Two-Dimensional Barcode Decoration Based on Real-Coded Genetic Algorithm

Satoshi Ono, *Member, IEEE*, Kensuke Morinaga, and Shigeru Nakayama, *Non-Members, IEEE*

***Abstract*— This paper proposes a system for decorating 2-dimensional barcode with some illustrations inside the code without detracting machine-readability and stored information. The proposed system formulates the task of finding appropriate positions, scales, and angles of illustrations, and solves the task by using real-coded genetic algorithm. The proposed system also uses multiple barcode decoder with the aim of improving decode feasibility of the decorated barcode. Experiments have shown that the proposed system can decorate barcodes with three illustrations, and that using more than one decoder can improve decoding feasibility of the decorated barcodes.**

I. Introduction

Barcode is used for various purposes such as merchandise control, book collection control in libraries, and so on. QR code (Quick Response code)[1], a kind of two-dimensional barcode investigated by Denso Wave, is currently used in Japan as a so-called shortcut to get a URL, e-mail address, phone number and so on. Most Japanese mobile phones have cameras and QR code scanner. By holding a mobile phone over QR codes printed on papers, billboards, or television screens, users can get decoded information and browse Web sites, or send e-mails without typing URLs or e-mail addresses on their mobile phones.

Although QR code can involve various information such as URL, e-mail address, short sound, and so on, users cannot know what kind of information is implanted in it (from QR code itself). In addition, QR code is unattractive and occupies an area in limited, worthy space of papers, billboards, or other media.

QR code has an error correction function which can supplement at most 30% data loss. An illustration therefore can be put inside a QR code. QR code decorated with an illustration is more attractive than a general, plain QR code. In addition, it can let humans know what kind of information is implanted in the code.

However, putting an illustration into QR code often affects machine-readability or damages implanted information. It is difficult to find an appropriate place in order to let the error correction work well and keep machine-readability. Larger illustrations or more images make this problem more difficult.

This paper proposes a system for generating two-dimensional barcode incorporated with some illustrations inside the code without affecting machine-readability and stored information. We formulate the task of finding appropriate position, scale, and angle of an illustration, photo, logo or other image item put into QR code as an optimization problem. By using evolutionary computation algorithm, the proposed system can find positions in which a given image item can be merged without damaging machine-readability and stored information. The proposed system can also incorporate more than one image into a QR code. Experiments have shown that our system can generate QR codes containing three illustrations without damaging QR code function and embedded data, and that using more than one decoder can improve decoding feasibility of the decorated QR codes.

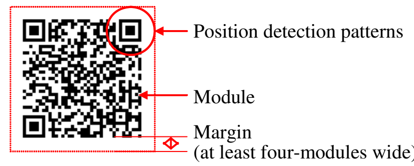


Fig. 1. QR code.

II. QR code

A. Overview

QR code, as shown in Figure 1, is a two-dimensional barcode created by Denso-Wave, and is the most popular type of two-dimensional codes.

QR codes are now used in broader contexts expanding for convenience-oriented applications aimed at mobile phone users. QR code storing addresses and URLs may appear in magazines, on signs, buses, business cards or just about any object that a user might need information about. A user having a phone equipped with a camera and decoder software[2] can scan the image of the QR code causing the phone browser to launch and redirect to the programmed URL. This act of linking from physical world object is known as a hardlink or physical world hyperlinks[1]. A user can also generate and print their own QR code for others to scan and use by visiting one of several free QR code generating sites or by using QR code generation software.

QR Code is capable of handling various types of data, such as numeric and alphabetic characters, Japanese characters, symbols, binary, and control codes. Up to 7,089 characters can be encoded in one symbol.

QR code has error correction capability. Even if the symbol is partially dirty or damaged, QR code can correct

Satoshi Ono, Kensuke Morinaga, and Shigeru Nakayama are with Department of Information and Computer Science, Faculty of Engineering, Kagoshima University, Japan (phone: +81-99-285-8453; fax: +81-99-285-8464: email: {ono, sc104059, shignaka}@ics.kagoshima-u.ac.jp).

