Design of composite image filters using interactive genetic programming

Ma, Jingye
Graduate School of Design, Kyushu University

Takagi, Hideyuki
Faculty of Desgin, Kyushu University : Professor

https://hdl.handle.net/2324/1444584

出版情報：International Conference on Innovations in Bio-Inspired Computing and Applications (IBICA). 2012, pp.274-279, 2012-09-26. IEEE
バージョン：

Design of Composite Image Filters Using Interactive Genetic Programming

Jingye MA
Graduate School of Design
Kyushu University, Fukuoka, Japan
Email: majingyesoft@yahoo.co.jp

Hideyuki TAKAGI
Faculty of Desgin
Kyushu University, Fukuoka, Japan
URL: http://www.design.kyushu-u.ac.jp/~takagi/

***Abstract*—We combine a method for designing composite image filters with interactive genetic programming (IGP). Human subjective tests are used to comparitively evaluate the IGP-based filter design method to a manual filter design method and the multi-stage filtering feature of a software photo-retouching program. The composite image filter has a tree structure, with its nodes consisting of multiple simple image filters, arithmetic operators, arithmetic functions, constant values, and the pixel value of the input image. Genetic programming (GP) optimizes the tree structure based on the visual inspection of the IGP users, i.e. filter designers. Ten filter designers design composite filters using three methods: an IGP-based design method, a manual-based design method, and using the photo-retouching features of a commercial software program to time-sequentially apply ready-made filters. The designers make filters that output images corresponding to the given design concepts - *relaxed* and *violent* - based on their visual inspection. Twenty subjects compare the obtained images in pairs and evaluate which image is closer to achieving the give design concept. Wilcoxon signed-rank test demonstrates that the IGP-base filter design method can produce filters that create images with impressions that are closer to the given design concept than the other two methods.**

***Keywords*-interactive genetic programming, image processing, image filter design**

I. Introduction

Designing filters manually for each and every task is time consuming and costly. It is not practical for those who are not experts in signal processing to design filters by hand; take the example of medical image diagnostics experts who are users of medical image processing but are likely not signal processing experts. To bridge this gap in experience, the automatic design of filters by defining objective functions for measuring filter performances and applying several optimization methods has been attempted.

Designing filters using evolutionary computation (EC) is one such approach. Even if we limit the topic of this paper to image filters, there is much existent research, such as: mixed constrained image filter design using configurable logic blocks and genetic algorithms (GA) or particle swarm optimization (PSO) [3], [4], [5], [6], robust noise-specific image filter design using Cartesian genetic programming (Cartesian GP) [19], image preprocessing for volume rendering optimization using GA [20], fitness landscape analysis and image filter evolution using Cartesian GP [16], design and implementation of multiplierless two-dimensional image filters using GA [11], and others.

When filter performance cannot be expressed quantitatively, interactive EC (IEC) has been applied to not only acoustic signal processing, where it has been used for such tasks as designing filters for recovering speech from distorted speech or for hearing-aid fitting, but also for designing image filters [17]. Some IEC-based image filtering tasks, directly related with the work in this paper, include interactive GP (IGP)-based design of filters for coloring MRI images and for enhancing the differences between two ultrasonic images [15], determination of filtering order of multiple image filters using simulated breeding [12], [13], designing non-linear input-output characteristics of image density for image enhancement and colorization [7], [8], image enhancement by adjusting gamma correction [18], impulsive noise reduction from color images [9], [10] and its application to beautifying human face images [2], and others.

The objectives of this paper are (1) to show that composite image filters can be designed without giving target images by combining ACTIT [1], which is a design method for composite image filters, with IGP and (2) to show that this design method exhibits higher performance than manual-based filter design methods or the cascade combination of image filters used by commercial photo-retouch software. The advantages of ACTIT are that (1) filter performance can be increased by combining not only arithmetic operators, basic mathematical functions, and constants but also multiple elementary image filters as nodes in a tree structured filter created by GP and (2) by simply providing input images and target images, the GP auto-generates filters which produce outputs that are close to the given target images. Its disadvantage is that it does not work if no target image is provided. Our goal is to establish a method whereby image filters are designed automatically once a filter designer has a mental image of the impression they wish to create.

