2013 International Computer Science and Engineering Conference (ICSEC 2013)

Web Page Template Design Using Interactive Genetic Algorithm

Davy Sorn, Sunisa Rimcharoen
Faculty of Informatics, Burapha University, Chonburi, 20131
Email: davysorn@gmail.com, rsunisa@buu.ac.th

Abstract—**Nowadays, there are many new web sites occur daily on the Internet and the demand for web page creation and design increases. The problem is it takes time to create and design the web pages. Moreover, designing web page templates to satisfy user preference is difficult. In order to reduce time and allow users to interact with the design process, we propose the web application which applies interactive genetic algorithm to generate web page templates and allow users to get involved with the system. The proposed algorithm evolves HTML files for the content and CSS style sheets for a presentation format. Users can rate each section of a web page template according to their preferences. The given scores feed back to the system and the generated template is evolved. The experiments are conducted by assuming user's preferential criteria. We designed score-rules to test the algorithm whether it can create web page templates to satisfy the criteria or not. The experimental results show that the proposed algorithm can generate web page templates that meet the criteria.**

Keywords: interactive genetic algorithm, website design, web site design, webpage design, web page template, web template design.

I. INTRODUCTION

Web sites are very popular at present. Many new web sites occur daily on the Internet [1]; thus, the requirements to create and design web pages are increased. These require users to know programming languages and tools for developing and designing web pages. It takes time to develop and design web pages as their need. In order to reduce time from both creating and designing web pages, they have to find a new solution by using web page templates. Web page templates consist of HTML files and CSS files. HTML files hold the structure and content of the page and CSS Style Sheet files hold the presentation styles of pages. Therefore, we propose a new web application which is developed by applying interactive genetic algorithm which requires the involvement of users to interact with the web application. The interactive genetic algorithm has been applied to many applications in design such as a fashion design which is used to model women's dress [2], a Japanese Kimono design to model Yukata which is a traditional Japanese garment often worn in mid-summer [3], a font generation system which is designed to emerge various fonts based on user's Kansei without hand drawing [4], evolving colors in user interfaces to search for a solution that provides a good trade-off between aesthetics and accessibility requirements [5], an office layout support system which can generate not only in square space but also in polygonal space [6], a user interface design which evolve user interfaces in the XUL interface definition language which is a user interface markup language developed by the Mozilla project [7], a sign sound design which is to generate melody based sounds freely and easily [8], a web site design system which has users to be involved in the process to generate web page [9, 12].

The difference between this work and the previous work [9, 12] is how to rate a web page template. In the previous work, users are involved in the evaluation process and they can rate a preferable web page template one at a time. The problem is we cannot know which section of the web page template they like or dislike. Thus, we have divided the web page template into many sections and users can rate each section of the web page template for what they like or dislike.

By applying interactive genetic algorithm, the proposed algorithm generates many web page templates with different layouts for users to choose. Users just rate each section of the web page templates, such as header, footer, sidebar, etc., to express their preference how much they like or dislike it. Then, the algorithm will evolve a new population of web page templates based on the old ones according to users' preferences Users can view and download web page templates which are created by the web application to apply these templates to their creation and design tasks.

The details will describe in the remaining of this paper which is organized as follows: section II describes the background of genetic algorithm and interactive genetic algorithm, section III presents the proposed method, section IV describes the experiments and results, and section V is a conclusion.

II. BACKGROUND

1) Genetic Algorithm

Genetic algorithm, which was first introduced by John Holland [10], is used to solve optimization and search problems by emulating principles of biological evolution. It represents a solution to a problem as a chromosome; then, it creates the initial population of solutions and uses genetic operators such as selection, crossover, and mutation to create offspring. The solutions are gradually improved by a selection scheme which selects the survivors by their fitness values.

2) Interactive Genetic Algorithm

Interactive genetic algorithm and genetic algorithm are similar, except an evaluation process. The interactive genetic algorithm requires users to give fitness to each individual instead of using a fitness function. The fitness here is a measure of how well each potential solution solves the problem at hand. The interactive genetic algorithm is applied to many application domains where the fitness function is difficult or impossible to design a computational fitness function [2, 11]. The flow chart of the interactive genetic algorithm is shown in the Fig. 1.

