Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 6561730, 12 pages
https://doi.org/10.1155/2021/6561730

Research Article

Design and Implementation of Embedded Real-Time English Speech Recognition System Based on Big Data Analysis

Lifang He,[1] **Gaimin Jin**,[1] **and Sang-Bing Tsai**[2]

[1]*Shijiazhuang University of Applied Technology, Shijiazhuang, Hebei 050000, China*
[2]*Regional Green Economy Development Research Center, School of Business, WUYI University, Nanping, China*

Correspondence should be addressed to Lifang He; clare2021@126.com and Sang-Bing Tsai; sangbing@hotmail.com

Received 14 July 2021; Revised 29 July 2021; Accepted 25 August 2021; Published 2 September 2021

Academic Editor: Xianyong Li

This article uses Field Programmable Gate Array (FPGA) as a carrier and uses IP core to form a System on Programmable Chip (SOPC) English speech recognition system. The SOPC system uses a modular hardware system design method. Except for the independent development of the hardware acceleration module and its control module, the other modules are implemented by software or IP provided by Xilinx development tools. Hardware acceleration IP adopts a top-down design method, provides parallel operation of multiple operation components, and uses pipeline technology, which speeds up data operation, so that only one operation cycle is required to obtain an operation result. In terms of recognition algorithm, a more effective training algorithm is proposed, Genetic Continuous Hidden Markov Model (GA_CHMM), which uses genetic algorithm to directly train CHMM model. It is to find the optimal model by encoding the parameter values of the CHMM and performing operations such as selection, crossover, and mutation according to the fitness function. The optimal parameter value after decoding corresponds to the CHMM model, and then the English speech recognition is performed through the CHMM algorithm. This algorithm can save a lot of training time, thereby improving the recognition rate and speed. This paper studies the optimization of embedded system software. By studying the fixed-point software algorithm and the optimization of system storage space, the real-time response speed of the system has been reduced from about 10 seconds to an average of 220 milliseconds. Through the optimization of the CHMM algorithm, the real-time performance of the system is improved again, and the average time to complete the recognition is significantly shortened. At the same time, the system can achieve a recognition rate of over 90% when the English speech vocabulary is less than 200.

1. Introduction

English speech recognition is a branch of pattern recognition, which is an interdisciplinary subject integrating microelectronics, communications, computers, automation, and acoustics [1, 2]. The most important technology of English speech recognition is the construction of speech signal processing technology and training model. The ultimate goal is to hope that humans can communicate with computers. This kind of human-computer dialogue scenes often appears in science fiction movies. In fact, this is a very complicated technology. In addition to English speech signal processing technology, how to recognize sounds and understand what is said is challenging. Due to the discontinuity of English speech, the difference in accent and pitch of each person, and the difference in speaking speed and volume, these factors will increase the difficulty of English speech recognition, so English speech recognition has always been regarded as a challenging subject. Early English speech input must separate the relationship between each word clearly and input each word separately, which is different from what we usually say. The object to be recognized is also the recognition of a specific person, which is not applicable to other recognizers.

With the rapid development of semiconductor technology, the continuous increase in the scale of integrated circuits, and the continuous improvement of various development technology levels, English speech recognition technology has gradually become smaller after being combined with embedded systems based on DSP, FPGA, ASIC,

and other devices. With the development of industrialization and practicality, the application field is also getting bigger and bigger [3]. As a modern information technology with extensive social and economic benefits, English speech recognition has made certain achievements, but there are still a series of problems when facing practical use. The mature technology and reliable performance of English speech recognition systems are still available at home and abroad [4]. There is a lot of research space and market potential, and there is still a lot of room for improvement in terms of recognition accuracy, speed, robustness, and system miniaturization. In order to achieve an English speech recognition system with excellent performance, on the one hand, it is necessary to study the theory and algorithm of English speech recognition to solve and improve various problems in the recognition process. On the other hand, it is also necessary to consider simplifying the complexity of the system [5].

This paper compares the implementation schemes of the system and chooses to implement the English speech recognition system by way of SOPC. This article analyzes the requirements of the system, introduces the overall design of the system, and points out the components and software that a complete English speech recognition control system should include and the functional modules and software that the SOPC system should have. This paper divides the software and hardware that constitute the system and uses hardware to accelerate the algorithm in the most computationally expensive part of the software algorithm. At the same time, this article also introduces the selection of peripheral devices of the English speech recognition control system, the selection of processors, memories and buses that make up the SOPC system, and so on. This article selects the MicroBlaze soft processor core as the system processor, uses BRAM as the data storage memory, connects the data storage memory and other peripherals using the OPB bus, and connects the on-chip memory using the LMB bus. According to the large dependence of the training CHMM on the initial value and the large amount of calculation, an improved algorithm, Genetic Continuous Hidden Markov Model (GA_CHMM), is proposed. From the specific process of English speech signal preprocessing and endpoint detection, feature extraction, English speech recognition training, and recognition, the English speech recognition system based on the improved algorithm is explained. This paper tests the embedded English speech recognition system based on DSP and DHMM. For PC auxiliary software, the function of each module is mainly tested; after testing, each module can work normally. For the testing of embedded system software, the main indicators tested in this article are system recognition rate and real-time response. After testing, when the vocabulary of the system is 100, the recognition rate of the system can reach about 90%. After the optimization of fixed-point and CHMM algorithm, the average real-time response speed of the system has been improved.

2. Related Work

In order to ensure reproducible test results, avoid errors introduced by human factors, and improve test quality, the performance evaluation of embedded English speech recognition systems urgently needs to introduce automated test tools to replace heavy manual testing [6]. At present, the mainstream English speech recognition automated test method at home and abroad uses TTS technology to convert a text file containing test keywords into an English speech file and then uses a playback device to play the English speech file to perform English speech recognition. The test tool monitors the English speech recognition system in real time, obtains the recognition result, and records it to a file. Automatically they call the result statistics tool, compare the recognition results and the marked files, and determine whether the recognition results are correct or not. After the identification is completed, automatic summary statistics are performed, and a CSV file is output for easy viewing by testers [7].

The domestic research work on embedded software testing technology started from the field of defense electronics. Nanjing University, Beijing University of Aeronautics and Astronautics, and Aviation Industry Corporation have successively developed a number of testing tools and testing systems, whose main role is to cover embedded systems. After that, the employees of China State Shipbuilding Corporation conducted in-depth research on the GUI automated testing technology of embedded software and proposed a nonintrusive embedded GUI automated testing framework [8, 9]. Others proposed testing based on the characteristics of ship embedded software [10]. Scholars studied the UML-based embedded software test case generation technology [11]. Domestic embedded software testing tools generally target specific embedded operating systems, and most of them test for code coverage. It is difficult to have a clear understanding of the overall performance of embedded software.