[1]QR code is trademarked by Denso Wave, inc.

[2]QR code software decoder example:
The Kaywa Reader http://reader.kaywa.com/

978-1-4244-1823-7/08/$25.00©2008 IEEE

i) A QR code example on a floor guide.

ii) The enlargement of the QR code.

iii) Reservation system of the theater linked by the code.

(a) A case study of a floor guide.

i) A QR code example on a hamburger package.

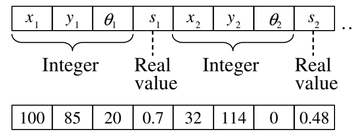
ii) The enlargement of the QR code.

iii) Information obtained by scanning the QR code.

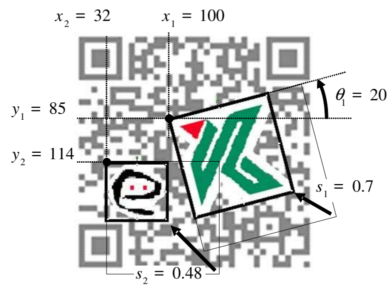
(b) A case study in a hamburger shop.

Fig. 2. Case studies of QR codes.

$$(x_1, y_1, \theta_1, s_1, x_2, y_2, \theta_2, s_2)$$
$$= (100, 85, 20, 0.7, 32, 114, 0, 0.48)$$



(a) Genotypic representation.



(b) Phenotypic represenatation.

Fig. 3. Chromosome representation.

errors approximately 7%, 15%, 25%, and 30% of codewords at Level L, M, Q, and H, respectively.

B. Case studies of QR codes

In recent years, QR codes' use has spread widely throughout Japan due to the dissemination of cameras and software decoders on mobile phones. For instance, the QR code on a floor guide of a shopping mall, as shown in Figure 2(a), enables customers to get detail information of stores in the mall, latest events and news. The customers can also access the reservation system of the theater in the mall. The QR code on a food item's package, as shown in Figure 2(b), enables customers to get calorie chart, nutrition facts, and allergies for this product.

III. Proposed QR-code design system

A. Basic idea

The principles of the proposed system for generating 2-dimensional bar code with illustrations are as follows:

1) **Formulating the task of decorating QR codes by placing illustrations into the code as an optimization problem.**
Deciding the positions of illustrations requires iteration of trial production of the barcode with the illustrations and test for decoding practicability. This is because we cannot know the appropriate places where the illustrations should be placed in the barcode in advance. We therefore formulate this task as an optimization problem. The objective function is composed of feasibility of decoding evaluated by QR code decoders, and adequacy of positions and other parameters of the incorporated image based on human-readability.
2) **Using real-coded genetic algorithm to find an appropriate image position.**
Genetic algorithm (GA)[2] is a well-known meta-heuristics[3] for optimization inspired by evolutionary biology. GA can produce better solutions by recombining good "building blocks"[2]. Traditionally, genes are represented in binary, but real values and other encodings have also become possible in recent years. Because positions and other parameters providing a barcode with illsutrations should be real numbers instead of sequences of binary variables, the proposed system utilizes real-coded GA[4], [5], [6].
3) **Evaluating solution candidates by multiple QR code decoders.**
Most recent mobile phones on the market in Japan involve a QR code decoder software developed by mobile phone manufacturers or other subcontract software developer. Although it is hard to prepare and use the same QR code decoders in our system as mobile phones use, we try to ensure the robustness of generated QR codes by using multiple decoders. The more QR code decoders the proposed system uses, the more mobile phones can read the generated codes.

