epochs, AlexNet usually converges to a stable loss level, achieving accuracy rates of over 99% on both the training and validation sets, often reaching 100% on the training set. For VGG16, after 30 to 50 training epochs, both the training and validation set accuracies exceed 90%. However, there is still evidence of underfitting after repeated experiments. Therefore, it can be preliminarily concluded that, in scenarios with a training dataset size ranging from 500 to 1000, AlexNet is more suitable for Lorenz scatterplot classification.

After confirming the selection of AlexNet, the next step is to discuss the algorithm model. The composition of the AlexNet algorithm model is shown in Figure 4, with the upper and lower groups representing two computing GPUs. In common algorithm implementations, only one GPU is often utilized for computation.

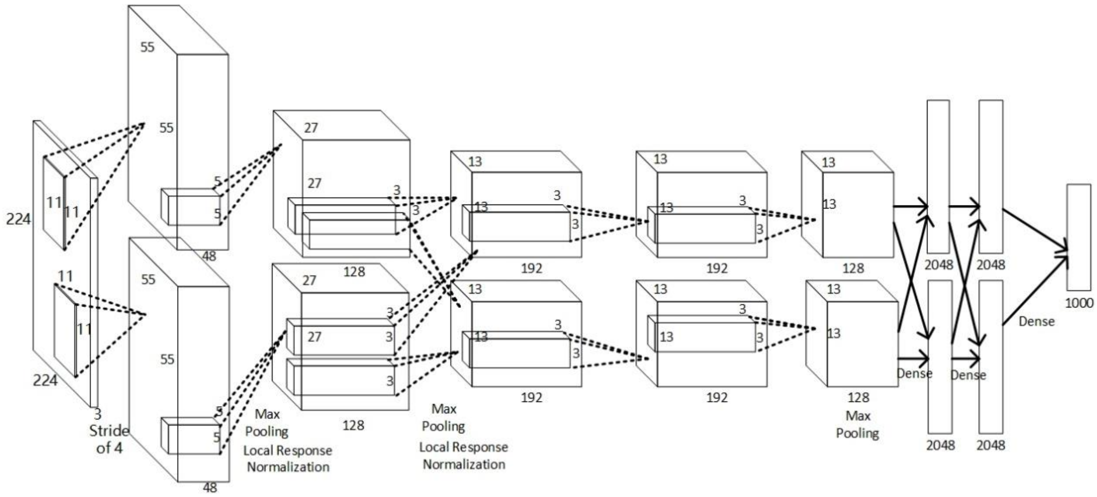

Figure 4. AlexNet Algorithm Model Structure Diagram

In the original model, the input is a 224x224x3 RGB image. It starts by applying a convolution operation with a 4x4 stride and an 11-sized kernel to the image, resulting in 96 feature layers with an output shape of (55, 55, 96). After processing, the model outputs 48 feature layers. Subsequently, the model utilizes a max-pooling layer with a stride of 2 for pooling, leading to an output shape of (27, 27, 96).

In the second layer, a convolution operation with a 1x1 stride and a 5-sized kernel is applied, resulting in 256 feature layers with an output shape of (27, 27, 256). After processing, the model outputs 128 feature layers. The model then uses a max-pooling layer with a stride of 2, resulting in an output shape of (13, 13, 256).

In the third layer, a convolution operation with a 1x1 stride and a 3-sized kernel is applied, resulting in 384 feature layers with an output shape of (13, 13, 384). After processing, the model outputs 192 feature layers.

In the fourth layer, a convolution operation with a 1x1 stride and a 3-sized kernel is applied, resulting in 384 feature layers with an output shape of (13, 13, 384). After processing, the model outputs 192 feature layers. Subsequently, a max-pooling layer with a stride of 2 is applied, resulting in an output shape of (6, 6, 256).

Finally, the model employs two fully connected layers with 2048 neurons each, resulting in an output of 1000 classes.

