

Design and Implementation of Digital Image Processing Teaching Experimental System

Zengguo Sun*, Yangguang Zhi

School of Computer Science
Shaanxi Normal University
Xi'an, China

Luwei Zhang

School of Food Equipment
Engineering and Science
Xi'an Jiaotong University
Xi'an, China

Xiaodan Hu, Jinke Jiang

College of Computer Science and
Technology
Huaqiao University
Xiamen, China

Abstract—In order to satisfy the teaching requirements of the digital image processing course, the teaching experimental system for this course was developed. Requirement analysis, function design, and structure design were provided successively for the digital image processing teaching experimental system, and then, the system was coded and implemented by Delphi language. With the beautiful interface, convenient operation, and strong practicability, the system has functions of basic gray level transformation, histogram processing, spatial filtering, frequency domain filtering, noise reduction, edge detection, and morphological operation. Therefore, the system can be regarded as the assistant tool for digital image processing course teaching.

Keywords—requirement analysis; structure design; Delphi language; digital image processing

I. INTRODUCTION

Digital image processing is one of the core courses in computer science. It contains rich content such as image transformation, spatial filtering, frequency domain filtering, image denoising, edge detection and morphological operation [1-3]. The contents of these studies also include a lot of algorithms, whose performance is directly dependent on the related parameters which you select. If we only study the theory, it is difficult to understand the performance of various algorithms of image processing, and it is not easy to grasp the intrinsic relationship between different functions, which are a great challenge to teach the course of digital image processing. If we can have a teaching experimental system about digital image processing course, which can integrate various categories of digital image processing and different algorithms, also can independently adjust the parameters, observe and compare the results, it will be convenient for the teachers to explain the various types of image processing algorithms vividly, and also for students to understand the internal links of different content of image processing so as to establish complete knowledge system of the course [4-6]. Therefore, the development of the teaching experimental system of the digital image processing course is of great significance to the teaching of the course. In this paper, the requirement analysis, function design and structure design of digital image processing

teaching experimental system are conducted successively, and the system is coded by using Delphi language that is a well-known rapid application development tool. The digital image processing teaching experimental system covers all aspects of the course, such as basic gray level transformation, histogram processing, spatial filtering, frequency domain filtering, image denoising, edge detection and morphological operation. Each function includes a variety of algorithm. At the same time, the system interface is beautiful, interactive, and algorithm parameters can be adjusted flexibly. The system is not only easy for teachers to teach, but also facilitates students to learn. Therefore, the system can be used as a highly effective auxiliary tool for digital image processing course, which can effectively stimulate students' interest in learning and facilitate the development of heuristic teaching, which has a positive effect on improving the quality of teaching.

II. REQUIREMENT ANALYSIS

In combination with the teaching content of the course, the teaching experimental system of digital image processing needs to realize the functions of basic gray level transformation, histogram processing, spatial filtering, frequency domain filtering, image denoising, edge detection and morphological operation. Among them, basic gray level transformation includes image reversal, logarithmic transformation and power conversion. Histogram processing includes histogram equalization, and spatial filtering includes mean filtering and median filtering. Frequency domain filtering includes ideal low-pass / high-pass filter, Butterworth low-pass / high-pass filter, Gaussian low-pass / high-pass filter, ideal band-stop / band-pass filter, Butterworth band-stop / band-pass filter and Gaussian band-stop / band-pass filter. Image noise reduction includes adding and suppression of Gaussian noise, multiplicative noise, salt and pepper noise and periodic noise. Edge detection includes three methods: Sobel template, Laplacian template and Prewitt template. Morphological treatment includes expansion, corrosion, boundary extraction, hole filling and extraction of connected components. For the convenience of use, observation and comparison, the basic gray level transformation, histogram processing, spatial filtering and frequency domain filtering include three windows corresponding to pre-processing, post-processing and parameter panel respectively. The image noise

This work is supported by National Natural Science Foundation of China (Grant No. 61102163), Natural Science Foundation of Fujian Province (Grant No. 2012J01271), the fourth batch of Information Demonstration Course Construction Project of Shaanxi Normal University, "Teaching Model Innovation and Practice Research" Special Foundation of Shaanxi Normal University, and All English Teaching Demonstration Course Construction Project of Shaanxi Normal University in 2017.