B. Chromosome representation

QR code design problem involves continuous variables of which the number is $4 \times N_i$, where N_i indicates the number of illustrations: x- and y-axis positions x_i, y_i , an inclination θ_i, and a scale s_i for each illustration i. A chromosome,

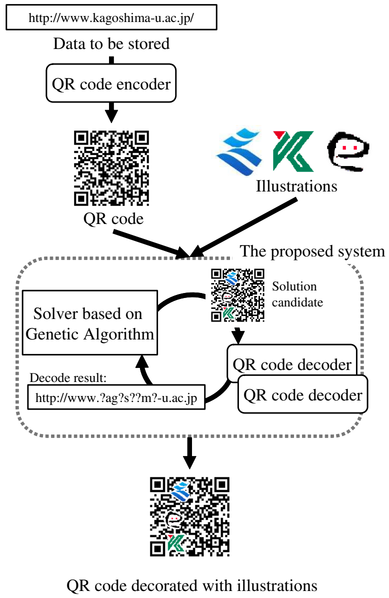


Fig. 4. The proposed system.

Step 1: Make initial population by random.
  Step 2: **Repeat** until the stop condition is satisfied.
    Step 3: Evaluate all individuals with decoders.
      Step 3-1: Generate decorated QR codes.
      Step 3-1: Decode the decorated QR codes.
      Step 3-1: Calculate fitness of the code.
    Step 4: Select elite individuals to preserve.
    Step 5: Crossover by BLX-α.
    Step 6: Mutation

Fig. 5. Pseudo-code of the proposed system.

i.e. genotypic representation, consists of the variables coded as integers and real values. Positions and an inclination are coded as integers and scales as real values. The phenotypic representation is formed by placing illustrations along the variables, as shown in Figure 3.

C. Process flow

Figure 4 shows the outline of the proposed system. The inputs of the proposed system are a QR code and illustrations. The QR code can be easily generated by using open software or services on various web sites.

The proposed system iterates generation of decorated QR codes and evaluation of the QR codes by using software decoders. Real-coded GA[4], [5], [6] is adopted to optimize the illustration placement, as shown in Figure 5. The proposed method utilizes blend crossover (BLX-α), a simple crossover operator for real-coded GA proposed by Eshelman[6]. The algorithm shown in Figure 5 stops when fitness of the best solution reaches 1.0, or the number of generations reaches the pre-defined limit.

D. Crossover: BLX-α

BLX-α blends the numeric parameters of the two parents by randomly choosing, for each parameter, a point on a discretized interval defined by the values of the parents' parameters. The value for the offspring's parameter is generated by randomly choosing a point within the discretized interval $[p_1 - \alpha I, p_2 + \alpha I]$ where p_1 and p_2 are values of parents, $I = |p_1 - p_2|$, and αI is a distance extending past the parameter value of the parents.

E. Fitness calculation with QR code decoders

The proposed system uses more than one QR code decoder to evaluate individuals, which are QR codes decorated with illustrations inside, for the sake of enhancing robustness of the decoding feasibility of the decorated code. Fitness $F(x)$ of an individual x is a product of success rate in decoding $P(x)$ and appropriateness of illustrations' positions $Q(x)$:

$$F(x) = P(x) \times Q(x) \tag{1}$$

$P(x)$ indicates how x can be decoded properly and is calculated as following equation and Figure 6:

$$P(x) = \frac{\sum_k p_k(x) \times w_k^{(p)}}{\sum_k w_k^{(p)}} \tag{2}$$

$p_k(x)$ is calculated from decode result of a decoder k; $p_k(x) = 1$ if a decoder succeeds in extracting information. In the case that a decoder k does not produce any data except the fact that the decoder failed to decode, $p_k(x) = 0$ when the decoder failed. In the case that a decoder k can present decode error amount, $p_k(x)$ is calculated by subtracting error rate from 1.0. $w_k^{(p)}$ is a weight parameter.

We implemented the proposed system with Psytec QR Code Decode Library [3] (Psytec decoder) and Open Source QR Code Decode Library [4] (open source decoder). Psytec library outputs information embedded in a given QR code only when the code is decodable, and does not produce any information when the code cannot be decoded. Open Source decode library outputs information embedded in the code even when the library cannot decode the code entirely. Thereby the decode error rate can be calculated by comparing the library's output and embedded information.