Following this section, we introduce our composite image filter design system consisting of ACTIT and IGP in section II, compare the performance of the proposed method with that of a manual filter design method and the cascade of ready-made filters using Adobe Photoshop and evaluate the proposed method through human subjective tests in section III.

II. IGP-based Composite Filter Design

Our proposed system mainly consists of a filtering part, where images are processed using the evolved composite filters, and a GP part for evolving the composited filters as shown in Figure 1. Processed images are displayed to an IGP user, and the GP part evolves the tree structure of the composite filter based on his/her subjective evaluations.

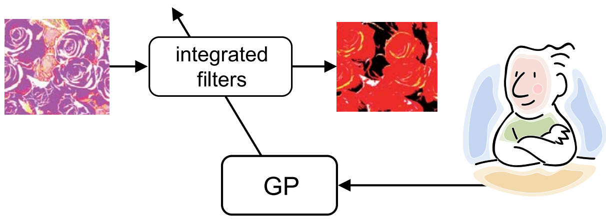

Figure 1. IGP-based composite image filter design system.

The composite filter in the filtering stage has a tree structure. Either of a prepared basic filter, an arithmetic operator, or an arithmetic function is assigned to its non-terminated nodes; either a numerical constant or a pixel value from the input image to the filter is assigned to its terminated nodes. We use 10 basic image filters in our experiments in this paper and describe them in detail in section III-B. Image files used in our experiments are 24 bits RGB color bitmap files.

Figure 2 shows a sample filter structure. X and F_i in the figure refer to the pixel values of the input image and the previously prepared i-th elementary filters, respectively. The number of nodes for the X is randomly initialized and depends on the generated tree structure. As Figure 2 has two X's, this filter inputs the same pixel values twice. In this case, the output value of the filter becomes $F_5(X+27, F_3(X))$, and the output value is assigned to the same pixel coordinate of the output image. Tree structured composite filters can handle more complex tasks than filters formed by simply cascading elementary filters. It is easy for us to understand this point because a cascade is a special form of the tree structure.

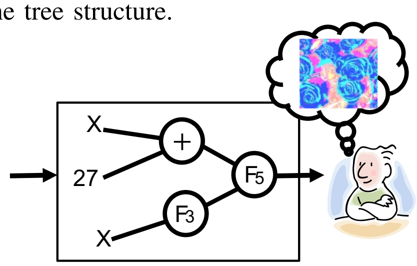

Figure 2. Example of a composite filter. F_3, F_5 and X refer to the third and fifth elementary filters and the pixel values of the input image, respectively.

The tree structured composite filters using in our experiments use numerical constants in the range of [1,100] and the pixel values, X, are presented from the terminated notes whereas arithmetic operators, arithmetic functions, and the 10 elementary filters are placed at the non-terminated nodes (see Table I). Filter performance is adjusted by changing type of the elementary filters, operators, and constants assigned to nodes and their numbers.

Table I
Nodes of the tree structured filters used in our experiments.

terminated nodes	non-terminated nodes
constants: 1 – 100 pixel value: X	elementary filters: F_1 – F_{10} unary operators: sin, cos, tan, log binary operators: $+$, $-$, $*$, $/$

Although maintaining diversity of tree structures is important, the performance of tree structures that are too small would be limited and those that are too big would require excessive filtering time. To overcome this problem in this paper, we try to keep the generated tree structures at a moderate size level by giving generation probabilities to terminated nodes and non-terminated nodes as shown in in Table II. The experimenter can control the sizes of the generated composite filters by adjusting these probabilities. Furthermore, we control the types of tree structures by giving generation probabilities to operators assigned to nodes. A maximum node level (or tree depth) is set to avoid generating huge trees (see Table IV), and terminated nodes of either a constant value or an input pixel value, X, are forcibly generated when a node level reaches the maximum.