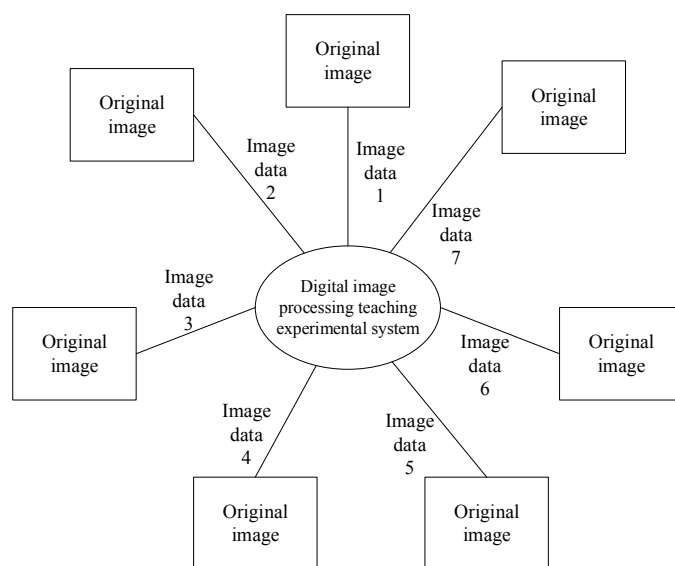
*Corresponding Author: Zengguo Sun, duffer2000@163.com.

reduction includes four windows of original image, noise-adding image, de-noising image and parameter panel. The edge detection contains four windows: the original image, the edge image, the binarized edge image and the parameter panel. The morphological operation consists of four windows: the original image, the binarized image, the processed image and the parameter panel. Each function module can realize the selection of the original image and the preservation and loading of the processed image, the loading operation of the processed image and the other original image as the new original image, so as to improve the flexibility and coherence of the user operation. It can be seen that the teaching experimental system of digital image processing contains rich content, which basically covers all aspects of course content and can meet the actual needs of course teaching.

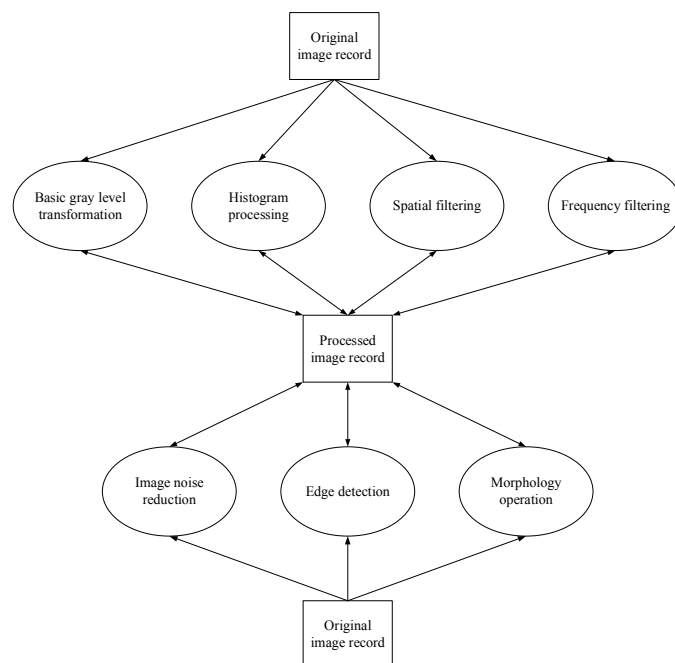
III. FUNCTION DESIGN

The goal of the functional design is to generate a functional model, which is described by the data flow diagram [7-9]. The data flow diagram is a description tool of structure analysis, which graphically depicts the flow and process of data in the system, reflecting the logical function that the system must complete. Digital image processing teaching experimental system consists of seven functional modules: basic gray level transformation, histogram processing, spatial filtering, frequency domain filtering, image denoising, edge detection and morphological operation. Each function module contains several sub-functions. Therefore, we should gradually break down this system according to the hierarchical structure of the problem, and use a set of hierarchical data flow diagram to reflect this structural relationship. The data flow diagram of the system shown in Fig. 1, includes the top layer, 0-layer graph and 1-layer graph (take the edge detection module as an example). In the top layer of the data flow diagram, the image data 1 to 7 denotes input image data corresponding to the basic gray level transformation module, the histogram processing module, the spatial filtering module, the frequency domain filtering module, the image noise reduction module, the edge detection module and the morphological operation module, respectively. In the 0-layer data flow diagram, the original image is a collection of original images, and the processed image is a collection of processed images. An image is selected from the original image records for a certain processing, and the processed result is stored in the processed image recording, or is loaded as a new original image for further processing. The seven image processing modules share the record of the image processing and the processed image can be immediately converted to the record of original image, which facilitates the continuous processing of the image and also reflects the intrinsic relationship between the different functional modules. For example, in order to improve the visual effect, noise reduction image outputted by noise reduction module needs to use the image transformation module to improve the contrast. The black-on-white blurred image must be subjected to the reverse processing of the basic gray level transformation module before morphological operation, and the edge image outputted by the edge detection module is often dealt with morphological refinement. In the 1-layer data flow graph, the edge detection algorithm is a set of edge detection algorithms for the edge detection module, including Sobel operator,