The recognition performance test of the embedded English speech recognition system is carried out in the system test stage [12]. Based on the black box test method, the English speech recognition system is placed in the test environment (host machine environment or target machine environment) to simulate actual use and operation. Recognition tests are performed on batches of English speech data, and then the recognition performance indicators are counted based on the recognition results. At present, there are two main methods for testing embedded English speech recognition systems at home and abroad: playing test audio and speaking on the spot. Before the test, record the English speech data for the test in advance to prepare the noise data. In actual operation, a tester is required to operate the playback equipment, one tester to operate the English speech recognition system, and the test results greatly waste human resources and introduce errors in manual operation [13]. On-site oral calls are organized to organize multiple testers to read the test corpus of the English speech recognition system and test the effect. This method is too random and is not conducive to the recurrence of the test, and the way of pronunciation of different emotional infections will also be different, which may cause the instability of the test effect. Both test methods require live broadcast of English voice and noise signals. In the case of heavy test tasks and large

amounts of English voice data to be tested, it will undoubtedly cause a long test time and ultimately lead to the risk of a long product development cycle [14].

Using TTS technology to convert text into English voice can save the complexity of manual recording, but the converted English voice file is too "mechanized" and cannot simulate the tone and speed of human speech. In addition, the existing automated test methods still control the playback of English voice files by humans, and the time interval between each English voice is not flexible, which may lead to a waste of test time. The method is to compare the labeling data and the recognition results, but the labeling of English speech files is generally calculated by humans. Related scholars have proposed new ideas for applying vector quantization technology to English speech coding and recognition [15]. Hidden Markov model theory has become a research hotspot and an important theoretical basis for English speech processing technology [16]. Among them, the most representative is the Sphinx system, which is a nonspecific continuous English speech recognition system built using vector quantization and hidden Markov models. Three obstacles of English speech recognition were solved in the laboratory: nonspecific person, large vocabulary, and continuous English speech. In addition, artificial neural network technology has brought about new opportunities to the wide application of English speech recognition. Related scholars have designed an English speech recognition system based on the principles of the auditory nervous system [17–19]. Although the unique advantages of artificial neural network technology bring about many benefits to English speech recognition, the large amount of computation and long training time make its development relatively slow.

3. FPGA-Based Embedded Real-Time English Speech Recognition System Design

3.1. Hardware Platform Selection. The system design must meet the constraints of performance, cost, function, and so on. The overall design of the system is to divide the large and complex system into several modules according to actual needs and compare the advantages and disadvantages of the system composed of modules [20–22]. This article will compare the widely used embedded English speech recognition system and FPGA-based English speech recognition system in detail. Option one is a common English speech recognition system; the structure diagram is shown in Figure 1.

Solution one uses DSP digital signal processor or ARM processor as the central processing unit and CPLD as the coprocessor. But it is more common to use a DSP processor, because it can better reflect the advantages of digital signal processing. The second scheme uses the SOPC system composed of FPGA to form an English speech recognition system.

From the perspective of cost, the cost of a system with DSP/ARM as the processor is significantly higher than that of an English speech recognition system with FPGA as the core. A system with DSP/ARM not only requires a better processor but also requires a large amount of external equipment. The support of design increases the cost of system implementation. However, FPGA itself has a lot of logic resources, which can save the use of some memory and external logic and make the system miniaturized. From the performance point of view, the performance of the DSP/ARM system of the same price will be worse than that of the SOPC system. The use of functions makes it possible to improve system performance without increasing costs. Huawei, ZTE, and other companies use their own ASIC chips to reduce the cost by at least half compared with imported devices, and the performance obtained is indeed higher than the performance that can be obtained by using DSP platform. These advantages are not possessed by the system composed of DSP/ARM processor.

3.2. Design of English Speech Recognition System. The FPGA-based English speech recognition system is an embedded system that integrates software and hardware. It is an independent system that collects English speech, processes English speech data, and finally outputs control commands. In order to complete a larger-scale SOPC system, the design often adopts a top-down (Top-Down) hierarchical design idea. The hierarchical design idea is to divide a larger system into several subsystems, each subsystem is designed independently, and each subsystem is designed to be assembled into a complete system. The wiring between each functional submodule is as few as possible, the interface function is clear, and the scale of the functional module is required to be moderate. This design method greatly reduces the design complexity of system. A single subsystem can be tested separately. The problems that arise only need to modify the internal subsystems without affecting the functions of other subsystems.

System design should follow certain principles, use tools to divide modules, and determine what kind of functional modules the system should have and how to integrate these functional modules. The system design should solve the problem of the overall structure of the system in the hierarchical design of the system, rather than solving the problem of how to realize the partial functions. By organically integrating the divided system modules, the appropriate continuous method is used to maximize the system optimization. The system design uses a modular design method to grasp the framework of the entire system as a whole, so that the system meets the needs of the application. The system design follows the principle from top to bottom, decomposing each function one by one. The overall structure design of the English speech recognition system is shown in Figure 2.

The system architecture design consists of two parts: one is the architecture design of the entire English speech recognition system, and the other is the architecture design of the SOPC system. In Figure 2, the power supply is the energy source of the system, providing stable current and voltage for the system. The analog-to-digital conversion device converts the English voice signal into a digital signal and transmits it to the SOPC chip, so that the SOPC chip can process the English voice signal. In addition to software computing