Table II
Generation probabilities given to node types and each node parameter.

terminated nodes: 0.3	non-terminated nodes: 0.7
constants: 0.7 pixcel value X: 0.3	basic filters:0.5 unary operators: 0.2 binary operators: 0.3

III. Experimental Evaluation

A. Overview

We compare an IGP-based composite filter design method with a conventional manual-based filter deign method and a preset filter combination from a commercial photo-retouching program to evaluate the performance of our proposed method.

For comparison with the manually designed filter method, the designer of that filter is requested to make a tree structured composite filter of the same form as in the IGP-based composite filter based method. The designer is required to enter inputs in Polish notation and develop the tree structure by trial-and-error. For example, he/she enters $F_5 + X 27 F_3 X$ via a keyboard to make a filter shown in Figure 2.

For comparison with photo-retouch software, we use Adobe Photoshop. Adobe Photoshop includes dozens of image filters, such as a watercolor painting filter or an oil painting filter. Since these multiple filters can be applied sequentially to a single input image, the number of its total combinations, i.e. the total number of filtering effects, is arbitrarily many. Furthermore, the result from filtering with A, B, and C is different than that which would be achieved with the same filters in the order C, B, and A. Although filters prepared in Photoshop are powerful and highly functional, it is not easy for ordinary users to find the best sequential combinations of these filters.

In the first experiment, we confirm that IGP converges and find a composite filter which outputs an impression close to the target. As it is difficult to evaluate this convergence using a human IGP user, we use a pseudo IGP user made by a Gaussian mixture model that is a multimodal function with a big valley structure. For the parameters of the Gaussian mixture model, we use those described in reference [14]. Although a target image is given for this IGP simulation, we simulate an IGP user's evaluation character by a given score value in five levels calculated from the difference between the image output by the filter and the target image.

In the second experiment, we compare the composite filter design based by IGP with that based on a manual design. Ten filter designers make their best filters using two design methods, and 20 human subjects evaluate which processed image is closer to the given design concept. The given design concepts are *relaxed* and *violent*. The experimental conditions are listed in Table III.

In the third experiment, we compare a composite filter design based by IGP with a cascade of Photoshop filters. Ten filter designers make their best filters using the two design methods, and 20 human subjects evaluate which processed image is closer to the given design concept, as they did in the second experiment. The given design concepts are also the same as in the second experiment.

Table III
EXPERIMENTAL CONDITIONS FOR HUMAN SUBJECTS AND DESIGN CONCEPTS.

the # of designers designing composite filters	10
the # of subjects evaluating in the human subjective tests	20
two design concepts	*relaxed* *violent*

B. Basic Image Filters

Ten elementary image filters are used to construct the composite filters in our experiments. Let M be $2^8 - 1$ and p or $p(x, y)$ be a pixel value at the coordinate (x, y) of an input image.

A histogram extension filter, two gamma correction filters, and two sigmoidal transformation filters are shown by Eq.(1), Eq.(2) with γ = 1.5 and 0.3, and Eq.(3) with a = 1 and 10, respectively. A sharpening filter is given by Eq.(4) with mask(x, y) = $(0, -1, 0, -1, 5, -1, 0, -1, 0)$, where mask$(x, y)$ = $(a, b, c, d, e, f, g, h, i)$ means a weighting matrix of Figure 3.

$$f(p) = M\frac{p - min}{max - min} \tag{1}$$

$$f(p) = M\left(\frac{p}{M}\right)^{\gamma} \tag{2}$$

$$f(p) = \frac{M}{1 + \exp\left(a\left(p - \frac{M}{2}\right)\right)} \tag{3}$$

$$f(p) = \sum_{x=-1}^{1}\sum_{y=-1}^{1} p(x, y) \times \mathrm{mask}(x, y), \tag{4}$$

where *max* and *min* are the maximum and minimum pixel values among all the pixels in an input image.
Edge detection filter is given by Eq.(5).