$Q(x)$ indicates how x places illustrations appropriately in its QR code, and calculated as follows:

$$Q(x) = \prod_l q_l(x)^{w_l^{(q)}} \tag{3}$$

The proposed system uses following four functions to calculate $Q(x)$. $q_1(x)$ indicates how illustrations are overlapped each other and is calculated by following equation:

$$q_1(x) = 1 - \sum_{i=1}^{N_i} \frac{S_i^O}{S_i} \tag{4}$$

[3] http://www.psytec.co.jp/product/03/
[4] http://sourceforge.jp/projects/qrcode

TABLE I

PARAMETER CONFIGURATION.

Parameter	Value
Number of individuals	10
Number of elites	2
Crossover	BLX-α
α	0.2
R	$\begin{cases} 0.25 & N_i = 1 \\ 0.3 & Otherwise \end{cases}$
Crossover rate	0.8
Mutation rate	0.03
Generations	$N_i \times 200$
$(w_1^{(p)}, w_2^{(p)})$	(0.2, 0.8)
$(w_1^{(q)}, w_2^{(q)}, w_3^{(q)}, w_4^{(q)})$	(0.5, 0.5, 1.0, 0.5)

where N_i is the number of illustrations, S_i is a size of illustration i, and S_i^O is an area size of illustration i covered by other illustrations. S_i is approximately calculated as the size of the rectangle circumscribing illustration i.

$q_2(x)$ indicates how illustrations are placed appropriately inside of QR code and is calculated by following equation:

$$q_2(x) = 1 - \sum_{i=1}^{N_i} \frac{S_i^B}{S_i} \tag{5}$$

where S_i^B indicates an area size of illustration i sticking out from QR code.

$q_3(x)$ indicates how large illustrations are magnified or reduced and is calculated by following equation:

$$q_3(x) = \min\left(\frac{\sum_{i=1}^{N_i} S_i}{R \times S_{QR}}, 1 \right) \tag{6}$$

where R is a constant which should be set to almost 0.3 due to the error correction capacity of QR code, and S_{QR} is an area size of QR code.

$q_4(x)$ indicates whether illustrations are magnified or reduced with the same scale between illustrations, and calculated by following equation:

$$q_4(x) = \frac{\min_{i=1,\ldots,N_i} s_i}{\max_{i=1,\ldots,N_i} s_i} \tag{7}$$

where $\max\{s_i\}$ and $\min\{s_i\}$ are the highest and lowest scale parameter of all illustrations respectively. $w_l^{(q)}$ is a weight parameter.

IV. EVALUATION

We tried to decorate QR code by implanting one to three illustrations. The experiments were performed by using Psytec library only, open source library only, or both of them. We performed five runs and generated five QR codes with illustrations for each number of illustrations. The QR code and illustrations shown at the top part of Figure 4 were used; the URL "http://www.ics.kagoshima-u.ac.jp/" was stored into the QR code with level H error correction, and

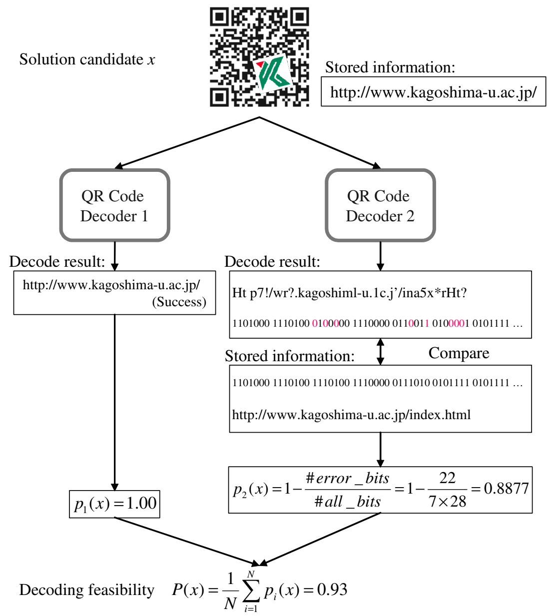


Fig. 6. Evaluation with multiple decoders.