Prewitt operator and Laplacian operator. It can be seen that the drawing of the data flow diagram clarifies the functional model of the system and prepares well for the structure of the system.



(a) Top layer graph



(b) 0-layer graph

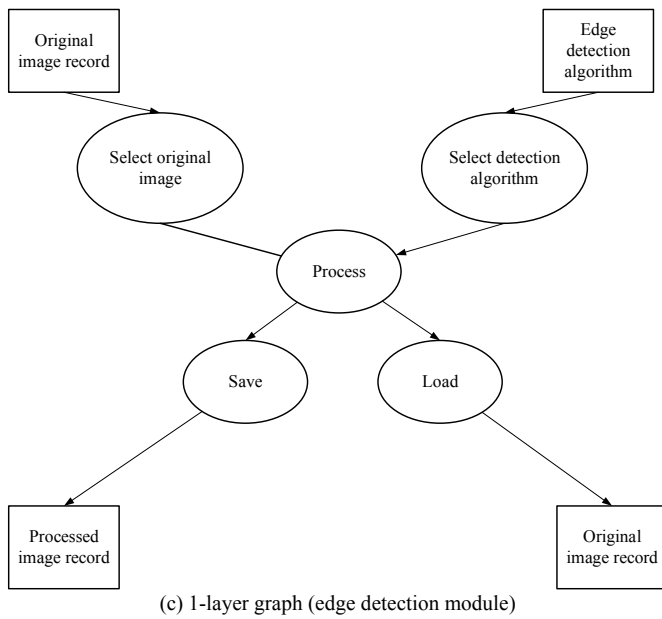


Fig. 1. Data flow diagram of digital image processing teaching experimental system.

IV. STRUCTURE DESIGN

According to the function design of the digital image processing teaching experimental system, this section further carries on the structure design, whose purpose is to determine the structure frame and the function module of system [7-9]. The digital image processing teaching experimental system includes seven function modules: basic gray level transformation, histogram processing, spatial filtering, frequency domain filtering, image denoising, edge detection, morphological operation, which is shown in Fig. 2. As described in the function design, each function module can perform different processing on the original image. For example, the basic gray level transformation includes three different sub-functions: image reversion, logarithmic transformation and power transformation. The original image is processed to obtain the processed image, and the processed image can be saved and loaded further.

The basic operation flow chart of the system is shown in Fig. 3. After entering the main interface, firstly select the function module to enter the specific image processing function interface. The system automatically determines whether the original image exists. If there is no original image, original image is required to load. Otherwise you cannot choose the sub-function algorithm. If the original image exists, you can further select the sub-function algorithm, and set the corresponding parameters of the algorithm to deal with the original image, and get the corresponding image results. If you need to continue processing, then load the resulted image. Then the resulted image is turned into the original image, and it is used for the follow-up continuous treatment. If you do not need, save the image and the entire process ends.