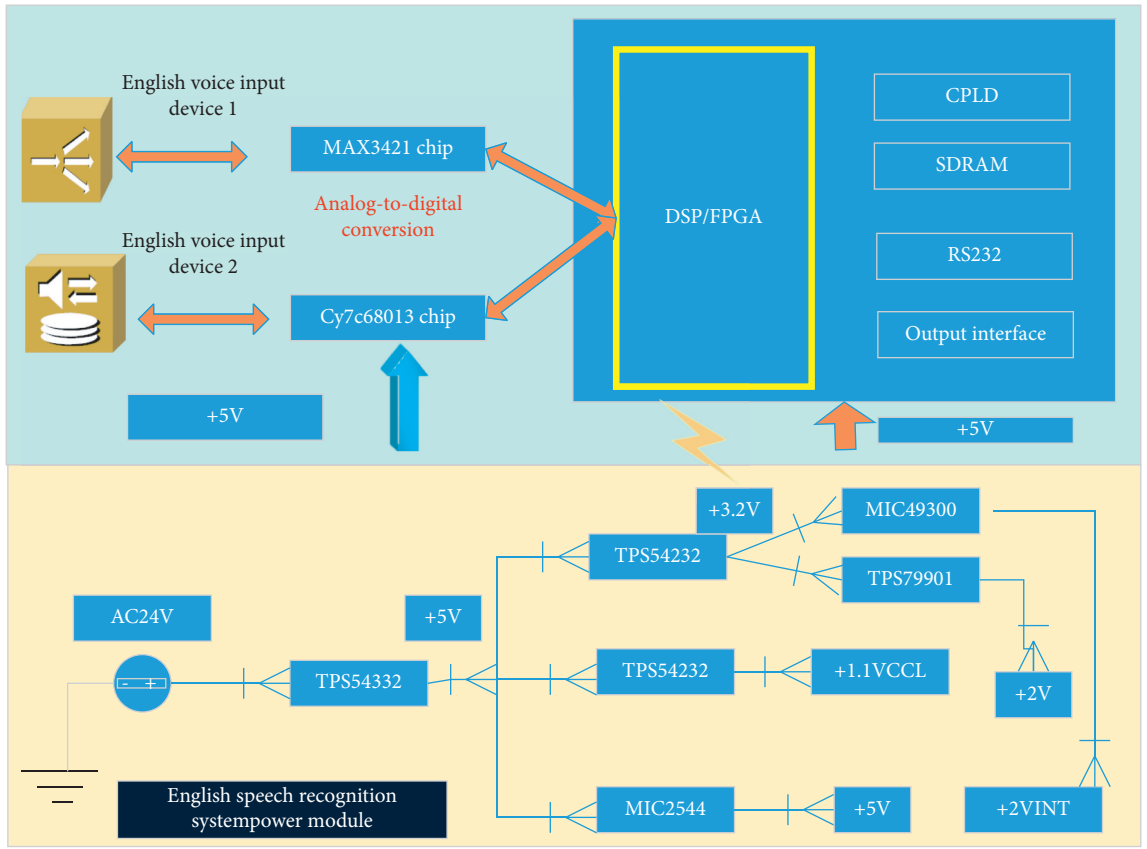

Figure 1: English speech recognition system scheme.

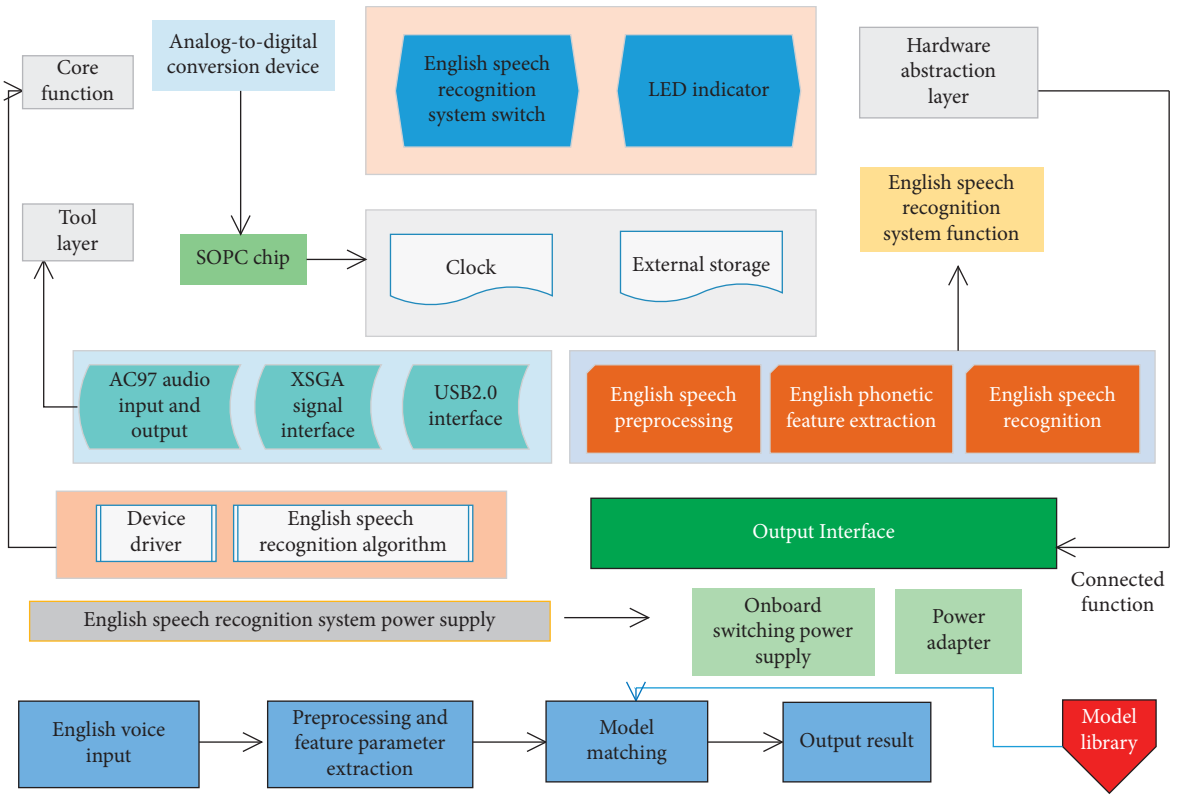

Figure 2: Structure diagram of English speech recognition system.

functions, SOPC chips also include hardware acceleration units. External storage is an indispensable part of a complete SOPC system, used to store models and data, programs, and so on. The output interface is the control signal interface output by the system, which provides the connection function between the system and other equipment.

3.3. SOPC System Design. The overall structure of the hardware design of the SOPC system is shown in Figure 3. It merges most of the devices that implement system functions to form a completed embedded system. The SOPC system accepts the input signal of the digital-to-analog conversion chip and outputs the control signal. The hardware part includes processor core, memory management IP core, memory, and hardware acceleration IP. At the same time, a complete SOPC system should also have an off-chip memory controller and so on.

As an integral part of the English speech recognition project, the part involved in the completion of this article includes program selection, overall system design, and system testing. The independently completed parts include software and hardware division, hardware configuration, English speech recognition algorithm recognition, software and hardware algorithm design optimization, hardware acceleration IP core, and its control logic design.

When designing an SOPC system, the software and hardware of the system are divided by requirements. The division of hardware, software, and hardware of the system is the key to whether the system can meet the needs of users. A 10% system-level design process has an 80% impact on the final cost and performance of the system design. Software is more flexible than hardware and easy to modify. However, because the software is executed programmatically, the speed is slower than the hardware. The use of hardware also has its disadvantages. The hardware method will increase the usage of SOPC system resources. When using FPGA to develop SOPC system, there may be the possibility that resources cannot meet the demand and require the use of higher-performance FPGA chips. If it is finally formed into an ASIC Chip, it may increase the area of the chip, thereby increasing the implementation cost of the system. Therefore, when realizing the goal of the system, the relationship between performance and cost must be considered comprehensively. After the hardware and software division of the system is completed, the designer then conducts the performance evaluation of the system. If the performance requirements are not met, the software and hardware must be redivided. To achieve the performance evaluation design, the hardware designer will consider the hardware physical realization of the system and go through the traditional IC design process, such as logic synthesis, layout planning, timing analysis, placement and routing, and physical verification. Software designers will consider which software operating environment to use, such as whether to use or not to use an operating system and which operating system to use.