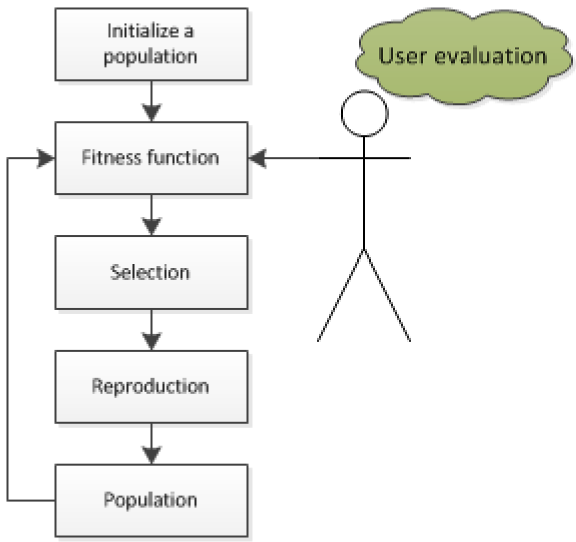

Figure 1. The flow chart of the interactive genetic algorithm.

III. The Proposed Method

A. *An Overview*

Fig. 2 shows the process of using the interactive genetic algorithm for generating web page templates. The web application creates an initial population of 10 web page templates which are the base population for creating the next population of web page templates. Then, it displays 10 web page templates to users. If the web page templates satisfy users, the process of the web application is finished; otherwise, users have to rate each section of every web page template for generating the next population. And then, the web application calculates a total score of every web page template and selects 2 web page templates which are not the same from the base population. A higher score template has more probability to be selected than a lower score template. After the 2 web page templates are selected, the web application generates a random value to decide which genetic operator is selected for producing an offspring. In this paper, we define a crossover rate as 0.7. If the random value is smaller than 0.7, a crossover operator is selected; otherwise, a mutation operator is selected. The web application continues producing many offspring until it reaches the size of the population which has 10 web page templates and displays the new offspring to users. The process continues doing many times and users are required to rate until requirements of users are met.

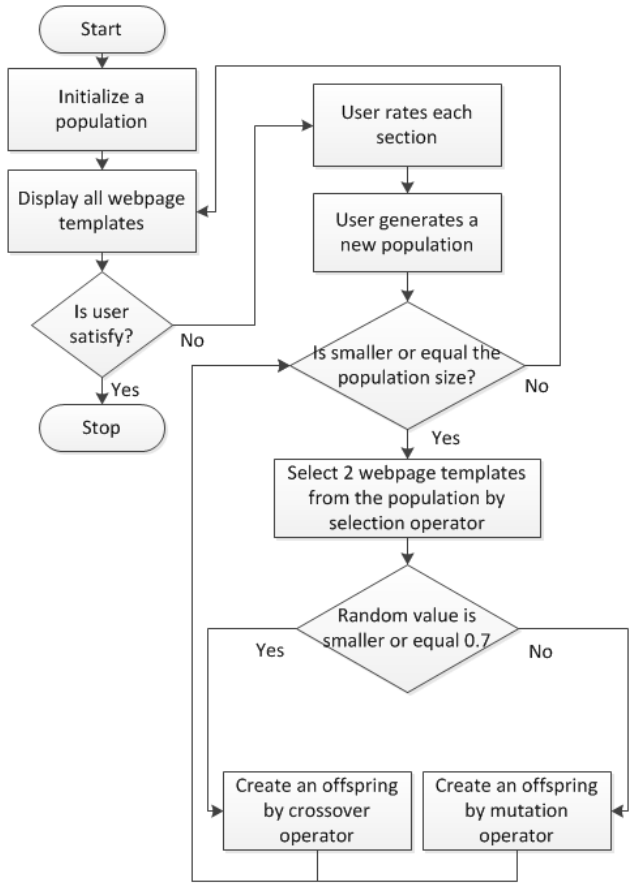

Figure 2. The overview of the proposed method.

B. *Genetic Encoding*

In this paper, each web page template is encoded as a chromosome which is divided into 2 parts, a layout part and a style part. The layout part consists of layout, container, header, navigation top, sidebar left, content, sidebar right, navigation bottom, and footer gene; the style part consists of body, header1, header2, header3, paragraph, list, image, anchor, and color scheme gene. Fig. 3 below shows the structure of the chromosome. All gene encodings are described in Table I and web page layouts are shown in Fig. 4.

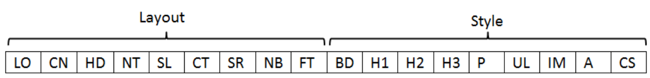

Figure 3. The chromosome encoding.