4.3. The Lorenz-RR Scatterplot Classification Algorithm Based on the Improved AlexNet Model

The system designed in this paper is only used for single-color scatterplot classification, so the final model structure only involves a single GPU for single-channel operation. The Lorenz-RR scatterplot classification algorithm in this design described in this paper is based on the improvement of the AlexNet algorithm model. The improved algorithm uses only one GPU for computation, and the specific improvements are as follows:

1.Change the input size of the images in the input layer to 128x128, and set the type of input images to grayscale.

2.Adjust the size of the convolutional kernels in the first layer of the AlexNet network.

3.Adjust the size of the convolutional kernels in the fourth layer of the AlexNet network.

4.Adjust the density parameters in the two fully connected layers of the network.

5.Adjust the output classifier of the network to a softmax with 13 categories.

To obtain stable conclusions, before each training, the seed is set to shuffle the order of the training and validation sets. In cases where the total sample size is greater than 500, dynamically allocate 20% of the samples as the validation set, which is not involved in training, and use the remaining 80% as the training set. For the learning rate decay rule, the algorithm is set to decrease the learning rate if the accuracy does not decrease continuously for three times. Regarding training results, the algorithm adopts a method of saving every three epochs to avoid missing well-performing training results. An early stopping mechanism is introduced, meaning that when val_loss does not decrease continuously, the model has basically completed training and can be stopped early.

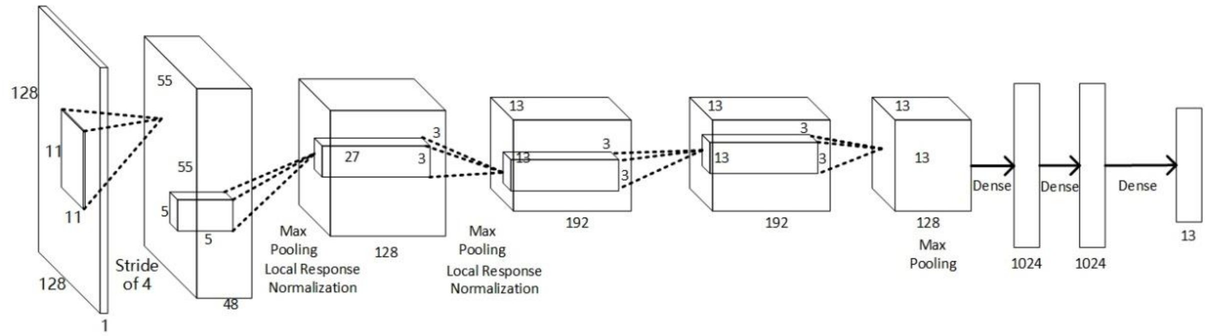

Figure 5. AlexNet Algorithm Model Structure Diagram(Improved)

4.4. Data Set Expansion Strategy

The dataset obtained in this system ultimately consists of Lorenz scatterplots in image form. Therefore, a combination of various image processing methods has been employed for dataset augmentation. Common processing methods include horizontal flipping, vertical flipping, cropping, angle-based rotation, Gaussian blur, regular blur, edge enhancement, sharpening, smoothing, color inversion, and detail enhancement. As the features of Lorenz scatterplots are not only related to the shape formed by all points but also associated with the position of this shape in the coordinate system, augmentation methods like horizontal and vertical flipping are not applicable. Similarly, since the coordinate system is a crucial reference, care should be taken during cropping to ensure the retained image maintains the complete coordinate system. The dataset augmentation algorithm used in this design described in this paper is an 11-fold augmentation algorithm, and its specific process is outlined below.

Step 1: Read all image format files in the specified directory.

Step 2: Read the nth image from the directory into memory.

Step 3: Apply random cropping to the image and output the processed image.

Step 4: Rotate the image at different angles (-2°, -1°, 1°, 2°) and output the processed images.

Step 5: Apply Gaussian blur to the image and output the processed image.

Step 6: Apply regular blur to the image and output the processed image.

Step 7: Apply edge enhancement to the image and output the processed image.

Step 8: Apply sharpening to the image and output the processed image.

Step 9: Apply smoothing to the image and output the processed image.

Step 10: Apply detail enhancement to the image and output the processed image.