There are multiple memory interfaces available on the XUPV2P platform, such as FLASH memory and DDR memory. The development tool provides IP cores for multiple memory controllers, and multiple memory controllers can be added to use these memories. OPB BRAM is faster than DDR memory, BRAM is faster, and the access of software and hardware to BRAM is much more efficient than that of DDR memory. BRAM has two ports, PORTA and PORTB; each port has an independent 32-bit address bus and 32-bit data bus, as well as read and write control lines. When implementing BRAM, four 8-bit RAMB16_s8_s8 devices are used to improve the parallelism of memory access. Only when each device is enabled, access is allowed. In SOPC design, external memory is often used to permanently save FPGA configuration and software data. This is because the contents of the FPGA will be lost after power failure. Therefore, we use external memory to save the configuration information and template data in the FPGA. At the same time, in order to speed up program execution, an on-chip memory of the processor should be set. The data used in the system is mainly divided into two types: one is model data, and the other is characteristic parameter data of samples. In order to facilitate the software and hardware to calculate the data and simplify the control operation, multiple BRAMs are designed in the system to save the model data transferred from the external memory, save the characteristic parameters of the samples, and save the results that need to be written back by the hardware acceleration calculation.

According to different application needs, MicroBlaze has a variety of buses available. The on-chip peripheral bus is a type of Core Connect bus. It is a low-speed bus that can be connected to buses of different widths and devices with different timings. In this article, the OPB bus is used to connect the hardware acceleration IP core control logic, because the OPB bus is mainly used for data transmission to external devices. The OPB bus supports multibit width data, which accepts input from the host when used for peripherals and performs specified operations, which meets the needs of hardware acceleration. OPB bus performance is shown in Table 1.

4. English Speech Recognition Algorithm Design Based on GA_CHMM

4.1. Parameter Selection of Genetic Algorithm. For genetic algorithms, choosing different control parameters will greatly affect the performance of the optimization. For the same encoding method and genetic operator, changes in parameters may cause greater performance changes. The control parameters of the genetic algorithm mainly include the length of the code string L, the population size N, the crossover probability P_c, the mutation probability P_m, and the termination algebra G.

① For length of encoding string L, regardless of whether the encoding method is real number encoding or binary encoding, according to the number of variables in the problem to be optimized and the encoding length L equal, each variable of the problem corresponds to each position in the encoding string. ② As regards group size N, the number of individuals in a group is called the group size. The size of

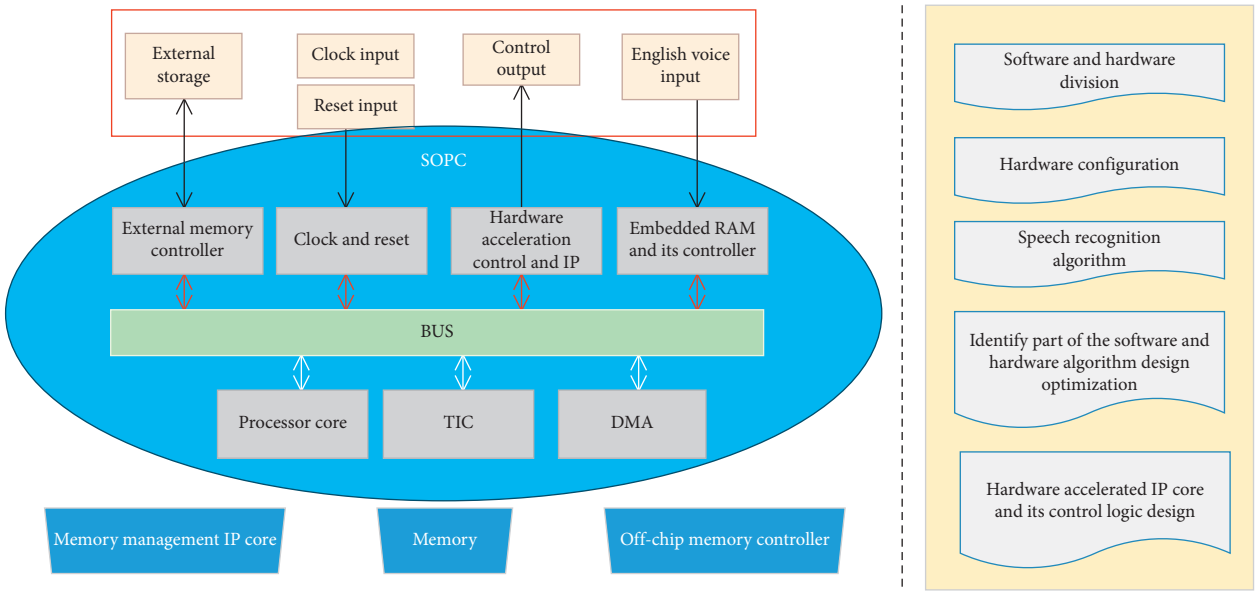

Figure 3: Overall scheme of SOPC system design.

Table 1: OPB bus performance.

Timing	Synchronize
Architecture	Multimaster/slave device
Interconnect	Bus independent read/write data line, does not support tristate
Connect	Multipath
Data line width	8-Bit, 16-bit, 32-bit
Data transfer protocol	Single read/write transmission, support burst transmission, word, byte, half word transmission
Address line width	32-bit

the group size N will affect the final result of optimization and the efficiency of the entire process. If the population size N is too small, the genetic algorithm cannot optimize the problem well; if the population size N is large, the genetic algorithm increases the probability of obtaining the local optimal solution, and the amount of calculation is relatively large, which will affect the optimization process of the entire process. So, depending on the actual situation, the value of N is also different, and generally the value is 20–100. ③ Regarding crossover probability P_c, the purpose of the crossover operation is to inherit the excellent genes in the parent to produce better individuals and get the best solution as possible in the iterative process. The crossover probability determines the search and optimization ability of the genetic algorithm. If P_c is small, the search and optimization ability of the genetic algorithm may fall into a relatively slow and sluggish state and cannot fully inherit the excellent genes; if P_c is large, the superior ability has been enhanced, but this may destroy the overall performance of the chromosome, so P_c is generally taken as 0.25–0.99. ④ For mutation probability P_m, the main purpose of mutation operation is to maintain the diversity of the population. When the mutation frequency P_m is small, although it can prevent the loss of important genes in the population, it reduces the possibility of population diversity. When the mutation frequency P_m is large, although the opportunity for diversity is increased, it may damage excellent individuals. Therefore, P_m is generally 0.0001~0.1. ⑤ For termination algebra G, it is a sign of the end of genetic algorithm. At this time, the best individual of the group is output as the optimal solution of the problem, and G is generally taken as 50~200.

4.2. Preprocessing of English Speech Recognition System Based on GA_CHMM. There are generally two methods for English voice collection: one is to use a hardware circuit system to collect English voice signals; the other is to directly use a multimedia sound card to collect English voice signals, that is, to use a computer sound card to collect English voice signals. The frequency of the English speech signal is within 40~4000 HZ. According to the Nyquist sampling theorem, the original English speech signal frequency is more than twice the sampling frequency for sampling. From the acquisition module, the sampling frequency is 11.025 kHz.