TABLE I. GENE ENCODING DESCRIPTION

Gene	Description
LO	Layout
CN	Container
HD	Header
NT	Top navigation
SL	Left sidebar
CT	Content
SR	Right sidebar
NB	Bottom navigation
FT	Footer
BD	Body
H1	Heading 1
H2	Heading 2
H3	Heading 3
P	Paragraph
UL	Bulleting
IM	Image
A	Anchor link
CS	Color scheme

C. Genetic Operators

1) Selection

We employ a tournament selection of size 2 for the selection process. It randomly selects 2 chromosomes which are not the same from a population and compares their total score which is calculated by summing all genes' score. A chromosome which has the highest total score is selected to be a parent for generating a next generation. Fig. 5 shows the process of the selection operator.

2) Crossover

The crossover operator requires 2 parents from a population to produce an offspring. First, it uses the selection operator to select 2 parents from the population and assigns all genes of a chromosome which has a higher total score to a child. It calculates a different value between the 2 parents as a percentage and generates a random percentage for comparing. If the random percentage is smaller than the difference percentage, the gene which has a high score is assigned to a child; otherwise, the gene which has a low score is assigned to the child. Fig. 6 shows the process of the crossover operator.

3) Mutation

The mutation operator requires 1 parent from a population to produce an offspring. First, it uses the selection operator to select 1 parent from the population. Then, it generates a random percentage and changes each attribute for all genes of the parent's chromosome if the random percentage is higher than 0.5.Fig. 7 shows the process of mutation operator.

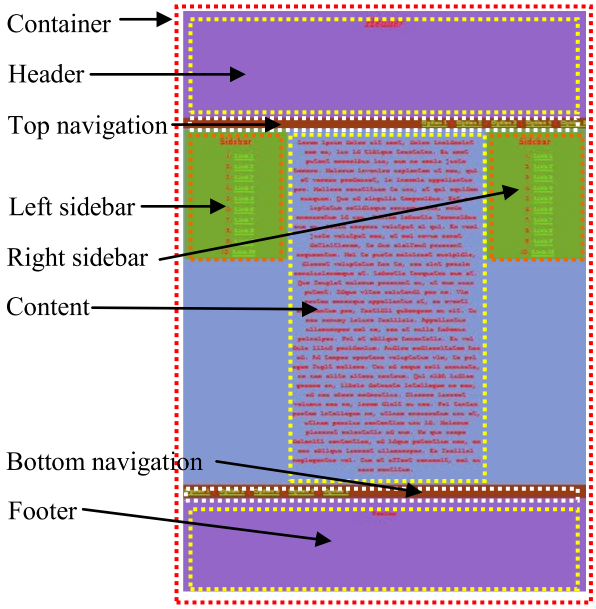

Figure 4. The web page layout.

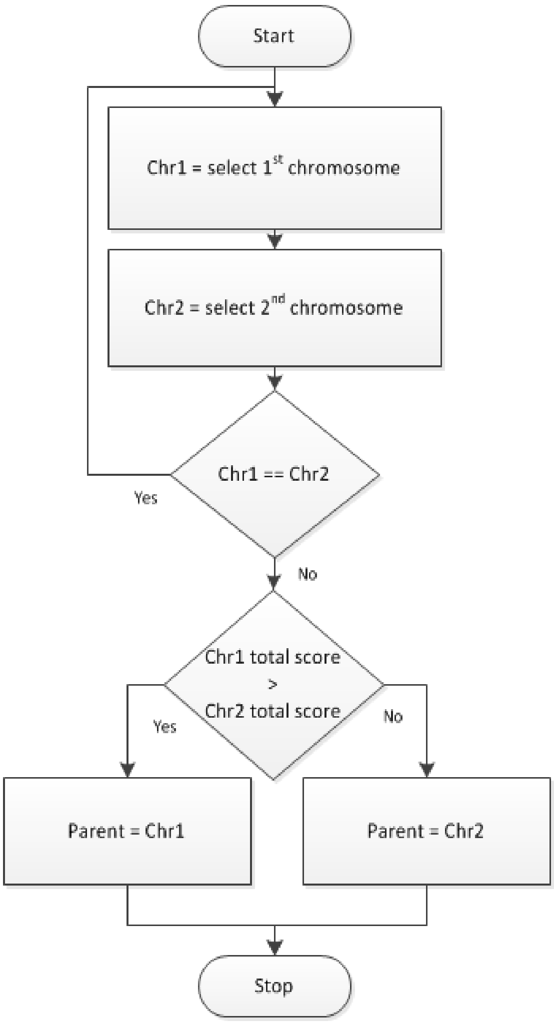

Figure 5. The selection operator.