Step 11: Return to Step 2 to read the next image. In the above process, random cropping in Step 3 involves randomly cropping 1%-10% of pixels from each of the image's four edges. Rotation in Step 4 involves rotating the image at four different angles: -2°, -1°, 1°, and 2°. This type of dataset augmentation method is generally suitable for deep learning algorithms that classify scatterplots, bar charts, line graphs, and similar graphical representations.

DOI: 10.54254/2753-8818/31/20241144

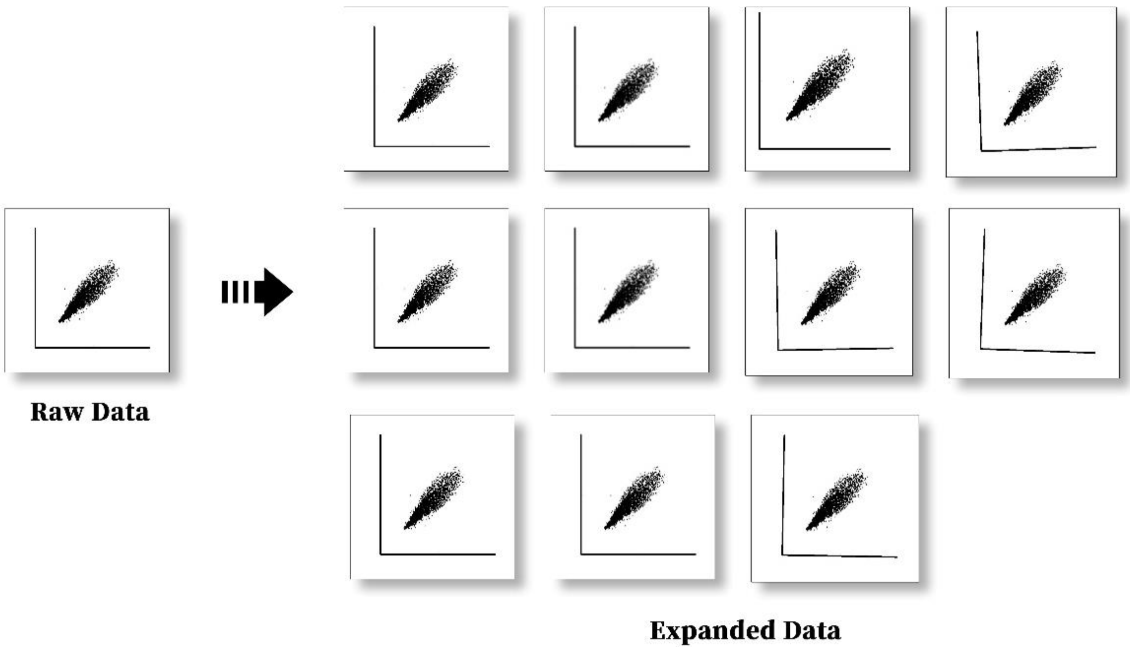

Figure 5. Expansion Diagram of Raw Data

5. Experimentation with Health Data Analytics End Functions

The most important function of the health data analysis end is the Lorenz scatterplot classification. The system provides an API for receiving the coordinate point data of the Lorenz scatterplot, and then the system automatically generates the Lorenz scatterplot based on the coordinate point data. The system then automatically generates a Lorenz scatterplot based on the coordinate point data. Finally, it enters the AlexNet network to classify and return the results based on the scatterplot.

5.1. Classification Algorithm Model Training and Testing

In this paper, we tested the existing dataset with the improved AlexNet model on Lorenz scatterplot classification with the following testbed information:

Table 3. Testbed Information Table

Unit	Model
CPU	AMD Ryzen 7 5800H with Radeon Graphics
GPU	NVIDIA GeForce RTX 3060 Laptop GPU
RAM	DDR4 3200 MHz 64G (Dual Channel)

The dataset used for training and validation was categorized into 13 classes, totaling 68 original samples, which were expanded to 748 data samples by the Data Set Expansion Strategy described in Section 4.4 of this paper. 80% of the data samples are used for model training and 20% are used for model validation, the specific distribution of data samples is shown in Table 4.