Through the preemphasis operation, the high-frequency part of the English speech signal is improved, and the power-frequency interference is filtered out to obtain a more pure and true English speech signal. Suppose that the English

speech sampling signal at time n is $x(n)$, and the signal obtained after preemphasis is $\hat{x}(n)$; that is,

$$\hat{x}(n) = nx(n) + 0.94x(n-1). \tag{1}$$

The transfer function of the preemphasis filter can be obtained as

$$H(z) = 1 + 0.94Z^{-1}. \tag{2}$$

After preemphasis, the frequency spectrum of the English speech signal is indeed improved in the high-frequency part, while filtering out the power-frequency interference. The English speech signal has short-term stability, so the English speech signal is divided into some equal time periods for analysis and processing, and this process can be achieved by using a movable finite-length window for weighting. Through the comparison of rectangular window, Hanning window, and Hamming window, this paper chooses Hamming window as the windowing function. $N = 256$ English speech samples are one frame. From the acquisition module, we know that the sampling frequency is 11.025 kHz, which can be calculated every time.

4.3. Optimization of HMM Parameters. According to the different description of the statistical characteristics of the observation sequence, there are two main types of HMM: discrete HMM (DHMM) and continuous HMM (CHMM). The biggest difference between the two is mainly in the method of calculating the probability B of the observation sequence. The parameter group B of DHMM is a probability matrix; that is, the probability $b_j(O_t)$ of an observation event produced by each state is satisfied:

$$\prod_{t=0}^{T-1} b_j(O_t) = -1. \tag{3}$$

The parameter group B of CHMM is that each state corresponds to an observation probability density function, which is a Gaussian probability density function:

$$b_j(O_t) = \prod_{m=0}^{M_j} N\left(U_{jm}, O_t, u_{jm}\right) \cdot C_{jm}. \tag{4}$$

CHMM does not need vector quantization. The mean value and variance are calculated from the feature vector after feature extraction, and the observation probability is calculated using the above formula. According to the needs of different practical problems, different HMM models can be selected.

Given the observation sequence O and the HMM model $\lambda = (A, B, \pi)$, the HMM parameter optimization problem is how to adjust the model $\lambda = (A, B, \pi)$ to maximize the probability of the observation sequence output $P(O, \lambda)$; here parameter reestimation is used to adjust the model parameters. The method of parameter reestimation is the Baum-Welch algorithm. The Baum-Welch algorithm continuously reevaluates the parameters through the reevaluation formula until convergence, the output probability $P(O, \lambda)$ is the largest, and then the parameter model obtained at this time $\lambda = (A, B, \pi)$ is the optimal HMM.

Given the observation sequence O and the model $\lambda = (A, B, \pi)$, the probability that the Markov chain is in the S_i state at time t and is in the S_j state at $t+1$ is as follows:

$$\xi_t(i, j) = P\left(S_i = q_t, S_j = q_{t+1}, O|\lambda - 1\right). \tag{5}$$

Then we launch

$$\xi_t(i, j) = a_{ij} b_j(O_t) \frac{\alpha_t(i)\beta_{t+1}(j)}{P\left(S_i, S_j, O|\lambda - 1\right)}. \tag{6}$$

The probability that the Markov chain is in S_i at time t is

$$\xi_t(i) = \prod_{j=0}^{N-1} \xi_t(i, j) = \frac{\alpha_{t+1}(i)\beta_t(i+1)}{P(O|\lambda - 1)}. \tag{7}$$

4.4. The Training Process of English Speech Recognition. The English speech signal is a time series. According to the characteristics of the English speech signal, each word can be represented by a CHMM model parameter. The system adopts the HMM model with $N = 5$ states from left to right without spanning. After the feature parameters of the English speech signal are extracted, a word feature vector is used as the input observation sequence of the CHMM model.

The CHMM training algorithm (Baum-Welch algorithm) is to iteratively calculate the observation sequence of the English speech signal through an estimation formula to obtain a new parameter model. The new parameter model will be better than the old parameter model. Through repeated iterations until the convergence condition is reached, the best CHMM model is obtained. The best model at this time is most likely to be a local optimal solution rather than a global optimal solution. In order to obtain the global best model as much as possible and obtain a better recognition effect, genetic algorithm is introduced in the process of CHMM training; that is, a new training algorithm GA_CHMM is obtained. GA_CHMM algorithm is realized from the following aspects.

4.4.1. Coding Scheme. In the process of applying genetic algorithm to CHMM training, the parameters that need to be optimized in the CHMM model are first arranged to form a chromosome. This paper uses the CHMM model with five states jumping from left to right. The parameters of the CHMM model mainly include the initial state distribution probability π, the state transition matrix probability A, and the probability density function B of each state corresponding to the observation sequence. There are 5 parameters in the initial state distribution matrix π. There are a total of $5 \times 5 = 25$ parameters in the state transition matrix A, and the mixing coefficient matrix C in the probability density function B has $5 \times 5 = 25$ parameters. In this paper, 24 order MFCC coefficients are used. CHMM does not need vector quantization. The feature vector after feature extraction is used to obtain the mean and covariance through the

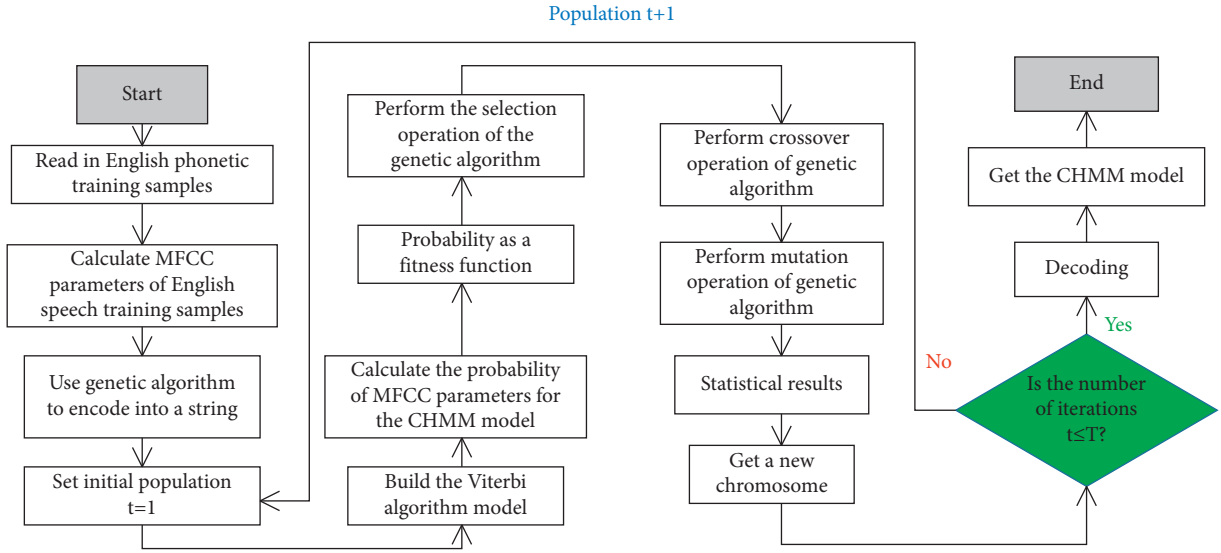

Figure 4: Training flowchart.

probability density function B. Mean μ and Covariance U are $5 \times 5 \times (24 + 24 \times 24) = 15000$ parameters which are combined into a string in rows to form the back part of the chromosome, so that the front part and the back part together form a chromosome. The sum of each row vector of the mixing coefficient C matrix is 1.0. After the genetic operation, parameters A and C must be normalized.

