Image edge detection based on quantum genetic algorithm

Lin Tang*

School of Information Engineering, Chongqing Vocational Institute of Engineering, Chongqing, 402260, China

Received 1 June 2014, www.cmnt.lv

Abstract

In asymmetric fingerprinting, the merchant can trace the traitors from a pirated copy by means of the embedded unique fingerprint, while the customer is immune of being framed due to the asymmetric property. In this letter, we propose an asymmetric fingerprinting scheme based on 1-out-of-n oblivious transfer, which is efficient from the bandwidth usage point of view. First, multicast that is an efficient transport technology for one-to-many communication is exploited, which can reduce the bandwidth usage significantly. Second, symmetric encryption instead of public-key encryption is performed on the multimedia content, which can reduce the complexity and communication cost.

Keywords: asymmetric fingerprinting, oblivious transfer, multicast communication

1 Introduction

Image edge detection is widely used in contour feature extraction and texture analysis, and plays an important role in the recognition of computer vision and pattern. However, the classical edge detection operators are mostly based on the numerical derivative near the pixel of the original image. Though such operators are very simple and convenient, they are only applicable to limited types of edge detection, and are very sensitive to noise, which makes it easy to cause edge breakage. After decades of research, many methods have been proposed, but due to the complexity and diversity of the image content as well as digital image imaging mechanism and other reasons, the detection accuracy is not high, the adaptability for different types of images is weak, and the image detection results in some regions is not ideal [1].

As a new function optimization algorithm, genetic algorithm has been widely applied due to its characteristics of efficiency, stability and parallel computing. With its features like strong robustness, positive feedback and distributed processing, etc. genetic algorithm has been successfully applied in combinatorial optimization. Its robustness feature enables basic genetic algorithm to solve other problems more easily after certain modification. There is no doubt that the edge detection based on quantum genetic algorithm will be applicable and effective.

This paper described the features of image edge detection, analyzed the basic process of quantum genetic algorithm, and applied it to image edge detection. It found the real edge corresponding to the target actual boundary line in the image through cost function minimization based on quantum genetic algorithm. This approach has a wide edge definition range, and can be used to detect many different types of edges. Its cost function uses not only image data but also edge continuity, edge thickness and region dissimilarity to measure the local edge structure information, therefore it is expected to generate images with continuous thin edge and good location.

2 Overview of image edge detection

Image edge detection is one of the basic features of image, which contains rich intrinsic information of image, like direction, step nature and shape. It is widely used in image segmentation, image classification, registration and pattern recognition.

In late 1970s, Marr and Hildretch from Massachusetts Institute of Technology proposed an edge detection theory similar with human visual system and kicked off the study of edge detection algorithm. Afterward, the detection and extraction methods of image edge features have always been the research focus of image processing and analysis [2].

2.1 PROBLEM DESCRIPTION

Image edge is always caused by the change of the physical properties of the scene in image, such as the brightness of photography, geometric characteristics (direction and depth etc.) and reflection coefficient of image, which exists between object and background, between object and object, and between region and region. There is direct connection between image edge and the physical properties of image content, so image edge detection contains the most of information of image [3].

Edge is the set of pixels due to the sudden changes of image grayscale in space or gradient direction. Image edge can be divided into step edge and roof edge, the step edge of which has significant change for grayscale values on both sides; while the roof edge is located in the junction of grayscale increase or reduction. Mathematically, the two

*** Corresponding author's** e-mail:lintang663@126.com

kinds of edges can be described with the first and second derivatives of grayscale change curve.

The following figure shows the changes of two edges with continuous functions, in which $f_s(x)$ and its derivative represent step edge and $f_r(x)$ and its derivative represent roof edge [4].

It can be found that, for the step edge, the first derivative of grayscale change curve has extreme value on edge and the second derivative has zero crossing point on edge; for the roof edge, the first derivative of grayscale change curve has zero crossing point on edge and the second derivative has extreme value on edge.