DOI: 10.54254/2753-8818/31/20241144

Table 4. Data Sample Distribution Table

Index	Name	Original Sample Size	Expanded Sample Size
1	Typical Network Distribution	4	44
2	Typical Sector Distribution	7	77
3	Typical Abnormal Three/Quad Distributions	17	187
4	Typical Normal Distribution	17	187
5	Typical Abnormal Distribution	7	77
6	Abnormal Distribution (Forked Shape)	2	22
7	Abnormal Distribution (Fusiform)	2	22
8	Abnormal Distribution (Grenade Shape)	2	22
9	Abnormal Distribution (Rocket Shape)	2	22
10	Abnormal Distribution (Short Rod Shape)	2	22
11	Abnormal Distribution (Rice Grain Shape)	2	22
12	Abnormal Distribution (Long Rod Shape)	2	22
13	Abnormal Distribution (Torpedo Shape)	2	22

In order to obtain better model performance, we set different values for the three main parameters, Batch Size, Epochs, and Learning Rate, during the model training and validation process. Specifically, we tested the cases of Batch Size of 10, 20, and 50, Epochs of 50 and 100, and Learning Rate of 1e-3, 1e-4, 1e-5, and 1e-6, respectively.

Since Epochs of 100 can reflect the situation when Epochs is 50, in this paper, we will only show the relevant data when Epochs is set to 100.

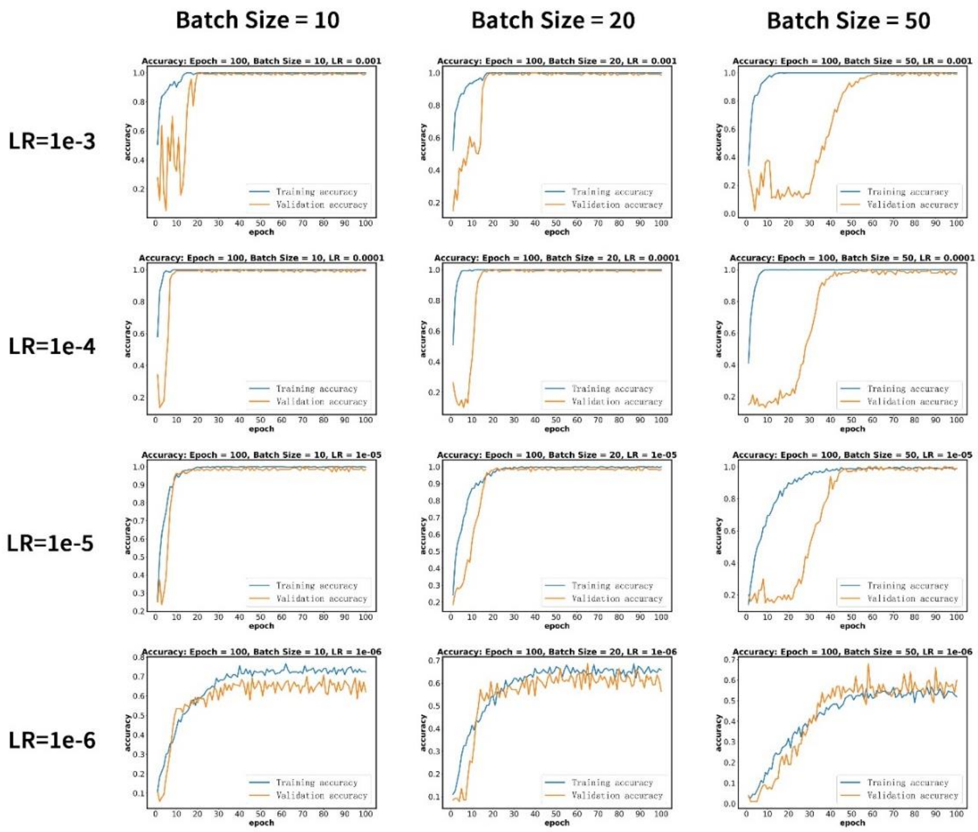

Figure 6. Training and Validation Metric Graphs (Accuracy)

When the Batch Size is 10-50 and the Learning Rate is between 1e-3 and 1e-6, the accuracies of the model training and validation process on the training and validation sets are shown in Figure 6.