4.4.2. Fitness Function. The higher the likelihood of the training data to the model, the better. Here, the fitness function of the individual chromosome is expressed by the log-likelihood probability of the observed sequence of feature parameters of the English speech signal; namely,

$$f(\lambda) = \ln[P(O(k)|\lambda - 1)]. \quad (8)$$

In the above formula, $O(k)$ represents the k-th observation sequence used to train the model.

4.4.3. Population Initialization. The population initialization produces 100 chromosomes; that is, the population size is 100. Based on the fitness function, the fitness value is compared according to the fitness function size of each chromosome. 40 excellent chromosomes were selected from the population directly as part of the next-generation chromosomes. In addition, another part of 60 chromosomes is generated through crossover and mutation operations, which together form a new generation of chromosomes.

After real-number encoding of the chromosomes, the length of the chromosome is $L = 15055$. Through population initialization, 100 chromosomes are generated. According to the fitness value, 40 excellent chromosomes are selected directly as the next-generation chromosomes. The chromosomes of the next-generation population are better than the previous-generation chromosomes, so, after repeated iterations until $G = 60$, the generation is terminated, and the corresponding model is the CHMM model. The training process of GA-CHMM is shown in Figure 4.

5. System Test and Result Analysis

5.1. System Recognition Rate Test

5.1.1. Test and Analysis of the System's Different Vocabulary Recognition Rate. Since the system can set the size of the selected isolated vocabulary, different vocabularies of different sizes are selected for the recognition rate of the system during the test. The size of the vocabulary selected for the test is 5~200. When the system tests the recognition rate on the embedded system, it uses the PC-assisted software to complete the collection of English speech samples, the training of the template, and the update of the template data.

Before the test, first collect the English speech samples of isolated words. When collecting samples, 10 people are selected, and 4 English speech samples are collected for each isolated word; in this way, there are a total of 40 samples for each isolated word to participate in the training. After the English speech samples of all isolated words are collected, the English speech template is trained according to the set vocabulary size and then downloaded to the embedded system through the USB cable to test the recognition rate of the system. In each test, the tester spoke to each isolated vocabulary 10 times and recorded the system's recognition of the isolated vocabulary. We divide the number of successful recognitions by the total number of speeches to get the recognition rate of the system test. The experimental test results of the system's recognition rate of different vocabularies are shown in Figure 5.

The average recognition rate in Figure 5 is the average of the recognition rates of three experimental tests for each vocabulary. It can be seen from the results in Figure 5 that the recognition rate of the system changes tortuously as the vocabulary increases. When the vocabulary reaches 200, the recognition rate of GA_CHMM algorithm is about 88%.

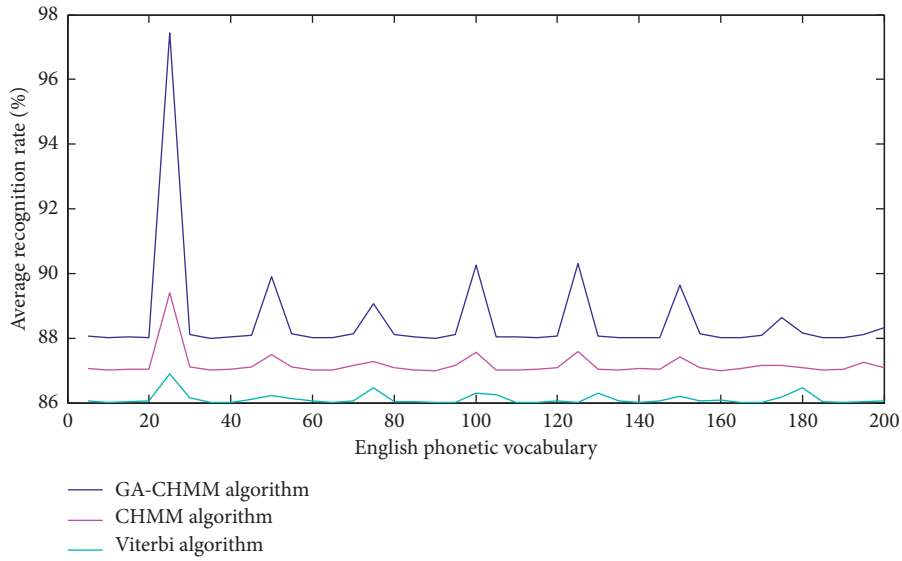

Figure 5: Recognition rate test results of different vocabularies.

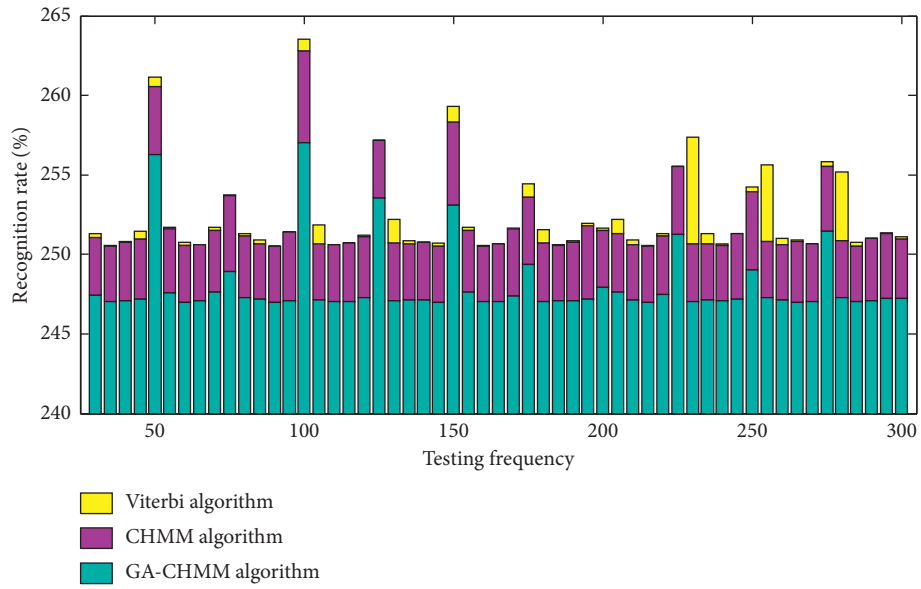

Figure 6: The test result of the recognition rate of the system nonspecific person.

5.1.2. Test and Analysis of the System's Recognition Rate of Unspecified Persons. The system selects 200 vocabularies, and 10 people are selected for the test. First select 8 people from 10 to train the English phonetic template, and each person samples 4 for each isolated vocabulary. In this way, there are 32 English phonetic samples for each isolated vocabulary. The other two people were used as the system unspecified person test, each isolated word was said 3 times, each person said the words 300 times in total, and the system's recognition rate was counted; the experimental results are shown in Figure 6.