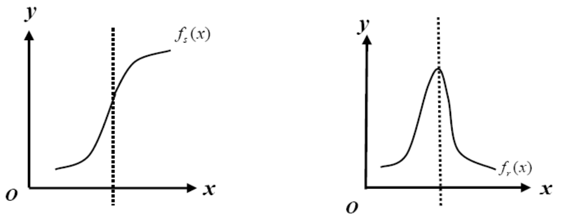


FIGURE 1 Image edge grayscale change curve

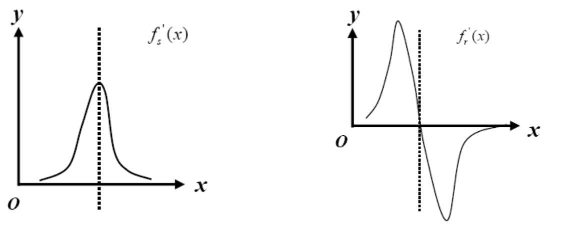


FIGURE 2 The first derivative change curve of image edge

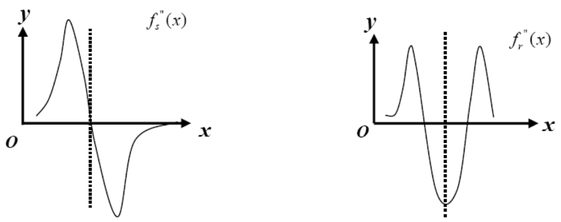


FIGURE 3 The second derivative change curve of image edge

2.2 ANALYSIS OF IMAGE EDGE CHARACTERISTIC

Human visual study shows that “image perception mainly depends on the correlation between objective stimuli”. Image perception is not the sum of all the parts of pattern but the organic composition of each part. Human pattern recognition has the following features:

1) The similar parts of pattern in space can form a whole easily.

2) Significant image is comparatively steady.

3) In an ordered pattern, if new parts are added, these new parts can be easily considered as the continuation of original pattern.

These visual features of human pattern recognition are reflected in the evaluation of image edge detection result to show that there are higher evaluation for continuous edge detection result and lower evaluation for broken edge detection result; there is higher evaluation for significant edge detection result and lower evaluation for the insignificant random noise detection result due to failure [5, 6].

In traditional edge definition, it uses step function and ramp function to represent the boundary changes of the two background regions as shown in the left picture of Figure 4. The edge features in reality are shown in the right picture of Figure 4, including (a) smooth gradual transition; (b) edge with unintensity; (c) edge with intensity; (d) edge in noise environment; (e)the combination of the above; (f) if these factors are far away enough from ideal conditions, it is hard to recognize edge data from noise.

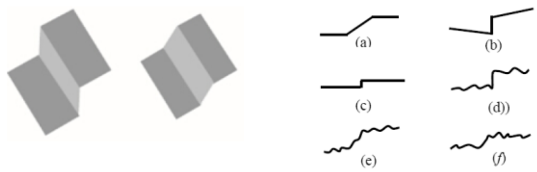


FIGURE 4 Traditional edge definition and edge features

Real image often contains a lot of incomplete or fuzzy data, especially image edge. Edge information is usually hidden (incompleteness) or twisted (fuzziness) partially due to various factors. Human visual system has a lot of experience of recognizing different objects under different conditions, meanwhile the organic combination of eyes and brain has already been familiar with recognizing the profiles of various objects under various conditions. People are trying to find out a heuristic algorithm for image understanding.

3 Genetic algorithm

3.1 OVERVIEW AND CHARACTERISTICS OF GENETIC ALGORITHM

Based on the law of biosphere (survival of the fittest), genetic algorithm was proposed in 1975. Genetic algorithm operates on structured object directly without the limitation of derivation or continuity of function; so genetic algorithm has intrinsic invisibility and parallelism and its global search capability is comparatively good. Genetic algorithm usually uses probability method for the optimal solution, so the algorithm can obtain search space automatically and give optimization guidance, correspond to and adjust search direction automatically without definite rule. The excellent properties of this genetic algorithm have been widely used in many modern fields [7].

Genetic algorithm has the characteristic of multi-path search. The most primitive methods have: linear programming, mixed data programming, nonlinear programming and dynamic programming etc.. These methods are simple and single search algorithms, with which the required solution shall transfer to another solution from some solution of search space. These common methods usually can obtain local optimal solution but not global optimal solution. Different from these common algorithms, genetic algorithm can obtain many solutions in a fixed time range and also can evaluate these results, which shows that the genetic algorithm is an intelligent algorithm that can be used in a fixed time range and find many peaks. Just because this characteristic of genetic algorithm, it is much possible for genetic algorithm to find global optimal solution.