At Learning Rate of 1e-4 vs. 1e-5, the model converges faster on the validation set and does not oscillate; when the Learning Rate is 1e-3 the model shows significant oscillations in accuracy during the training process; and when the Learning Rate is 1e-6 the model converges slowly, and at 100 Epoch it still fails to obtain a better model.

Since there are only 748 samples for training and validation, the accuracy of the model decreases at the same Epoch as the Batch Size increases, resulting in slow convergence of the model accuracy. In the process of model training and validation, when the Batch Size takes the value of 10, it can make the model get higher accuracy rate more first and improve the training efficiency.

In order to get the model with the smallest gap between the prediction result and the real result, we analyze the loss values in the training and validation process. When the Batch Size is 10-50 and the Learning Rate is between 1e-3 and 1e-6, the loss on the training and validation sets during the training and validation process of the model is shown in Figure 7.

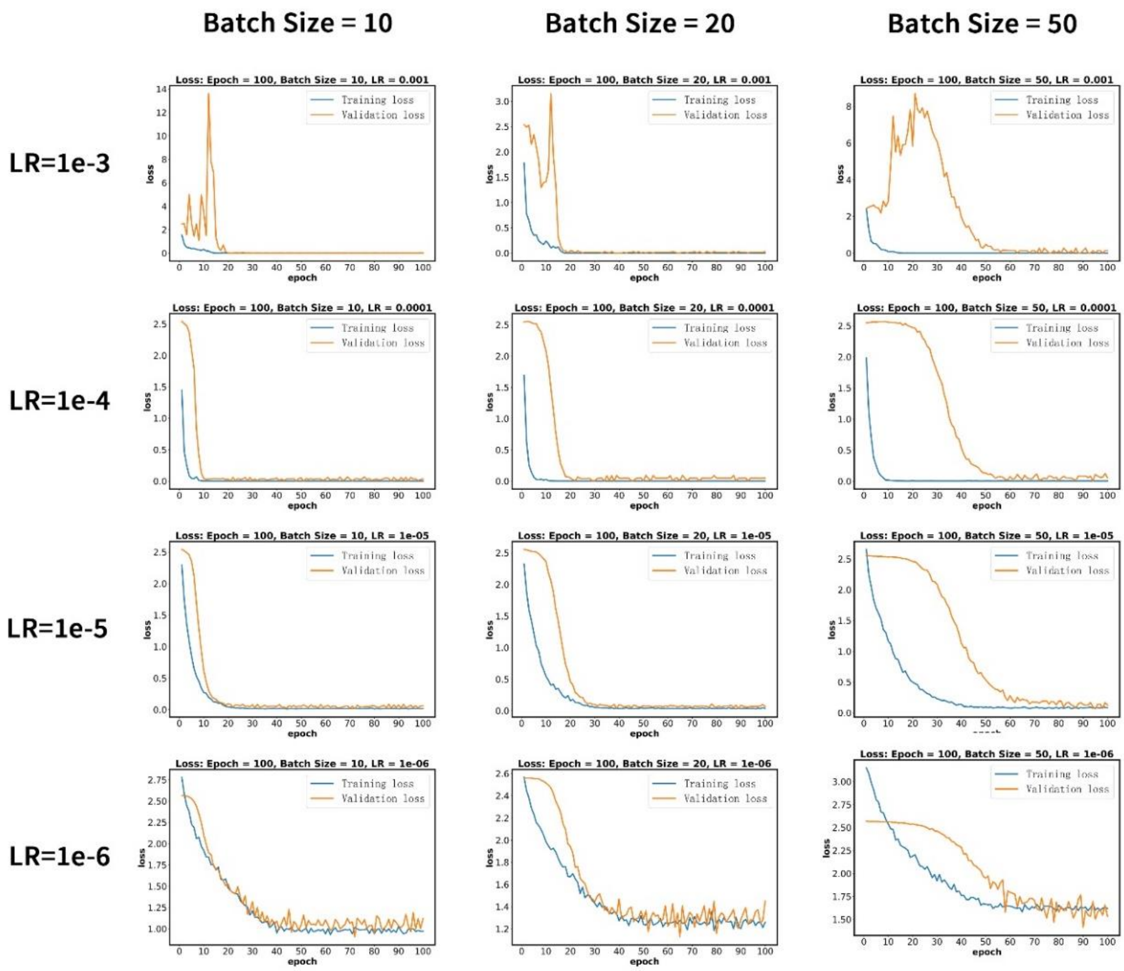

Figure 7. Training and Validation Metric Graphs (Loss)

According to the line graphs of the loss values on the training and validation sets during the training and validation process of the model, it can be concluded that the model can get better performance when the Learning Rate is 1e-4 vs. 1e-5 and the Batch Size is 10, and the loss values can converge faster on both the training set and the test set.