the illustrations were from left to right the symbols of Kagoshima prefecture, Kagoshima University, and Faculty of Engineering. The number of individuals, the number of elites, and the generation limit were set to 10, 2, and $N_i \times 200$, respectively, where N_i is the number of illustrations as shown in Table I. Weights of open source decoder and Psytec decoder, $w_1^{(p)}$ and $w_2^{(p)}$, were set to 0.2 and 0.8 respectively. This was because the existence of some bugs of open source decoder had been pointed out and Psytec decoder was more reliable than open source decoder.

The generated QR codes were evaluated with ten mobile phones on the Japanese market. Each QR code was printed out on a paper using a color-laser printer, and mobile phones were used to try scanning and decoding without any specific lighting condition – under fluorescent office lighting – and without being fixed.

Figure 7 shows the success rates in decoding by the mobile phones with bar graphs, and the processing time to generate the QR codes with line graphs. The average success rates in decoding and the average processing time are shown in Table II. Overall, the QR codes generated with two decoders were well-decoded by the mobile phones, whereas some of the codes generated with Psytec decoder only or open source decoder only could not be decoded by the mobile phones. This result shows that using multiple decoders can improve the robustness of generated QR codes, i.e. QR codes generated with several decoders can be decoded by various mobile phones.

Figure 8 shows transitions of average $P(x)$, $Q(x)$, $F(x)$, and rates of total illustration size to QR code size. When placing one illustration, the proposed system can quickly find

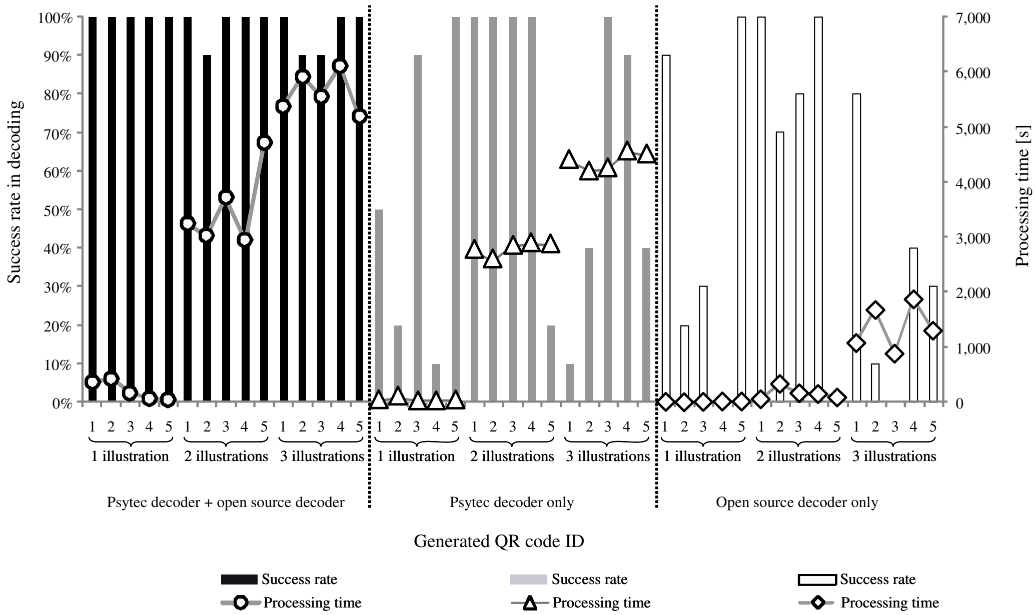


Fig. 7. Experimental results on success rate in decoding and processing time.

TABLE II
AVERAGE SUCCESS RATE IN DECODING AND PROCESSING TIME.