Genetic algorithm has the characteristics of invisibility and parallelism. Maybe genetic algorithm only makes mathematical calculation with n individuals, but in fact, about n^3 patterns are processed invisibly. It means that each generation only needs to execute the calculation with a certain proportion to group size so as to reach the needed requirement and perform mathematical calculation on about n^3 patterns. Genetic algorithm can meet modern requirements more than basic algorithms because such intrinsic hidden characteristics[8].

3.1.1 Basic genetic algorithm steps

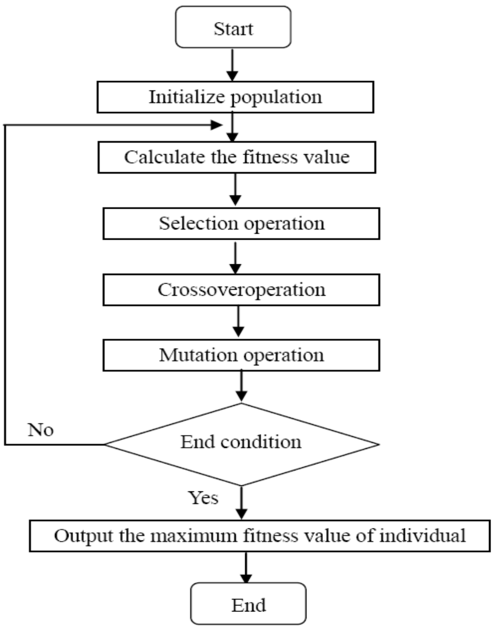


FIGURE 5 Genetic algorithm steps

3.1.2 Basic genetic algorithm operation

The selection of parameters of the algorithm.

1) Bit String Length L, Bit string length L will decide the accuracy of result. The higher the accuracy of result is, the longer the bit string is. Therefore, our calculation time will be longer and more data will be used. In order to shorten calculation time, we can select more excellent L value and also can recode in small range of available area, which is a feasible policy showing its excellent property.

2) Population Number N, when population number is big, we need to set population capacity and strengthen population quality with GA search method. In order to avoid the premature of population, we add the evaluation value of population adaptation for large number of population. Usually, we select n between 20-200.

3) Crossover Probability P_c, Crossover probability plays the role of control in the whole calculation process, which is used in the mathematical crossover calculation of (P_c,n) individual of new population. The higher crossover probability is, the more new individuals there are in a population. However, it is more possible to lose good genes. If the crossover probability is too low, the calculation will terminate in advance. Generally P_c =0.6-1.0.

4) Mutation Probability P_m, Mutation operation is the prerequisite of maintaining the diversity of population. After calculation, each gene on the bit string of each individual of copulation region changes randomly according to the mutation rate P_m, so secondary mutation (P_m, n, L) may occur. If P_m is too small, many excellent genes will be lost. If P_m is extremely big, meaningless random operation will be made. So, P_m= 0.005-0.01.

3.2 QUANTUM GENETIC ALGORITHM

3.2.1 Overview

Quantum genetic algorithm combines quantum computing and genetic algorithm. Quantum computing uses quantum state as basic message unit and can solve the NP problems in classical computing with the properties of the superposition, entanglement, interference and other properties of quantum state. Quantum genetic algorithm is a kind of genetic algorithm based on quantum computing theory. It introduces the state vector of quantum into genetic encoding, realizes chromosomes evolution with quantum logic gate and obtains better effect than common genetic algorithm. Based on the state vector of quantum, quantum genetic algorithm applies the probability amplitude expression of quantum bit to the encoding of chromosomes to make a single chromosome express the overlay of many states and realize update of chromosomes with quantum logic gate so as to achieve the target of optimal solution [9].

3.2.2 Quantum bit encoding.

In genetic algorithm, chromosomes are expressed with certain values (like binary number and floating point number). However, in quantum genetic algorithm (QGA), chromosomes are expressed by quantum bit (*qubit*) or random probability. The two basic states of quantum bits are different from classical bits as their quantum bit can be located in the random linear superposition of “0” and “1” besides the two states. That is to say, state “0” and state “1” exist at a certain probability at the same time. If state “0” or binary coding similar with genetic algorithm exists after measurement, quantum bit encoding can be made for multi-state genes. If a quantum bit represents two-state gene, two quantum bits represent four-state gene, k

quantum bits represent 2^k-state gene. The probability amplitude of *k* quantum bits can be expressed by:

$$q_j^t = \begin{bmatrix} \alpha_1 \left| \alpha_2 \right| \cdots \left| \alpha_k \right| \\ \beta_1 \left| \beta_2 \right| \cdots \left| \beta_k \right| \end{bmatrix}. \quad (1)$$