To summarize, at Learning Rate of 1e-4 and Batch Size of 10, the model can be optimized in terms of accuracy.

Finally in the system described in this paper, we have used the parameters of Learning Rate of 1e-4 with Batch Size of 10 to train the model. The accuracy and loss on the training set and validation set during model training and validation are shown in Figure 8. According to the relationship between Epoch and accuracy and loss respectively, we adopt the model obtained when Epoch is 13 during the training process to ensure the performance and accuracy of the system in practical applications.

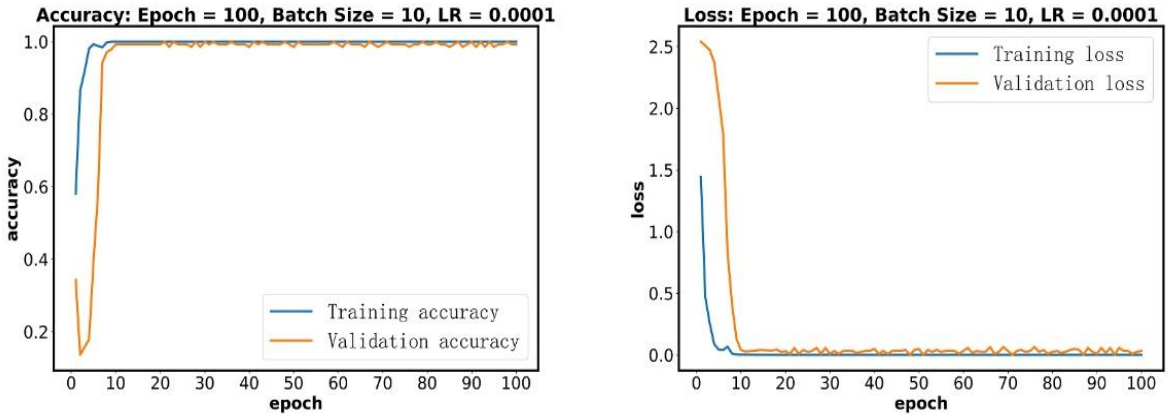

Figure 8. Training and Validation Metric Graphs (Batch Size=10 and Learning Rate=1e-4)

5.2. System Validation

Based on the data received from the interface, the PyPlot library can directly generate a Lorenz scatterplot of the specified style, as shown in Figure 9.

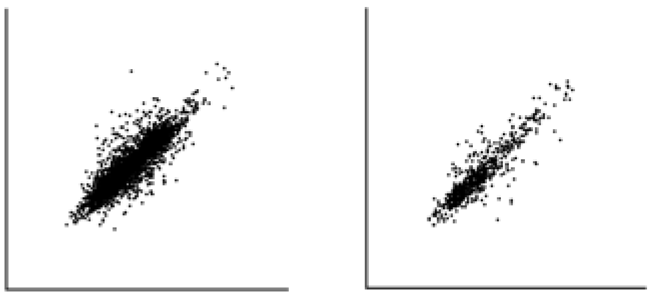

Figure 9. Style of the Generated Lorenz Scatterplot

Since Lorenz scatterplots with the same shape but different locations have different medical interpretations, the left-left, bottom box line needs to be retained for system localization when generating images. In the center of the frame line is a graph composed of individual points.