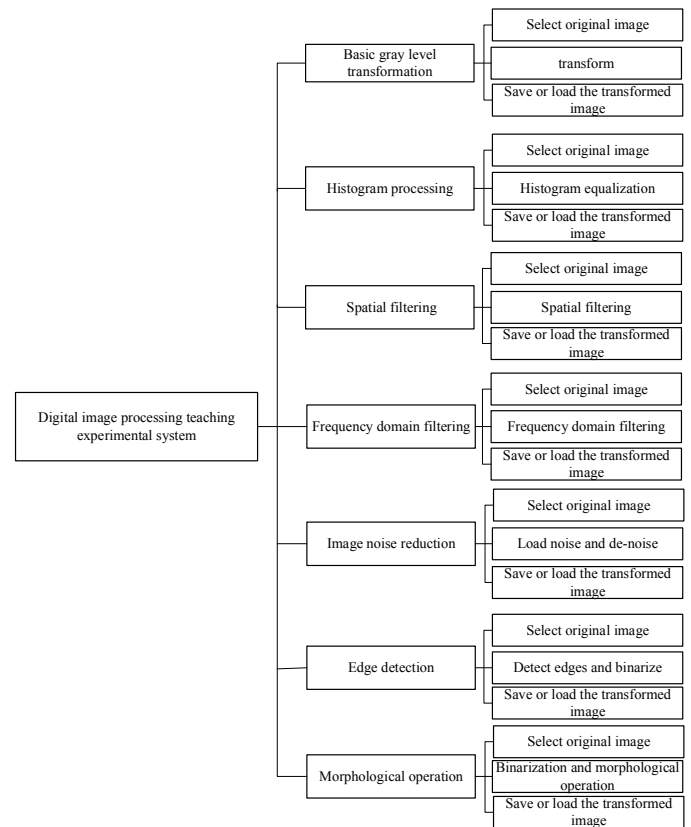


Fig. 2. Structure of teaching system of digital image processing.

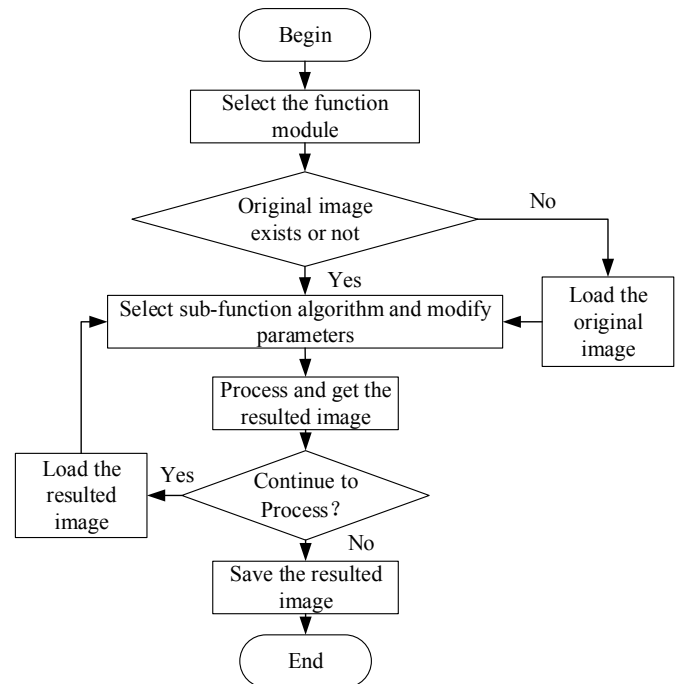


Fig. 3. Flow chart of basic operation.

V. CODE DESIGN

According to the results of structure design, the code of digital image processing teaching experimental system can be

designed [7-9]. The system interface and image processing algorithms are written by Delphi. Delphi is a well-known fast application development tool on the Windows platform that uses many advanced features and design ideas of the Microsoft Windows graphical user interface, and it is a flexible, reusable, complete object-oriented programming language and the world's fastest compiler and the most advanced database technology [10-12]. So the use of Delphi to develop the platform can significantly improve the programming efficiency. The part of main interface uses Metro style, which is clear and beautiful. According to different classification of image processing, different function module windows are entered. The realization of each sub-function of each function module window is through the independent label of parameters panel to separate. Among them, the function module window is composed of several independent windows. Images before and after processing and the parameter setting panel occupy different windows respectively, which can not only compare the whole operation, but also observe results independently. In addition, images before and after processing can be synchronized to enlarge, reduce, drag and drop. And the processed image can also be copied to the original image window for further processing. In this section, the expansion of morphological operation module is taken as an example to illustrate how to use Delphi to realize the coding. The code is listed as follows. It should be noted that there appear some Chinese characters in the code because this system is developed just for Chinese students.

```
if PageControl.ActivePage = TabPxtxcl then // morphological
operation
begin
if PageXtxcl.ActivePage = TabPz then // dilate
begin
if IntTempl <> nil then
begin
SetLength(MyIntMatrix2, Height, Width);
for x := 0 to Height - 1 do
for y := 0 to Width - 1 do
begin
MyIntMatrix2[x, y] := PicMatrix[x, y];
end;
MatrixUnit.BW(MyIntMatrix2, Pxyz.value); // 0-255
MatrixUnit.SaveToImage(MyIntMatrix2,
Image2.Image2Form.Image2);
MatrixUnit.Dilate(MyIntMatrix2, IntTempl, MyIntMatrix);
IntTempl := nil;
MyIntMatrix2 := nil;
MatrixUnit.SaveToImage(MyIntMatrix,
Image3.Image3Form.Image3);
end
else
MessageBox(Self.Handle, '请将参数填写完整!', '注意',
MB_OK);
End
```