So, the probability amplitude of the chromosomes with m genes can be expressed by:

$$q_j^t = \begin{bmatrix} \alpha_{11} \left| \alpha_{12} \right| \cdots \left| \alpha_{1k} \right| \alpha_{21} \left| \alpha_{22} \right| \cdots \left| \alpha_{2k} \right| \cdots \left| \alpha_{m1} \right| \alpha_{m2} \left| \cdots \right| \alpha_{mk} \\ \beta_{11} \left| \beta_{12} \right| \cdots \left| \beta_{1k} \right| \beta_{21} \left| \beta_{22} \right| \cdots \left| \beta_{2k} \right| \cdots \left| \beta_{m1} \right| \beta_{m2} \left| \cdots \right| \beta_{mk} \end{bmatrix}, \quad (2)$$

where, q_j^t represents the chromosomes of the *j*-th individual of the *t*th generation, m is the number of genes of chromosomes, *k* is the number of quantum bits of gene encoding.

Quantum genetic algorithm uses quantum bits to store and express a gene and the gene will express all possible information besides a certain information. Any operation on the gene will work on all possible sates, which makes the quantum genetic algorithm have better variety feature than traditional genetic algorithm; for the same optimization problem, the population size of quantum genetic algorithm is much smaller than traditional genetic algorithm. Besides, quantum genetic algorithm can be used to obtain better convergence. With $|\alpha|^2$ or $|\beta|^2$ approaches 0 or l, the chromosomes of quantum genetic encoding will converge to a single state [10].

3.2.3 The process of quantum genetic algorithm

Based on the representation method of quantum bit and the quantum superposition, the process of quantum genetic algorithm is the following [11,12]:

Step 1: Initialize population.

The population with *n* individuals $P(t) = \{p_1^t, p_2^t, \cdots, p_n^t\}$ shall be initialized, in which $P_j^t \ (j = 1, 2, \cdots, n)$ represents the *j*-th individual of the *t*-th generation in the population, as shown in Equation (2).

All the genes $(\alpha_{ij}, \beta_{ij})$ of all the chromosomes of the population are initialized to $\left(1/\sqrt{2}, 1/\sqrt{2}\right)$, which means that all possible states expressed by a single chromosome superimposes at equal probability.

Step 2: Binary strings coding.

The individuals of initialized population shall be measured to obtain a set of solutions $R(t) = \{\gamma_1^t, \gamma_2^t, \cdots \gamma_n^t\}$, where $\gamma_j^t \ (j = 1, 2, \cdots, n)$ is the measurement value of the *j*-th individual of the *t*th generation in the form of binary string with length m and each bit is 0 or 1 obtained by the probability $|\alpha_{ij}|^2$ and $|\beta_{ij}|^2$ of quantum bit. The measurement process is: to get a random number between 0 and 1, if it is larger than the squared probability amplitude, the measurement value is 1 otherwise it is 0.

Step 3: Evaluate each individual of *R(t).*

Each individual of *R(t)* shall be evaluated with fitness evaluation function and the optimal individual of this generation shall be maintained. If a satisfactory solution is obtained, the calculation can end, otherwise, step IV shall be done.

Step 4: *P(t)* shall be updated with proper quantum gate *U(t)*.

Step 5: Genetic algebra $t=t+1$ shall be performed and Step 2 shall be taken.

4 Application of quantum genetic algorithm in edge detection

4.1 IMAGE PREPROCESSING

Given an original gray-scale image $G=\{g(i,j); 1\le i\le M, 1\le j\le N\}$, the first step of edge detection with optimized algorithm is to preprocess the image, which includes image dissimilarity enhancement and initial cost function evaluation.

4.2 DISSIMILARITY ENHANCEMENT

The basic characteristic of the edge pixel is separating different region. In dissimilarity enhancement, such characteristic of the pixel is enhanced, that is, to give pixels with such characteristic a large value. In the image $D=\{d(i,j); 1\le i\le M, 1\le j\le N\}$ obtained through enhancement dissimilarity, values of each pixel is proportional to the region dissimilarity of the pixel position. The value range of pixel is limit to $0\le d(i,j) \le 1$. Image dot with a value close to 1 can be considered as a good candidate edge dot. Obtain the enhanced image *D* through the following steps:

1) initialize all the pixel dots *d(i, j)* as zero.