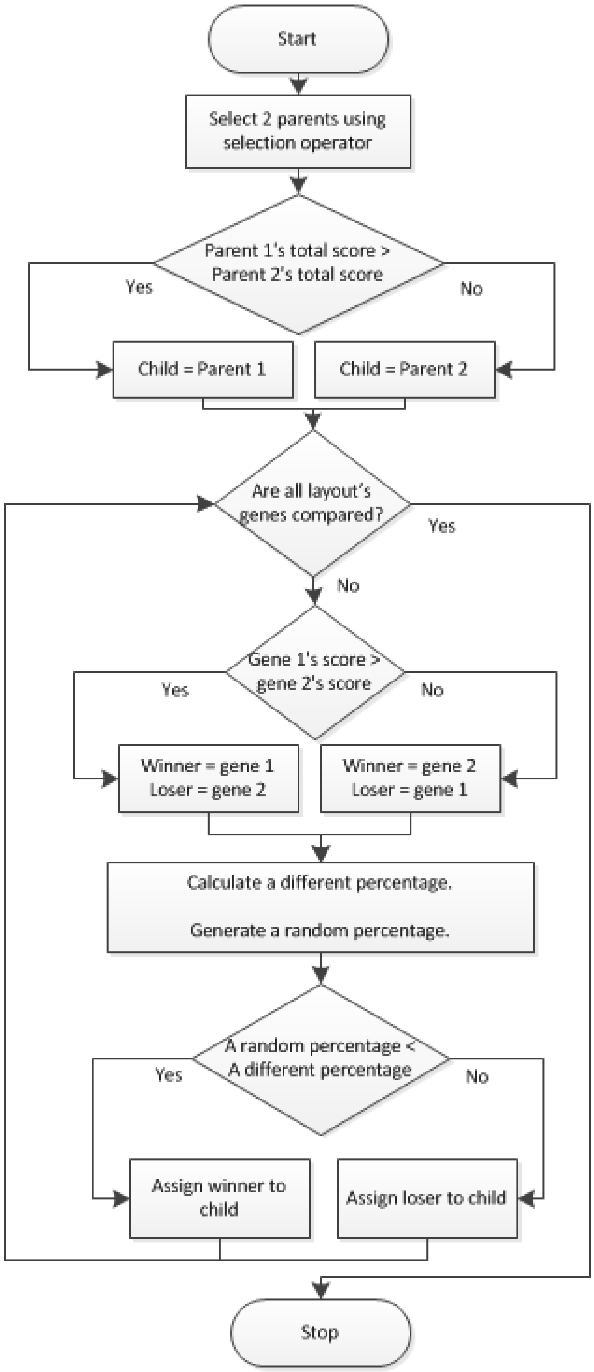

Figure 6. The crossover operator.

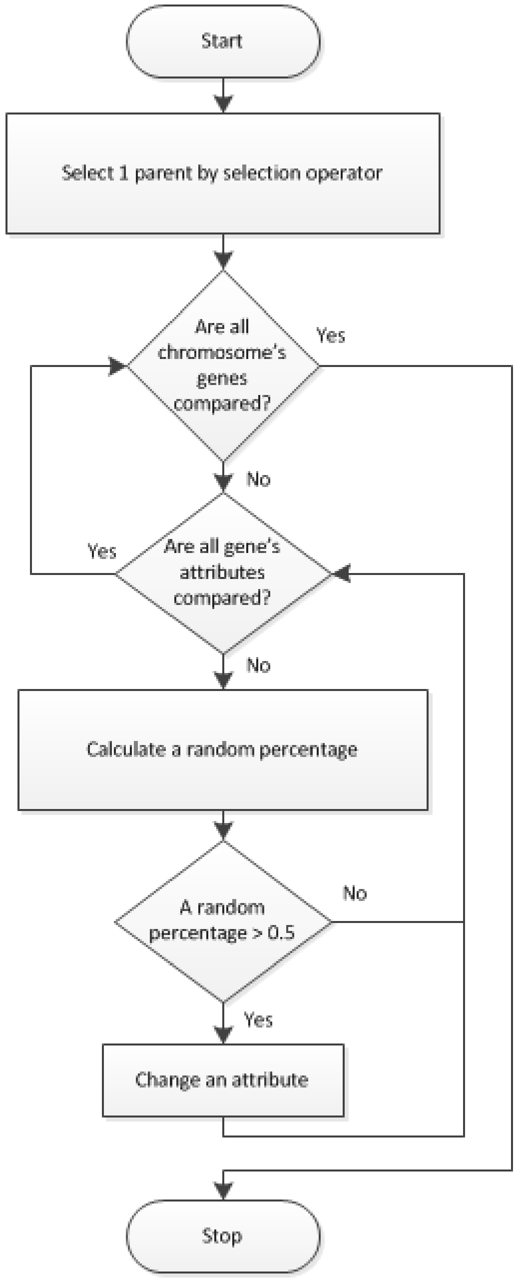

Figure 7. The mutation operator.