VI. INTERFACE DISPLAY

This section presents the main interface of the digital image processing teaching experimental system, and takes the three modules of frequency domain filtering, image denoising and edge detection as an example to elaborate the operation flow of the module, and finally give an example of continuous processing of the system. Then it explains the use and functional characteristics of the system. It should be noted that Chinese characters appear in such interfaces because this system is designed for Chinese students.

A. Main Interface

The main interface of the digital image processing teaching experimental system is shown in Fig. 4. The left is a label of function module, and the right side shows the corresponding introduction of module label and sub-function that the module contains. For example, if you select the "spatial filtering" module, the right side shows the introduction of spatial filtering and the sub-functions (mean filter and median filter) of this module. By clicking the arrow in the lower right corner to enter the selected function module, and start the image processing. It can be seen that the system interface is beautiful, easy to operate, full-featured, and able to meet the actual needs of curriculum teaching.



Fig. 4. The main interface of digital image processing teaching experimental system (Appearance of Chinese characters is because this system is just used for Chinese students. The following interfaces are in the same situation).

B. Frequency Domain Filtering Module

The interface of frequency domain filtering module is shown in Fig. 5. Here, we only take the Gaussian high-pass filter as an example. In operation, the original image is loaded firstly. Then, the "Gaussian high-pass" filter in the parameter panel is selected and the corresponding parameters are inputted. Lastly, click the "Process Image" button to get the result image. Comparing the original image with the resulted image, you can see the characteristics of Gaussian high-pass filter, i.e., suppressing the background and other low-frequency components, and retaining the edge and other high-frequency components.

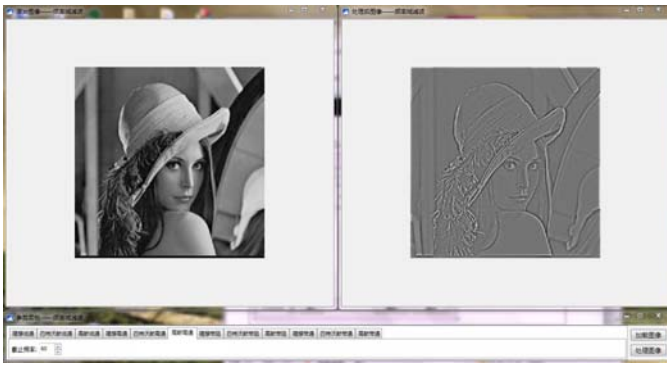


Fig. 5. Frequency domain filtering module interface (Gaussian high-pass filter).

C. Image Noise Reduction Module

Interface of image noise reduction module is shown in Fig. 6, where only the salt and pepper noise is taken as an example. In this system, the median filter is used to suppress the salt and pepper noise. Firstly, the original image needs to be loaded. Then, select the “salt and pepper noise” function in the parameter panel, and input the noise density and the width and height of the filter template. Lastly, click the “process image button” to get the noise image in the middle and the denoising image on the right. It can be seen that the median filter can effectively suppress the granular salt and pepper noise.

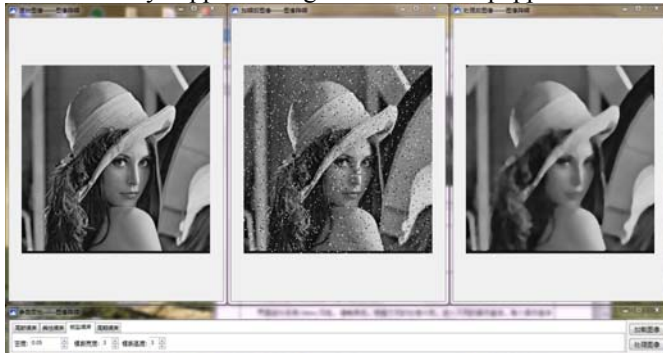


Fig. 6. Module interface of image noise reduction (salt and pepper noise).