$$f(p) = \sqrt{\frac{1}{2}\left(I_x^2 + I_y^2\right)}, \tag{5}$$

where $I_x = p(x-1, y) - p(x+1, y)$ and $I_y = p(x, y-1) - p(x, y+1)$.
The Sobel operator filter is given by Eq.(6).

$$f_{Sobel}(p) = \frac{1}{8}\sqrt{I_h^2 + I_v^2} \tag{6}$$

$$I_h = \sum_{x=-1}^{1}\sum_{y=-1}^{1} p(x, y) \times \mathrm{mask}_h(x, y)$$

$$I_v = \sum_{x=-1}^{1}\sum_{y=-1}^{1} p(x, y) \times \mathrm{mask}_v(x, y),$$

where $\mathrm{mask}_h(x, y) = (-1, 0, 1, -2, 0, 2, -1, 0, 1)$ and $\mathrm{mask}_v(x, y) = (-1, -2, -1, 0, 0, 0, 1, 2, 1)$.
An edge-preserving smoothing filter using a Sobel operator is expressed using a weighting matrix, Mask(x, y), of 5×5.

$$f(p) = \sum_{x=-2}^{2}\sum_{y=-2}^{2} p(x, y) \times \mathrm{Mask}(x, y), \tag{7}$$

where all elements of Mask(x, y) are $f_{Sobel}(p)$ given by Eq.(6) except $\mathrm{Mask}(0, 0) = M$.

a	b	c
d	e	f
g	h	i

Figure 3. Mask specified 3 horizontal pixels $\times$ 3 vertical pixels. The mask(x, y) is used to weight input pixels and calculate weighted sum as an output.

C. Experiment 1: Convergence experiment with a pseudo-IGP user

We use 24 bit color images of 511×456 pixels. The IGP experimental conditions are shown in Table IV. Figure 4 shows convergence curves for 30 generations with 10 individuals. One of the differences between IGP simulation with a pseudo-IGP user and GP search with a fitness function is the selection of elite individuals. Since a five-scale rating is adopted to simulate an IGP user, there may be multiple individuals with the same fitness value. Elite in this case is chosen randomly among the individuals with the same fitness values. The selection is thus influenced by the quantization noise inherent in the rounded fitness values.

Table IV
EXPERIMENTAL CONDITIONS OF GP.

population size	10
total # of generations	30
crossover rate	80%
mutation rate	1%
elite	the two best individuals
selection	roulette wheel selection
max. node level of a tree structure	20

Experimental results are shown in Figure 4. Individuals with the same fitness in a five-grade system are selected with the same probability during a selection operation; the same is true of elite. Even though all individuals are different, rounding errors in fitness cannot be avoided due to the five-grade system simulating a human IGP user's evaluation, which results in non-monotonic convergence curve despite the use of an elite strategy. However, we can see that the proposed IGP-based image filter design method converges from a broad view of the convergence curve.

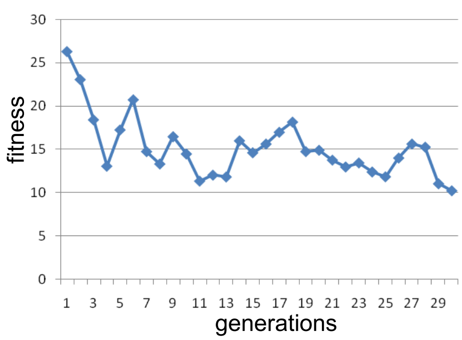

Figure 4. Convergence result with a pseudo-IGP user.

D. Experiment 2: IGP-based composite filter design vs. manual design

1) IGP-based filter design: Ten filter designers design composite filters generating images that match the two design concepts of *relaxed* and *violent* for 30 minutes per each design concept. The designers are undergraduate students and graduate students and have little experience of designing filters. The experimental conditions are the same as those in Table IV.

Each IGP user evaluates each of 10 displayed images on a 5-grade scale every generation. To maintain selection pressure, in any given generation the user must give the worst image or images a rating of one and the best image or images a score of 5. The images given the best score, 5 points, are saved automatically. After designing filters for 30 minutes, all saved images with the best score are displayed to the IGP user, and he/she selects the most corresponding *relaxed* and *violent* images.