The final classification effect is shown in Figure 10.

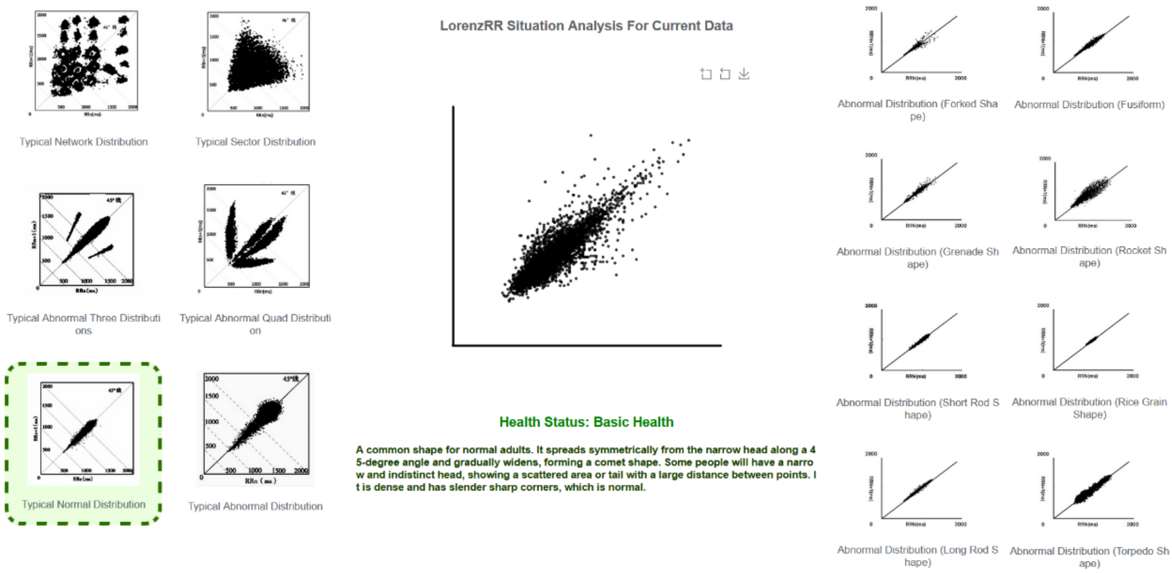

Figure 10. Lorenz-RR Report Example Chart(Normal)

The system displays the results of the classification in the interface and automatically marks the reference legend of the type of Lorentzian scatterplot with a green or red box for the healthcare provider's reference. A message is also provided to inform the administrator or healthcare professional of the possible medical condition of the wearer.