2) execute the following steps on each pixel location:

a) take the pixel dot *(i, j)* as the center, conduct fitting to the centralized 12 edge structures, calculated the mean grayscale difference value $f(R1,R2)$ between region *R1* and *R2* in each fitting edge structure, find the edge structure with the maximum $f(R1,R2)$, take it as the best-fit edge structure, in which the three edge pixels are denoted as $(i,j)(i_1,j_1,)$ and (i_2,j_2).

b) alter the position of the best-fit edge structure, and conduct non-maximum suppression. Decide the displacement direction according to the actual edge structure. For horizontal, vertical and diagonal edge structure, move an edge position in all directions perpendicular to the edge, for the other edge structure, move an edge position in the four directions of up, down, left and right. For each displaced edge structure, decide the new *R1* and *R2*, and calculate the corresponding *f(R1,R2)*. If the *f(R1,R2)* obtained after the displacement is not bigger than the initial one, then increase the values of$\delta f(R1,R2)/3, d(i,j), d(i,j)$ and $d(i_1,j_2)$ by δ; if not, then the values of $d(i,j)$, $d(i_1,j_1)$ and $d(i_2,j_2)$ remain the same.

3) shorten the *d(i,j)* on each position to the maximum 1 to make sure the dissimilarity value range in image D is within [0,1].

4.3 COST FUNCTION EVALUATION

The cost function in each pixel position *(i,j)* is the weighted sum of the following five cost factors:

1) C_d: cost factor based on local region dissimilarity. This factor is to locate the edge pixel in the position with larger region dissimilarity, in the meanwhile, penalize the pixel position with smaller grayscale variance.

If pixel(i,j) is edge pixel, then $C_d(i,j)$=0, if not, $C_d(i,j)=d(i,j)$.

$$C_d(i,j)=\begin{cases}0, & (i,j)\, is\ edge\ pixel\\ D(i,j), & (i,j)\, is\ not\ edge\ pixel\end{cases}. \quad (3)$$

2) C: cost factor based on edge thickness. This factor is used to penalize pixel positions generating thick edges, in the meanwhile, reward the pixel position generating thin edges. If the pixel in(i,j) is thick edge, then $C_t(i,j)$=1, if not, $C_t(i,j)$=0.

$$C_t(i,j)=\begin{cases}0, thin\ edge\\ 1, others\end{cases}. \quad (4)$$

3) C_c: cost factor based on edge curvature. This factor is used to smooth or eliminate twisted edge. Its value is obtained through inspecting the local edge structure of the non-endpoint edge pixel (at least has two neighborhood territory edge pixels) on (i,j). If the pixel on (i,j) only has one pair of neighborhood territory edge pixel, which makes the flip angle of the edge structure surpass 45°, then $C_c(i,j)$=1; if the flip angle is 45°, then $C_c(i,j)$=0.5; if the flip angle is 0°, then $C_c(i,j)$=0;

$$C_c\begin{cases}0.0, edge\ curvature=0\\ 0.5, edge\ curvature=45\\ 1.0, edge\ curvature\geq 90\end{cases}. \quad (5)$$

4) C_f: cost factor based on edge fragment. This factor is used to connector eliminate the fragment edge. If the edge pixel on(i,j) has no neighborhood territory edge pixel, then $C_f(i,j)$=1; if has only one neighborhood territory edge pixel, then $C_f(i,j)$=0.1;for other situations, $C_f(i,j)$=0

$$C_f=\begin{cases}0, & more\ than\ one\ neighborhood\ territory\ edgepixels\\ 0.5, & only\ one\ neighborhood\ territory\ edgepixels\\ 1, & no\ neighborhood\ territory\ edge\ pixels\end{cases} \quad (6)$$

5) C_e: cost factor based on edge pixel amount. The former cost factor C_d rewards the edge pixel detection with a $f(R1,R2)$ value not equal to zero, which can lead to too much edge pixel amount being detected. C_e factor can restrain such tendency. If the pixel on (i,j) is edge pixel, then $C_e(i,j.)$=1,if not, $C_e(i,j)$=0.