2) Manual filter design: Filter designers are requested to manually create two filters for the two given design concepts of *relaxed* and *violent*, as was done in the IGP-based composite filter design in the previous section, for 30 minutes for each concept. The kinds of nodes used for composite filters are the same as the experiment in the previous section and shown in Table I.

Following an introduction to the use of Polish notation for designing the tree structure of the composite filter (see section III-A), filter designers enter the notation for their filter via a keyboard and observe an output image processed by the manually designed filter; this task is repeated for 30 minutes. All designed filters are saved automatically. After 30 minutes, each of filter designers choose the filter that generated an image which best matched the design concept from among all the saved filters. An outline of the process is shown below.

1) Filter designs are instructed what the elementary filter functions are and how to express a tree-structured filter and enter it using the keyboard.
2) The input image is loaded and filtering begins.
3) A tree-structured filter is entered using the keyboard.
4) The filtered image is evaluated.
5) Items 2, 3 and 4 are repeated for 30 minutes.
6) The best *relaxed* and *violent* image among the saved images are chosen according to the given design concept.

3) Human subjective tests: Following the experiments of sections III-D1 and III-D2 where 10 filter designers designed the best image filters for meeting two design concepts, 20 university students compared the filters chosen as the best in those sections with a five-grade system. These comparisons are statistically tested to determine if there are significant differences.

Table V shows the results of Wilcoxon signed-rank tests. They show that the IGP-based filter design method is significantly better than the manual method for creating *relaxed* and *violent* filters for 8 among 10 designers and all 10 designers, respectively ($p < 0.05$). There was no case where the manual-based filter design method was significantly better than IGP-based one.

E. Experiment 3: IGP-based composite filter design vs. cascade combination of Photoshop filters

1) Experiment of connecting Photoshop filters sequentially: The same 10 filter designers who participated in the experiment 2 in section III-D are requested to process

Table V
WILCOXON SIGNED-RANK TEST RESULTS OF EXPERIMENTS 2 AND 3 ($p < 0.05$). IGP, MANUAL, AND PHOTOSHOP MEAN IGP-BASED COMPOSITE FILTER DESIGN METHOD, MANUAL FILTER DESIGN METHOD, AND THE METHOD OF CASCADING PHOTOSHOP FILTERS, RESPECTIVELY.

experiment #	design concepts	results
experiment 2	*relaxed*	8 among 10 designers are "IGP > Manual"
	violent	10 among 10 designers are "IGP > Manual"
experiment 3	*relaxed*	7 among 10 designers are "IGP > Photoshop"
	violent	6 among 10 designers are "IGP > Photoshop"

the same input image and make images that meet the same design concepts of *relaxed* and *violent* using Adobe Photoshop CS5.1. Although it is possible for Photoshop to realize tree-structured filtering used in the experiment 2 by saving filtered images temporally and combining them later, only cascade filtering, i.e. applying filters A, B, C, and others time-sequentially, is used in our experiment because it is the most common use of Photoshop, especially for non-experts. Concretely speaking, 10 filter designers choose filters prepared in Photoshop (see these example filters in Figure 5) and apply them to the input image one after another until they obtain processed images of the given design concept up to 30 minutes.

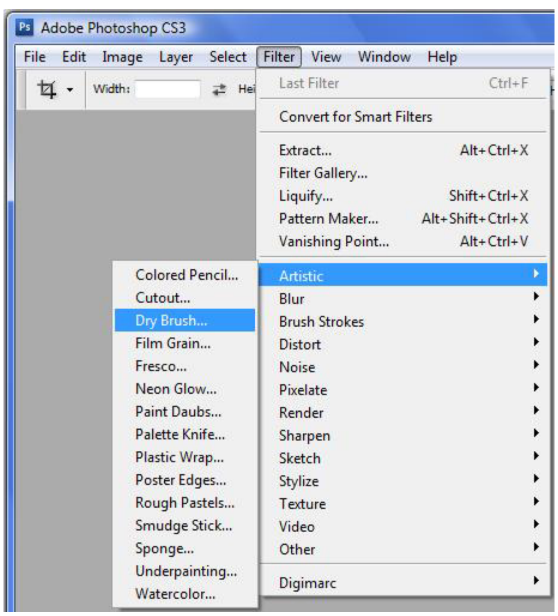

Figure 5. Examples of Photoshop filters.