This system provides the learning function of isolated words in English speech recognition. For vocabulary whose recognition rate is not high in the recognition process, the learning function is used to learn this vocabulary. After the learning is completed, we test the recognition rate of the system. The experimental test results are shown in Figure 7.

The experimental results in Figure 7 show that, after the learning function, the recognition rate of the system has been greatly improved. After the learning of isolated words, the recognition rate has reached about 90%. The system can achieve the learning function of isolated words through the learning function.

5.2. Real-Time Testing of the System. The vocabulary size of isolated words used in the experiment is 100. The time taken by the system from the completion of the collection of English speech to the recognition of the result is recorded as the system's response time.

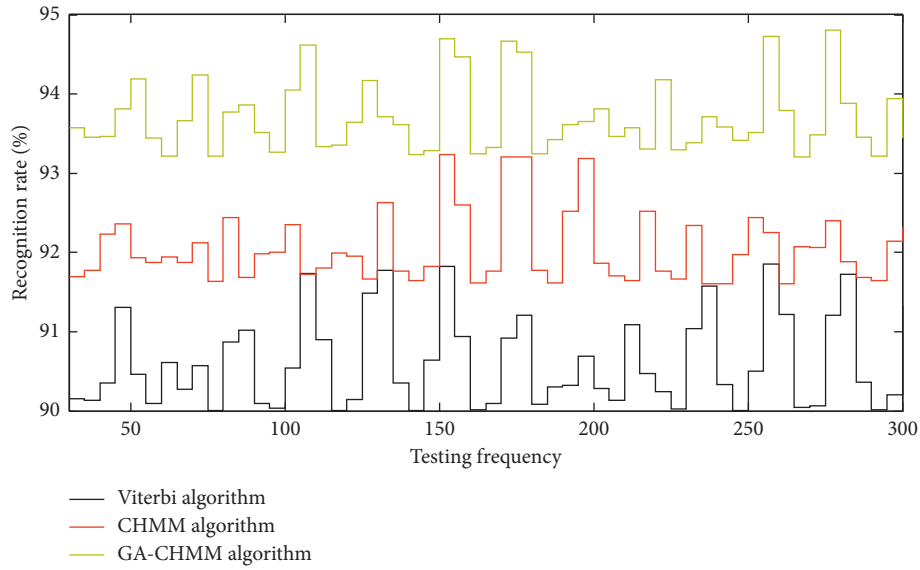

Figure 7: Recognition rate test results after using the learning function.

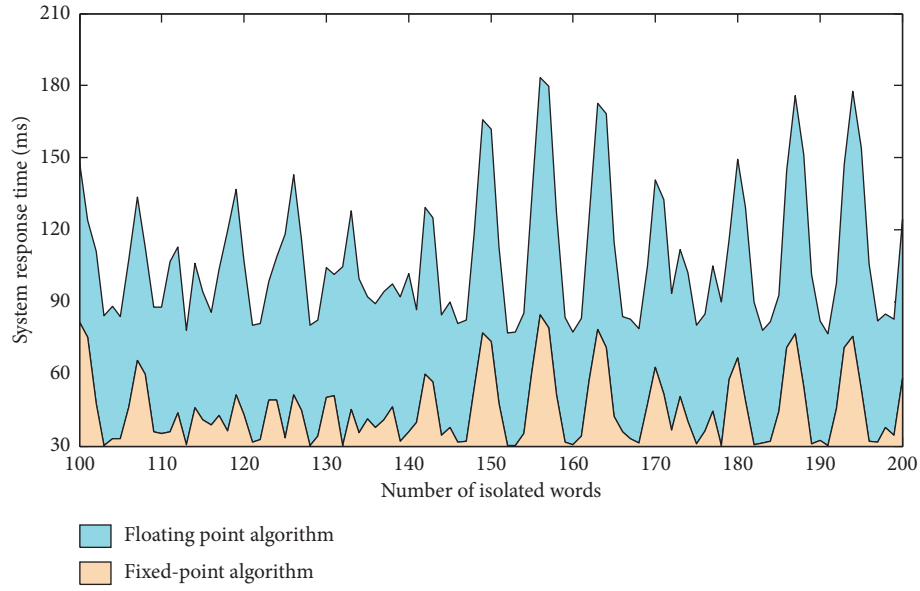

Figure 8: The system's response time using floating-point arithmetic and fixed-point arithmetic.

5.2.1. The Response Time of the Test System in the Case of Fixed-point and Floating-Point Arithmetic. We select 100 to 200 isolated words as test objects and test the system's response time when each word is recognized on the embedded system software, using floating-point arithmetic and fixed-point arithmetic, respectively. The test result is shown in Figure 8.

It can be seen from the results shown in Figure 8 that the system has a maximum response time of 180 ms when using floating-point arithmetic. After adopting fixed-point algorithm, the real-time performance of the system has been greatly improved, and the maximum response time of the system is 85 milliseconds.

5.2.2. The Response Time of the Test System Using Different Recognition Algorithms. We set the vocabulary of the system to 100 and perform English speech recognition for each vocabulary and record the real-time response time of GA_CHMM algorithm, CHMM algorithm, and Viterbi algorithm, respectively. The test results are shown in Figure 9.

In the test results shown in Figure 9, after the system adopts the optimized CHMM algorithm, the average response time of the system is the lowest, while the average response time of the system using the unoptimized CHMM algorithm is higher; the response time of the Viterbi algorithm is the highest.

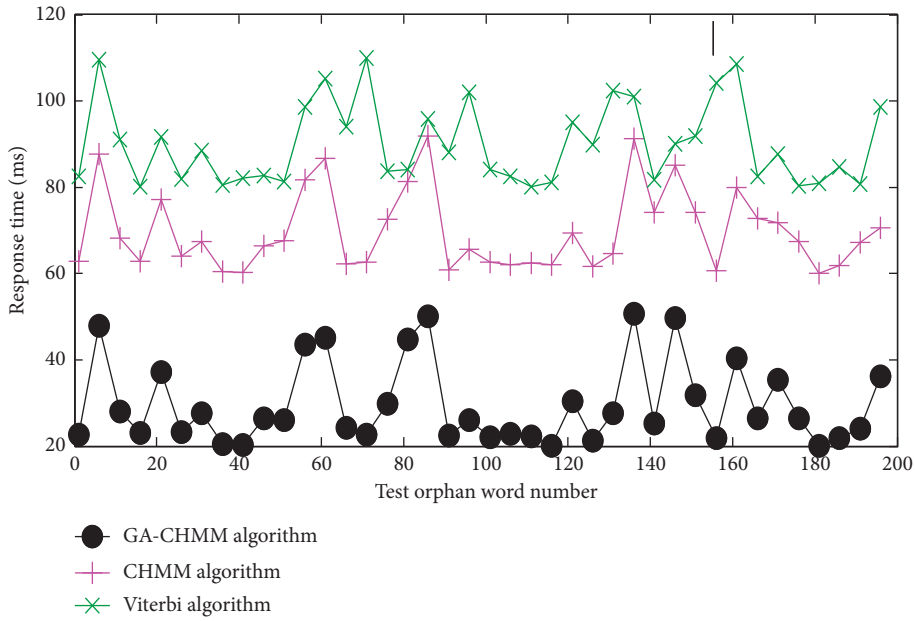

Figure 9: Response time of system algorithm.