$$C_e=(i,j)=\begin{cases}1, is\ edge\ pixel\\ 0, is\ not\ edge\ pixel\end{cases}. \quad (7)$$

Each coat factor can be regarded as met the various requirements that often conflict each other of the final edge image. Therefore, the final edge image can be regarded as the optimized image that meets all the restraints caused by the cost factor, that is, regarded it as a restraint optimization. The cost function on each pixel position is:

$$F(i,j)=\sum_k w_k C_k, k\in\{d,t,c,f,e\}, \quad (8)$$

where, w_k is the weights of the corresponding cost factor, and is usually taken as: w_d=2, w_e=1, w_f={2,3,4}, w_c={0.25,0.50,0.75}, $w_1=2w_f-w_c+w_d+0.01$.

Then, the cost function of a whole image of $M\times N$ is:

$$F=\sum_{i=1}^{M}\sum_{j=1}^{N}F(i,j). \quad (9)$$

4.4 COST FUNCTION MINIMIZATION BASED ON QUANTUM GENETIC ALGORITHM

4.4.1 Two-dimensional chromosome encoding and fitness function

Each individual in the population is correspond to a edge image, which is shown by the two-dimensional Boolean Matrix. Its fitness function can be calculated by the following equation:

$$f[i]=(F[w]-f[i])^n, \quad (10)$$

where $F[w]$ refers to the cost function of the worst individual, $F[i]$ refers to the cost of the No. i individual, n determines the difference among the fitness. The larger the n, the larger the difference.

4.4.2 Population initialization

This most crucial thing of population initialization is the ascertain of the population size. Population initialization refers to generating the population randomly, and each gene locus of each individuals is generated randomly. However, when seeking optimal solutions for some high-dimensional and multi-modal functions, since the function itself has multiple poles, and the position of the global optimum is unknown, it requires to make the initial population evenly cover the entire feasible region when initializing the population. For binary bit strings with image size of 256×256, and population individual of 256×256, 0 means the pixel dot is not an edge dot, and 1 means the pixel dot is an edge dot, and take the population size as 300.

4.4.3 Selection, crossover and mutation operators

Selection operator refers to the process of selecting individuals with stronger fitness to form a new population. The genetic algorithm uses selection to eliminate the weak ones according to the evaluation mechanism of fitness function. It gives individuals with stronger fitness higher probability to be inherited by the next generation, while individuals with weak fitness have lower probability, which continuously improves the fitness of the individual and enable it to approach the optimal solution continuously.

Crossover operator simulates the mating recombination of living things to crossover the individuals in a population. It continuously generates new individuals to maintains the diversity of the population, which enlargers the searching range of the solution and enhances the optimization ability of the algorithm. Since the representation of the individual is two-dimensional, the crossover operator adopts the two-dot crossover, which randomly selects two dots respectively in the X direction and Y direction for crossover algorithm. The design of the crossover operator mainly includes issues like the ascertain of the position of the crossover dot, how to operate the gene of the crossover dot, etc. The value of the crossover probability usually ranges from 0.4 to 0.99.

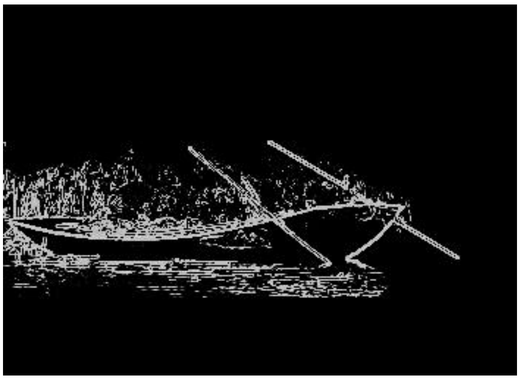
Mutation operator mutates each gene locus independently by a certain possibility. It generates new individuals through conducting special changes to one or several gene loci of the individual by relatively small probability. The mutation probability directly determines the usage frequency of the mutation operation in the algorithm. If the mutation probability is too high, the performance of the algorithm is close to that of the random searching algorithm; if it is too low, the algorithm cannot predict the prematurity. The value of the mutation probability usually ranges from 0.0001 to 0.1.