IV. Experiments and Results

We develop a web application to provide a web page template generating tool for users. For each generation, the web application generates 10 web page templates and the fitness values are given by the user. Fig. 8 shows the snapshot of the web application.

This section describes the experimental setup. In order to test whether the proposed algorithm can generate web page template that satisfy user preference or not, we assume that a user wants to generate web page templates which have 4 characteristics such as container, header, content, and right sidebar. The user can rate from 1 to 5 which 1 is very bad, 2 is bad and up to 5 which is very good. For the assessment of the algorithm, we have designed criteria which are shown in Table II – V. Table II shows criteria of the generated web page's container width, if the width is between 640 to 752 pixels, it will be rated by 1 and the others criteria are used in the same manner.

TABLE II. Container's width score

Characteristic	Width range in pixels	Score
Container	640 – 752	1
	752 – 864	2
	864 – 976	3
	976 – 1088	4
	1088 – 1200	5

TABLE III. Content's width score

Characteristic	Width range in pixels	Score
Content	40 – 272	1
	272 – 504	2
	504 – 736	3
	736 – 968	4
	968 – 1200	5

TABLE IV. Sidebar's width score

Characteristic	Width range in pixels	Score
Sidebar	100 – 140	1
	140 – 180	2
	180 – 220	3
	220 – 260	4
	260 – 300	5

TABLE V. Header's height score

Characteristic	Height range in pixels	Score
Header	100 – 140	1
	140 – 180	2
	180 – 220	3
	220 – 260	4
	260 – 300	5

Fig. 9 shows the average results of 30 runs. It shows that the fitness values are increased significantly for the first generation which randomly generates web page templates for users to evaluate. Most significantly, the proposed algorithm is able to generate web page templates that more correspond to the satisfaction of users as can be seen from the increased fitness in the next several generations. We experimented with collected data from 20 generations and we can see obviously that the fitness values are relatively constant and just changing only slightly after running 5 generations. Thus, in a real application users simply give scores to web page templates in a few times, the proposed method can improve the web page templates to converge users' preference.