D. Edge Detection Module

The edge detection module interface is shown in Fig. 7, where the use of Sobel template is taken as an example. Firstly, load the original image that needs to be processed. Then, select the “Sobel template” function in the parameter panel, and input the width, height and threshold of the template (for subsequent binarization). Click “Input Template 1” and then “Input Template 2”, and fill in the coefficients of the Sobel template (shown in Fig. 8). Lastly, click the “Process Image” button to get the not binarized edge image in the middle and the binarized edge image on the right. It can be seen that the Sobel template can effectively detect the edge of the image, and the final edge thickness and continuity are directly dependent on the selection of the binarization threshold.

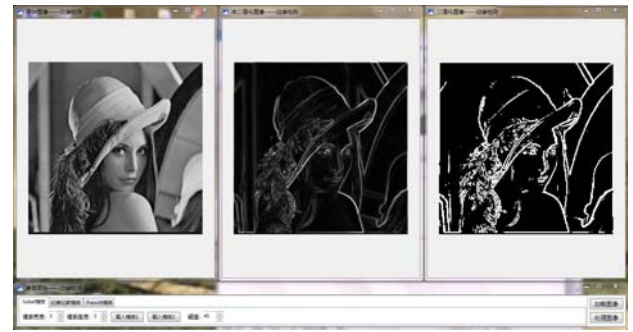
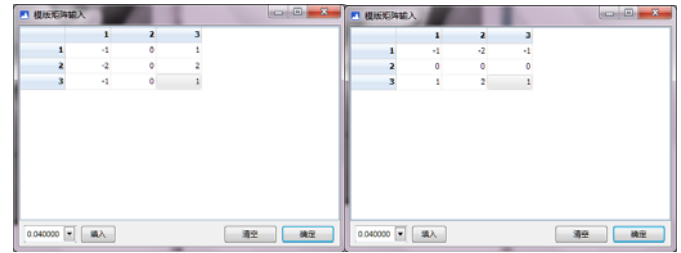


Fig. 7. Edge detection module interface (using Sobel template).



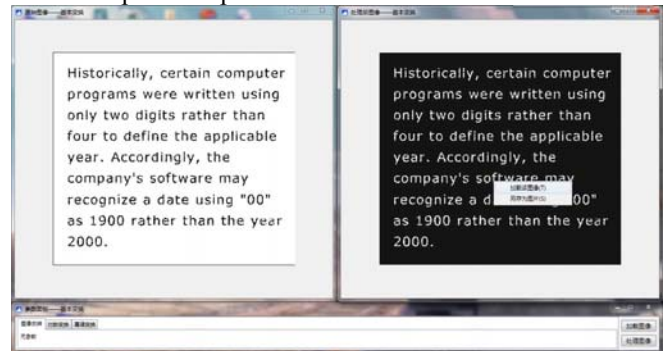
(a) Vertical direction

(b) Horizontal direction

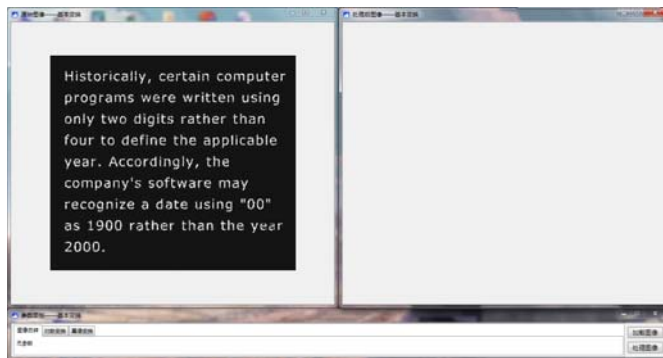
Fig. 8. Sobel template coefficients.