5 Results and performance analysis

All the detection results in this paper are obtained under the Matlab2012 environment. The original image is the level 256 grayscale image with a size of 256x256. (b), (c) and (d) in Figure 6 are the edge detection results of image (a) obtained respectively through QGA, Sobel operator and Canny operator. Sobel operator did not detect parts with smaller grayscale change. The image detected through Canny operator has too many details and too strong edge continuity, and certain details are distorted. It can be seen that the algorithm proposed by this paper can effectively detect parts with smaller grayscale change, and the detection results are more accurate and the algorithm is more efficient. However, it can be seen that "dot accumulation" appears on some parts of the edge, which is cause by the local optimum solution occurring in QGA searching.

a)

b)



c)

d)

FIGURE 6 a) Original image; b)Result of detection using QGA; c) Result of detection using Sobel operator; d) Result of detection using Canny operator

6 Conclusion

This paper described the current application of genetic algorithm in image processing, introduced in detail the application process of quantum genetic algorithm in image edge detection, and carried out a detailed discussion on the impact of each parameters in quantum genetic algorithm to the detection results. It also conducted a computer simulation experiment for the algorithm, and compared its detection results with that of the traditional edge detection operator, which proved the efficiency of this approach.

Acknowledgments

The work was supported by the science and technology research projects of Chongqing Education Commission with its project number KJ132001 and its project name Research on Fixed Assets Management System Based on Internet of Things.

References

[1] Williams I, Bowring N, Svoboda D 2014 A Performance Evaluation of Statistical Tests for Edge Detection in Textured Images *Computer Vision and Image Understanding* **122** 115-30

[2] Zareizadeh Z, Hasanzadeh R P R, Baghersalimi G 2013 A Recursive Color Image Edge Detection Method Using Green's Function Approach *International Journal for Light and Electron Optics* **124**(21) 4847-54

[3] Harikiran J, Narasimharao Y, Saichandana b, Lakshmi p v, Kumar R K 2012 Spot Edge Detection in Microarray Images Using Bi-Dimensional Empirical Mode Decomposition *Procedia Technology* **4** 227-31

[4] Chaira T 2012 A Rank Ordered Filter for Medical Image Edge Enhancement and Detection Using Intuitionistic Fuzzy Set *Applied Soft Computing* **12**(4) 1259-66

[5] Lopez-Molina C, De Baets B, Bustince H 2014 A Framework for Edge Detection Based on Relief Functions *Information Sciences* **278**(10) 127-40

[6] Lim T Y, Ratnam M M 2012 Edge Detection and Measurement of Nose Radii of Cutting Tool Inserts From Scanned 2-D Images *Optics and Lasers in Engineering* **50**(11) 1628-42

[7] Takahashi K, Kurokawa M, Hashimoto M 2014 Multi-layer Quantum Neural Network Controller Trained by Real-Coded Genetic Algorithm *Neuro computing* **134**(25) 159-64

[8] Dey S, Bhattacharyya S, Maulik U 2014 Quantum Inspired Genetic Algorithm and Particle Swarm Optimization Using Chaotic Map Model Based Interference for Gray Level Image Thresholding *Swarm and Evolutionary Computation* **15** 38-57

[9] Barberot C, Boisson J C, Gérard S, Khartabil H, Thiriot E, Monard G, Hénon E 2014 A Tool Coupling A Linear-Scaling Quantum Method with A Genetic Algorithm for Exploring Non-Covalent Interactions *Computational and Theoretical Chemistry* **1028**(15) 7-18

[10] Soleimanpour-moghadam M, Nezamabadi-pour H, Farsangi M M 2014 A Quantum Inspired Gravitational Search Algorithm for Numerical Function Optimization *Information Sciences* **267**(20) 83-100

[11] Ibrahim A A, Mohamed A, Shareef H 2014 Optimal Power Quality Monitor Placement in Power Systems Using An Adaptive Quantum-Inspired Binary Gravitational Search Algorithm *International Journal of Electrical Power & Energy Systems* **57** 404-13

[12] Dey S, Saha I, Bhattacharyya S, Maulik U 2014 Multi-level Thresholding Using Quantum Inspired Meta-Heuristics *Knowledge-Based Systems* **67** 373-400

Author

Lin Tang, September 1982, China.

Current position, grades: docent at Chongqing Vocational Institute of Engineering, China.
University studies: master's degree in Chongqing University, China in 2010.
Scientific interests: computer software and computer application.