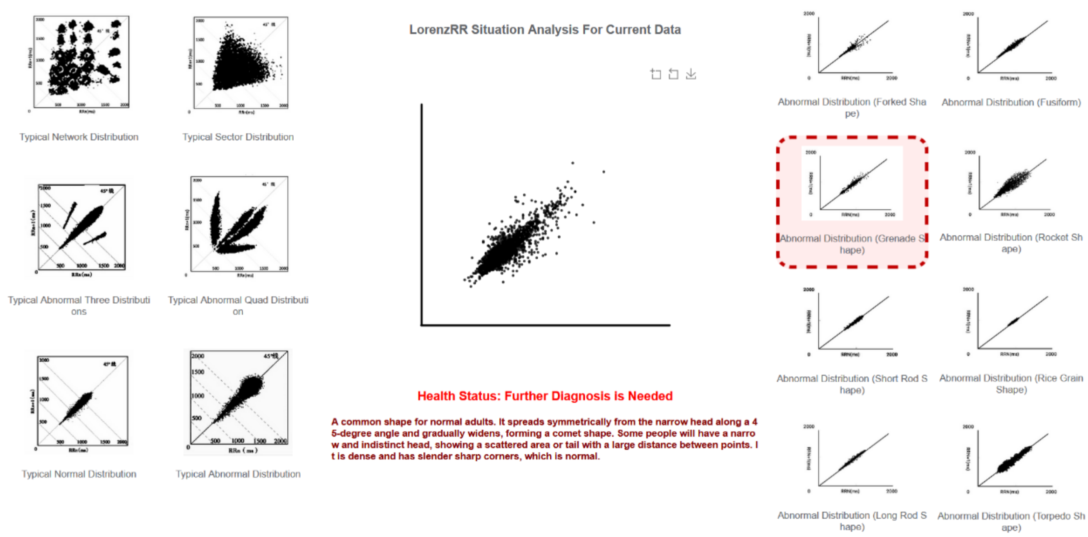

Figure 11. Lorenz-RR Report Example Chart(Abnormal)

As shown in Figure 11, the Lorentz scatterplot classification for the 13 abnormal conditions, the system automatically labeled the example samples of Lorentz scatterplot of Grenade Shape according to the classification results.

6. Conclusion

This design described in this paper leverages digital technology to create and implement a solution for a new digitalized social organization system, establishing a blended online and offline intelligent elderly care model to enhance the efficiency of contemporary traditional social organizations. By collecting user health data and caregiver service data, the design described in this paper further analyzes the real needs

DOI: 10.54254/2753-8818/31/20241144

of social organizations, service recipients, social workers, and caregivers. It addresses challenges in the elderly care industry, including low levels of medical care, a shortage of caregiver talent, inadequate service supervision, insufficient resource integration capabilities, and a lack of accurate positioning and reasonable planning for the elderly care service industry.

Simultaneously, the system explores a new type of intelligent elderly care model applicable to the construction of intelligent social workstations. The service recipients are the elderly and their families, while the service providers include caregivers and social workers. In the specific implementation process, the design described in this paper conducted research on software design methods, ECG and PPG sensor technologies, HRV and Lorenz-RR data analysis technologies, simulated health monitoring and warning schemes, and researched health data display solutions for non-professional groups. This addressed five key issues in the system, ultimately achieving a multi-terminal health monitoring platform based on HRV and Lorenz-RR analysis technology that meets the expected requirements.

This paper is limited by the objective conditions and data sources and other factors, the number of samples of the data set collected has to be supplemented, and it is hoped that scholars can further expand the classification of the Lorenz scatterplot and the classification samples in the future. At the same time, this paper does not involve clinical research and other medical fields, and only uses technical means to classify the known Lorenz scatterplot according to its external characteristics, and all the results are for reference only, and cannot be used as the basis for disease judgment. For accurate conclusions, further diagnosis and analysis by professionals in related medical fields are required.

References

[1] WANG Lijian,ZHU Yixin,MA Wei. Realistic demand and development progress of intelligent healthy aging industry[J/OL]. Journal of Xi'an Jiaotong University (Social Science Edition):1-14[2023-12-09].

[2] Shi Lingyun. Poverty caused by disease and poverty alleviation in rural areas[J]. China Rural Health,2021,13(14):71-72.

[3] Yuan Han, Qingqing Song, Xinquan Huang, and Jie Yin. 2021. Hierarchical Software Design Methodology for High Concurrency and Fine-Grained Permission Control Scenarios. In Proceedings of the 2021 2nd International Conference on Control, Robotics and Intelligent System (CCRIS '21). Association for Computing Machinery, New York, NY, USA, 122–128.

[4] Frain B .Responsive Web Design with HTML5 and CSS3[J].Packt Publishing Limited, 2012.

[5] Tianshan University. A nonlinear method for analyzing heart rate variability[EB/OL].[2019-02-25].https://www.tsu.tw/edu/11482.html.

[6] Tianshan University. A comparative study of Lorenz scatterplot and AECG diagnosis [EB/OL].[2019-02-25].https://www.tsu.tw/edu/11490.html.

[7] Michelucci U. Advanced applied deep learning: convolutional neural networks and object detection[M]. Apress, 2019.