	Psytec decoder + Open source decoder		Psytec decoder only		Open source decoder only	
	Success rate	Processing time [s]	Success rate	Processing time [s]	Success rate	Processing time [s]
1 illustration	100%	210	54%	34	48%	10
2 illustrations	98%	3,524	84%	2,768	70%	149
3 illustrations	96%	5,623	56%	4,363	32%	1,361

solutions whose fitness is 1.0 within 50 generations. As the increase of number of illustrations, it becomes difficult to find a solution x with $F(x) = 1.0$. The rates of total illustration size to QR code size were kept at almost 0.3, the same rate as the error correction capability that a QR code of level H has.

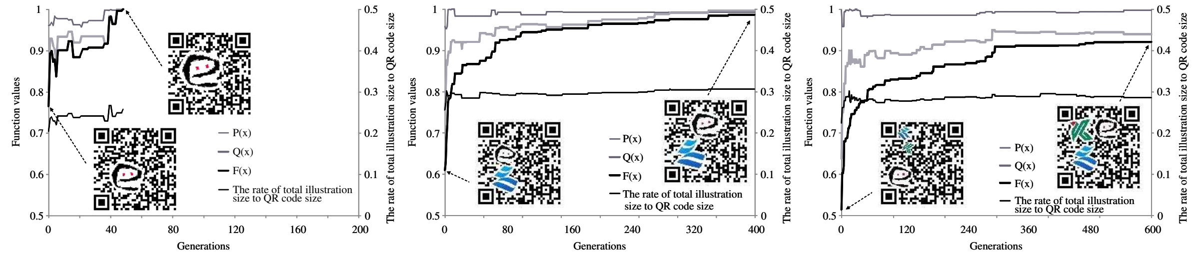
QR codes involving one illustration generated with one decoder are quickly generated as shown in Figure 8 (b) i) and (c) i). But some of the QR codes involving one illustration were difficult to decode, whereas the QR codes involving two illustrations were easy to decode as shown in Figure 7.

Most of QR codes generated with two decoders were well-decoded by the mobile phones, whereas only one decoder sometimes failed to generate QR codes in which one or three illustrations were placed. The second QR code with two illustrations was hard to decode by a mobile phone. In addition, two mobile phones could not succeed in decoding the second or third QR code with three illustrations. The reason can be that the position detection pattern on the left-bottom corner is partly covered by an illustration, which makes it difficult to recognize precise code area for the mobile phones.

The process of decoding a QR code by a mobile phone is composed of three procedures: scanning the image involving the QR code, recognition of the QR code from the scanned image, and decoding the recognized QR code. But the proposed system can only simulate the last part of the above process. The optical performance of the cameras mounted on the mobile phones would differ from device to device. Image processing methods for recognition of QR code is also differ from each other in mobile phones. It is therefore difficult to generate a decorated QR code which can be decoded by all mobile phones.

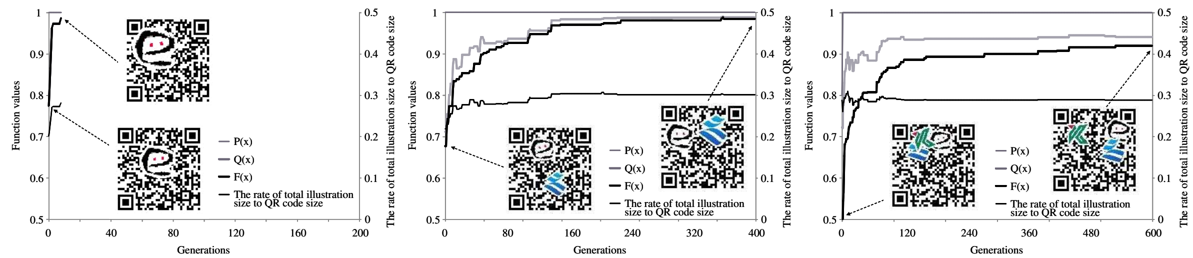
Older design or dirt produced by long time use of built-in camera, or older image processing software might disturb the recognition of the QR codes. In fact, the average success rates of older mobile phones manufactured in 2005 were less than 70%, whereas the average success rates of other newer mobile phones were almost 75%.