E. Continuous Processing Examples

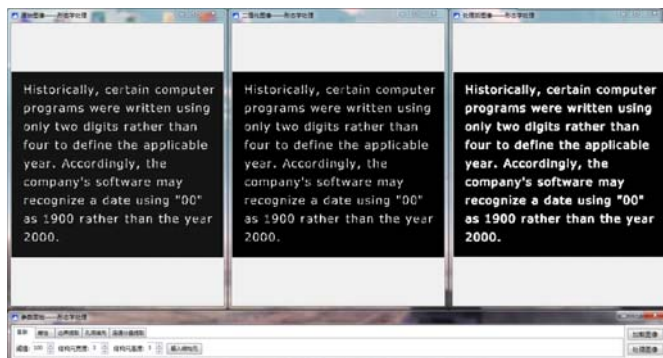
This section takes the text enhancement on the white background as an example to specify the use of continuous operation of the system. As shown in Fig. 9, for the text on a white background, the resolution is not high, and the text appears the phenomenon of breaking, and is not easy for the follow-up operation of classification and recognition. Firstly, the image is subjected to an reversion operation of the basic gray level transformation so that the background of the image becomes black and the text content becomes white; Secondly, right-click on the processed black background image and select “load the image”, loading the image as a new original image for subsequent processing; Lastly, the new loading image is subjected to the dilation in the morphological operation, and the enhanced high-resolution result image is obtained. The fracture problem of the text is solved well, which helps the subsequent machine processing. It can be seen that the system allows continuous processing of different functional modules, which can not only establish the overall link between the modules, but also provide a more flexible solution of practical problems.



(a) reverse operation



(b) Load the new original image



(c) Dilation operation

Fig. 9. Continuous processing example (text enhancement on white background).

VII. CONCLUSION

In this paper, the requirement analysis, function design and structure design of the digital image processing teaching experimental system are carried out, and the Delphi language is used to complete the code design of the system. The requirement analysis analyzes the function of the system from the user's point of view. The function design clarifies the logical function that the system needs to complete in the form of the data flow diagram. The structure design determines the structure frame and function module of the system from the macroscopic aspects. The code design uses the efficient language of Delphi to complete the coding of the system. The digital image processing teaching experimental system includes basic function modules such as basic gray level transformation, histogram processing, spatial filtering, frequency domain filtering, image denoising, edge detection

and morphological operation. The interface is beautiful, interactive, and easy to use. The parameters can be adjusted independently and easy to observe the processing results, which can not only demonstrate the function of a particular module, but also facilitate the continuous processing between different modules to meet the practical application of image processing needs. Therefore, the system can be used as a highly effective auxiliary tool for digital image processing course, effectively stimulating students' interest in learning and facilitating the development of heuristic teaching. This system also has a positive effect on improving the quality of teaching. In addition, as a multimedia teaching resource, the system is helpful for the development of information teaching and the cultivation of innovative talents.

REFERENCES

- [1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd Edition. New York: Prentice Hall, 2008.
- [2] Hongyan Quan and Guitao Cao, *Principles and Implementations of Digital Image Processing*. Beijing: Mechanical Industry Press, 2014.
- [3] Rongchun Zhao, Zhongming Zhao, and Xinbo Zhao, *Digital Image Processing and Analysis*. Beijing: Tsinghua University Press, 2013.
- [4] Huanxin Zou and Shilin Zhou, "The construction of experimental teaching system on digital image processing course," *Journal of EEE*, vol. 34, no. 6, pp. 48-51, December 2012.
- [5] Zhanpeng Huang, Shizhong Jiang, and Jinyu Liu, "Realization of digital image processing virtual laboratory based on Web," *Experimental Technology and Management*, vol. 27, no. 12, pp. 110-112, December 2010.
- [6] Jian Li, Qinghua Yu, and Ying Zheng, "Development of digital image processing web experimental platform based matlab," *Experimental Technology and Management*, vol. 23, no. 10, pp. 80-84, October 2006.
- [7] Lang Li, Yali Zhu, and Jiang Xiong, *Software Engineering*. Wuhan: Huazhong University of Science and Technology Press, 2013.
- [8] I. Sommerville, *Software Engineering*, Ninth Edition. Beijing: Mechanical Industry Press, 2011.
- [9] S. L. Pfleeger and J. M. Atlee, *Software Engineering*, 4th edition. Weidong Yang, translated. Beijing: People's Posts and Telecommunications Press, 2010.
- [10] Yunguo Hong, *Delphi Programming*. Beijing: Beijing University of Science and Technology Press, 2013.
- [11] Shui Liang, *Delphi Program Development Example Book*, 2nd edition. Beijing: People's Posts and Telecommunications Press, 2010.
- [12] Aqi Zheng, *Delphi Practical Tutorial*, 2nd edition. Beijing: Electronic Industry Press, 2008.