2) Human subjective tests: Following the Photoshop experiment, 20 university students compare the filters chosen as the best in sections III-D1 and III-E1 with a five-grade system. These comparisons are statistically tested to determine whether there are significant differences.

Table V shows the results of Wilcoxon signed-rank tests. These results show that the IGP-based filter design method is significantly better than the cascade of Photoshop filters at creating a filter that conveys *relaxed* or *violent* for 7 among 10 designers and 6 among 10 designers, respectively ($p < 0.05$). There was no case where the filter designed using Photoshop was significantly better than the IGP-based one.

IV. DISCUSSIONS

The results of experiments 2 and 3 are summarized in Table V. Basically speaking, we can say that the proposed IGP-based composite filter design method can be a powerful image processing approach when we do not have concrete target images.

Since each of the filter parameters (genotype) and parts of a filtered image (phenotype) do not have a one-to-one correspondence but rather a more complex one, it is not easy for human users to change a filter structure or exchange node operators by observing a filtered image, and therefore, they have to design the filter based on trial-and-error. From this point of view, our experimental result, i.e. the superiority of the IGP-based filter design method to the manual design method, seems reasonable in general for tasks where the target image cannot be provided, as in the design task in this paper.

Because the component filters used by the IGP-based filtering and the cascade of Photoshops are different, we cannot say that the same experimental results would necessarily be reproduced with other comparative experiments; we cannot deny the possibility that the photo-retouch filters prepared in Photoshop may be suitable for certain design concepts. However, because the cascaded connection of elementary filters, which we used with Photoshop in this experiment, is a special case of a tree structured composite filter when the same elementary filters, arithmetic operators and arithmetic functions are prepared for tree nodes, we think it is reasonable to think that the proposed method should show better results than photo retouching software if the same experimental node conditions are given.

Here, we describe three points that could be improved to make our method more practical and effective at image processing in our future works.

The first point is to increase the number of elementary filters, arithmetic operators, and functions used for tree nodes and thus increase the expressive capability of the composite filter. We used 118 kinds of node parameters for the tree structured composite filters in our experimental system in this paper; see Table I. There were, however, only 18 excepting 100 constants. We should increase the kinds of nodes, especially the number of elementary image filters, in order to increase the generality and the capability of the system to handle more complex tasks.

The second point is to start searching after narrowing the search area rather than starting from randomly initialized individuals, which is common to all optimization methods. If we can narrow the search areas by using a priori or other knowledge, it could reduce IGP user fatigue.

The third point is to improve the user interfaces. User interface specifications have a deep influence on user fatigue. We did not use a graphical user interface (GUI), but rather Windows command line programs in our experiments. If a GUI had been used in the experimental evaluations, the effect of our proposed method would have been bigger, especially in experiment 3, because user fatigue in both IGP-based and manual composite filter design must be reduced.

V. Conclusions

We proposed to apply IGP as the design method for a tree structured composite image filter in which non-terminated nodes include not only arithmetic operators and arithmetic functions but also basic image filters. This combination provides us with a framework for designing image filters without requiring concrete target images for the filter. We experimentally demonstrated that we can make processed images closer to the given design concept by using the composite image filters made by this method than by those made with a manual designed composite filter or time-sequentially connected filters made using photo-retouching software. We discussed these experimental results and concluded that they have generality.

Acknowledgement

This work was supported in part by Grant-in-Aid for Scientific Research (23500279).