6. Conclusion

This text has carried on the overall design of the English speech recognition system based on FPGA. We choose to use SOPC this way to realize the system and choose the software module through the comparison of algorithms. Equipped with the English speech recognition control system and the hardware part of the SOPC system, the software and hardware modules of the SOPC system are divided. The processor and memory of the hardware acceleration unit are configured in detail, and a memory implementation method is designed according to the needs of the system. The design of English speech recognition control system and hardware acceleration unit IP core is realized. This article analyzes the data and gives the scope of the data and draws a way of data representation. Combining modules from top to bottom describes the design of hardware IP. The genetic algorithm has the characteristics of superior global search capability and parallel computing. It improves the traditional CHMM algorithm and uses the genetic algorithm to directly train the CHMM model. This process mainly includes coding method, fitness function design, population initialization, selection operation, crossover operation, mutation operation, and termination strategy. The English speech recognition system based on the improved algorithm (GA_CHMM) is studied, which mainly includes the establishment of English speech template, preprocessing, endpoint detection, feature extraction, and training process. The English speech recognition system was simulated by MATLAB software, and a better recognition effect was obtained. This paper studies the real-time index of embedded English speech recognition system and the system optimization method of recognition rate. Through the analysis of the system hardware, technical methods such as fixed-point embedded system algorithm and storage space optimization are adopted to ensure the real-time requirements of the system. At the same time, this article focuses on the optimization of the CHMM algorithm in the recognition phase and proposes an optimized CHMM algorithm based on the irreversible characteristics of the English speech signal, and the real-time performance of the system is once again improved.

Data Availability

All the data are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this study.

References

[1] B. Kurniadhani, S. Hadiyoso, and S. Aulia, "FPGA-based implementation of speech recognition for robocar control using MFCC," *Telkomnika*, vol. 17, no. 4, pp. 1914–1922, 2019.

[2] A. A. Gilan, M. Emad, and B. Alizadeh, "FPGA-based implementation of a real-time object recognition system using convolutional neural network," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 4, pp. 755–759, 2019.

[3] T. Posewsky and D. Ziener, "Throughput optimizations for FPGA-based deep neural network inference," *Microprocessors and Microsystems*, vol. 60, pp. 151–161, 2018.

[4] M. A. A. de Sousa, R. Pires, and E. Del-Moral-Hernandez, "SOM processor: a high throughput FPGA-based architecture for implementing Self-Organizing Maps and its application to video processing," *Neural Networks*, vol. 125, pp. 349–362, 2020.

[5] M. Ling, M. Javad Esfahani, H. Akbari, and F. Amin, "Effects of residence time and heating rate on gasification of petroleum residue," *Petroleum Science and Technology*, vol. 34, no. 22, pp. 1837–1840, 2016.

[6] H. L. Ma and S. B. Tsai, "Design of research on performance of a new iridium coordination compound for the detection of

Hg2+," *International Journal of Environmental Research and Public Health*, vol. 14, no. 10, p. 1232, 2017.

[7] L. Y. Mo, W. H. Z. Sun, S. Jiang et al., "Removal of colloidal precipitation plugging with high-power ultrasound," *Ultrasonics Sonochemistry*, vol. 69, p. 105259, 2020.

[8] X. Yang, Q. Zhou, J. Wang et al., "FPGA-based approximate calculation system of General Vector Machine," *Microelectronics Journal*, vol. 86, pp. 87–96, 2019.

[9] C. Gu, N. Hanley, and M. O'neill, "Improved reliability of FPGA-based PUF identification generator design," *ACM Transactions on Reconfigurable Technology and Systems*, vol. 10, no. 3, pp. 1–23, 2017.

[10] L. Xia, L. Diao, Z. Jiang et al., "Pai-FCNN: Fpga based inference system for complex CNN models,"vol. 2160, pp. 107–114, in *Proceedings of the 2019 IEEE 30th International Conference on Application-Specific Systems, Architectures and Processors (ASAP)*, vol. 2160, pp. 107–114, IEEE, New York, NY, USA, July 2019.

[11] S. Afifi, H. GholamHosseini, and R. Sinha, "A system on chip for melanoma detection using FPGA-based SVM classifier," *Microprocessors and Microsystems*, vol. 65, pp. 57–68, 2019.

[12] M. Sahani and P. K. Dash, "FPGA-based online power quality disturbances monitoring using reduced-sample HHT and class-specific weighted RVFLN," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 8, pp. 4614–4623, 2019.

[13] D. Xu and H. Ma, "Degradation of rhodamine B in water by ultrasound-assisted TiO_2 photocatalysis," *Journal of Cleaner Production*, vol. 313, Article ID 127758, 2021.

[14] D. Gao, Y. Liu, Z. Guo et al., "A study on optimization of CBM water drainage by well-test deconvolution in the early development stage," *Water*, vol. 10, no. 7, 2018.

[15] H. Y. Kim, L. Xu, W. Shi et al., "A secure and flexible FPGA-based blockchain system for the IIoT," *Computer*, vol. 54, no. 2, pp. 50–59, 2021.

[16] K. K. Guner, T. O. Gulum, and B. Erkmen, "FPGA-based wigner–hough transform system for detection and parameter extraction of LPI radar LFMCW signals," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–15, 2021.

[17] J. W. Chang, K. W. Kang, and S. J. Kang, "An energy-efficient FPGA-based deconvolutional neural networks accelerator for single image super-resolution," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 1, pp. 281–295, 2018.

[18] O. Oballe-Peinado, J. A. Hidalgo-Lopez, J. Castellanos-Ramos et al., "FPGA-based tactile sensor suite electronics for real-time embedded processing," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 12, pp. 9657–9665, 2017.

[19] C. Xu, Z. Peng, X. Hu et al., "FPGA-based low-visibility enhancement accelerator for video sequence by adaptive histogram equalization with dynamic clip-threshold," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 11, pp. 3954–3964, 2020.

[20] T. Grubljesic, P. S. Coelho, and J. Jaklic, "The shift to socio-organizational drivers of business intelligence and analytics acceptance," *Journal of Organizational and End User Computing*, vol. 31, no. 2, pp. 37–64, 2019.

[21] L. X. Z. Zhang, M. Mouritsen, and J. R. Miller, "Role of perceived value in acceptance of bring your own device policy," *Journal of Organizational and End User Computing*, vol. 31, no. 2, pp. 65–82, 2019.

[22] A. Shahri, M. Hosseini, K. Phalp, J. Taylor, and R. Ali, "How to engineer gamification: the consensus, the best practice and the grey areas," *Journal of Organizational and End User Computing*, vol. 31, no. 1, pp. 39–60, 2019.