From Figure 7 and Table II, the processing time to decorate the QR codes with two decoders was longer than the total sum of the processing time with Psytec decoder only and open source decoder only.



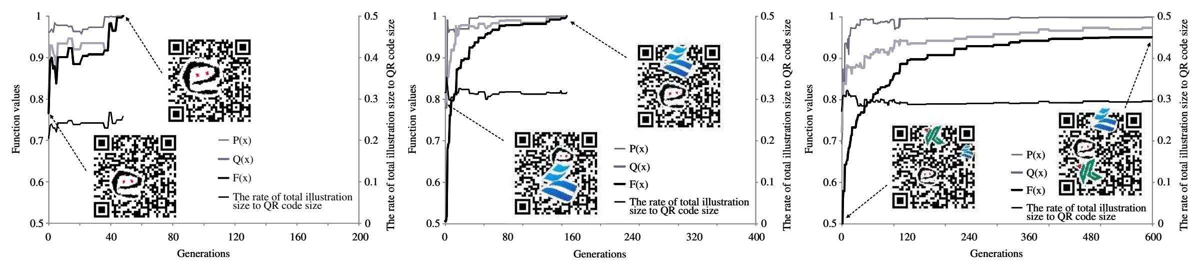
i) 1 illustration. ii) 2 illustrations. iii) 3 illustrations.

(a) QR codes generated with Psytec decoder and open source decoder.



i) 1 illustration. ii) 2 illustrations. iii) 3 illustrations.

(b) QR codes generated with Psytec decoder only.



i) 1 illustration. ii) 2 illustrations. iii) 3 illustrations.

(c) QR codes generated with open source decoder only.

Fig. 8. Fitness transitions.

V. Conclusions

Proposed in this paper is a QR code design system that places illustrations, photographs, logo or similar inside the QR code. The proposed system utilizes real-coded GA to optimize illustration positions, scales, and angles, and software decoders of QR codes to evaluate generated QR codes involving illustrations.

Experiments have shown that the proposed system generates QR codes with illustrations which can be readable by various mobile phones on the Japanese market.

In the future, we plan to combine interactive evolutionary computation [7] and multiple solution search method [8], and apply the combined model to the proposed QR code generation system.

References

[1] M. Rohs, "Real-world interaction with camera phones," in *in Proceedings of 2nd International Symposium on Ubiquitous Computing Systems 2004*, 2004, pp. 74–89.

[2] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley, Reading, 1989.

[3] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM Computing Surveys*, vol. 35, no. 3, pp. 268–308, 2003.

[4] A. H. Wright, "Genetic algorithms for real parameter optimization," in *Foundations of genetic algorithms*, G. J. Rawlins, Ed. San Mateo, CA: Morgan Kaufmann, 1991, pp. 205–218. [Online]. Available: citeseer.ist.psu.edu/wright91genetic.html

[5] L. J. Eshelman and J. D. Schaffer, "Real-coded genetic algorithms and interval-schemata," in *Foundations of Genetic Algorithms*, vol. 2, no. 1993, pp. 187–202.

[6] L. J. Eshelman, K. E. Mathis, and J. D. Schaffer, "Crossover operator biases: Exploiting the population distribution," pp. 354–361, 1997.

[7] H. Takagi, "Interactive evolutionary computation – fusion of the capabilities of ec optimization and human evaluation," in *in Proceedings of the IEEE*, vol. 89, 2001, pp. 1275–1296.

[8] S. Ono, Y. Hirotani, and S. Nakayama, "Multiple solution search based on hybridization of real-coded evolutionary algorithm and quasi-newton method," in *in Proceedings of IEEE Congress on Evolutionary Computation*, 2007, pp. 1133–1140.