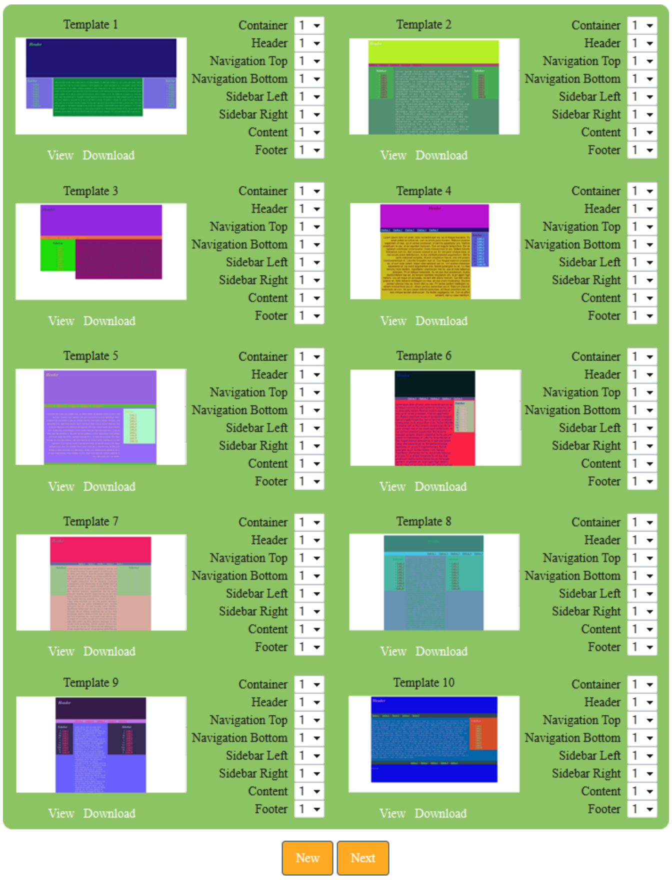

Figure 8. The snapshot of the web application

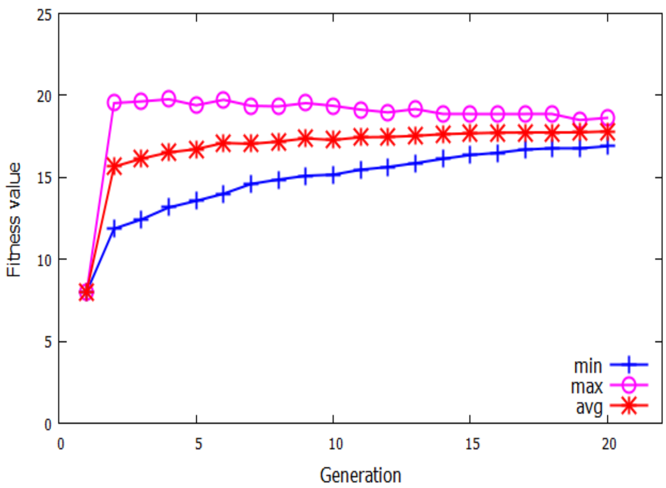

Figure 9. The fitness values of web page templates.

V. Conclusion

In this paper, we propose using interactive genetic algorithm to generate web page templates. Layout and style of a web page template are encoded in a chromosome which is divided into 2 parts. The first part is used to represent a HTML file and the second part is used to represent a CSS file. The main difference between this work and the previous work is that we allow users to rate each section of the generated web page template. By doing this, it can help the algorithm knows which section of the web page template users like. We conduct the experiments by assuming user preferences and score criteria. The experiments show that the web application can generate web page templates which satisfy users within five generations. For future work, color schemes should be implemented in order to make web page templates look more attractive.

References

[1] "December 2012 Web Server Survey | Netcraft." [Online]. Available: http://news.netcraft.com/archives/2012/12/04/december-2012-web-server-survey.html. [Accessed: 18-Jun-2013]..

[2] H.-S. Kim and S.-B. Cho, "Application of interactive genetic algorithm to fashion design," Engineering Applications of Artificial Intelligence, vol. 13, no. 6, pp. 635–644, Dec. 2000.

[3] M. Sugahara, M. Miki, and T. Hiroyasu, "Design of Japanese Kimono using Interactive Genetic Algorithm," in IEEE International Conference on Systems, Man and Cybernetics, 2008. SMC 2008, 2008, pp. 185–190.

[4] K. Yoshida, Y. Nakagawa, and M. Koppen, "Interactive genetic algorithm for font generation system," in World Automation Congress (WAC), 2010, 2010, pp. 1–6.

[5] C. Birtolo, P. Pagano, and L. Troiano, "Evolving colors in user interfaces by interactive genetic algorithm," in World Congress on Nature Biologically Inspired Computing, 2009. NaBIC 2009, 2009, pp. 349–355.

[6] Y. Araki and Y. Osana, "Office layout support system for polygonal space using interactive genetic algorithm — Generation of Layout Plans for Workspace —," in 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2012, pp. 1039–1044.

[7] J. C. Quiroz, S. J. Louis, A. Shankar, and S. M. Dascalu, "Interactive Genetic Algorithms for User Interface Design," in IEEE Congress on Evolutionary Computation, 2007. CEC 2007, 2007, pp. 1366–1373.

[8] M. Miki, H. Orita, S. H. Wake, and T. Hiroyasu, "Design of Sign Sounds using an Interactive Genetic Algorithm," in IEEE International Conference on Systems, Man and Cybernetics, 2006. *SMC '06*, 2006, vol. 4, pp. 3486–3490.

[9] A. Oliver, N. Monmarche, and G. Venturini, "Interactive design of web sites with a genetic algorithm," in In Proceedings of the IADIS Interational Conference WWW/Internet, 2002, pp. 355-362.

[10] Holland, J. H. 1975. Adaptation in Natural and Artificial Systems. University of Michigan Press. (Second edition: MIT Press, 1992.)

[11] "Interactive evolutionary computation," Wikipedia, the free encyclopedia. 03-Mar-2013

[12] T. Yokoyama, H. Takenouchi, M. Tokumaru, and N. Muranaka, "Website design system based on an interactive genetic algorithm using tournament evaluation by multiple people," in Proc. of SCIS-ISIS 2012, 2012, pp. 2260-2263.