References

[1] Aoki, S. and Nagao, T., "Automatic Construction of Tree-Structural Image Transformations Using Genetic Programming," 10th Int. Conf. on Image Analysis and Processing (ICIAP'99), Venice, Italy, pp.136–141 (Sept., 1999).

[2] Arakawa, K. and Nomoto, K., "A Nonlinear Image Processing System for Beautifying Face Image Using Interactive Evolutionary Computing and Its Subjective Test," IEEJ Trans. on Electronics, Information and Systems, vol.131, no.3, pp.576–583 (2011) (in Japanese).

[3] Bao, Z., Watanabe, T., "Evolutionary design for image filter using GA," IEEE Region 10 Annual Int. Conf. (TENCON2009), Singapore, pp.1–6 (Jan., 2009).

[4] Bao, Z., Wang, F., Zhao, X., and Watanabe, T., "Fault-tolerant image filter design using GA," IEEE Region 10 Annual Int. Conf. (TENCON2010), Fukuoka, Japan, pp.897–902 (Nov., 2010).

[5] Bao, Z., Watanabe, T., "Mixed constrained image filter design using particle swarm optimization," Artificial Life and Robotics, vol.15, no.3, pp. 363–368 (2010).

[6] Bao, Z., Wang, F., Zhao, X., and Watanabe, T., "Mixed constrained image filter design for salt-and-pepper noise reduction using Genetic Algorithm," IEEJ Trans. on Electronics, Information and Systems, vol.131, no.3, pp. 84–591 (2011).

[7] Jakša, R., Nakano, S., and Takagi, H., "Image Filter Design with Interactive Evolutionary Computation," Int. Conf. on Computational Cybernetics (ICCC2003), Siófok, Hungary, pp.1–6 (Aug., 2003).

[8] Jakša, R. and Takagi, H. "Tuning of Image Parameters by Interactive Evolutionary Computation," IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC2003), Washington D.C., pp.492–497 (Oct., 2003).

[9] Katsuyama, Y. and Arakawa, K., "Impulsive Noise Removal in Color Image Using Interactive Evolutionary Computing," IEICE trans. on fundamentals of electronics, communications and computer sciences, vol.93, no.11, pp.2184–2192 (2010).

[10] Katsuyama, Y. and Arakawa, K., "Color image interpolation for impulsive noise removal using interactive evolutionary computing," Int. Symposium on Communications and Information Technologies (ISCIT), Tokyo, Japan, pp. 877–882 (Oct., 2010).

[11] Lockett, D.J., Roblee, C.D., and Rudko, M., "Genetic algorithm based design and implementation of multiplierless two-dimensional image filters," Intelligent Engineering Systems Through Artificial Neural Networks, vol.13, St. Louis, USA, pp.333–339 (Nov., 2003).

[12] Mutoh, T., Komagata, N., and Ueda, K., "An experimental study for automatically generating image filter sequence by using simulated breeding," Workshop on Interactive Evolutionary Computation, Fukuoka, Japan, pp. 7–12, (Mar. 1998) (in Japanese).

[13] Mutoh, T., Nakayama, M., Ueda, K. and Kitamura, S., "A study for making image filter sequence using interactive Evolutionary Computation," Human Interface Symposium, Tsukuba, Japan, pp.261–264 (Aug., 2000) (in Japanese).

[14] Nakano, Y. and Takagi, H., "Influence of Quantization Noise in Fitness on the Performance of Interactive PSO," IEEE Congress on Evolutionary Computation (CEC2009), Trondheim, Norway, pp.2146-2422 (2009).

[15] Poli, R. and Cagnoni,S., "Genetic programming with user-driven selection: Experiments on the evolution of algorithms for image enhancement" 2nd Annual Conf. on Genetic Programming, pp.269–277 (1997).

[16] Slaný, K. and Sekanina, L., "Fitness landscape analysis and image filter evolution using functional-level CGP," 10th European Conference on Genetic Programming (EuroGP2007), Valencia, Spain, pp.311–320 (April, 2007).

[17] Takagi, H., "Interctive evolutionary computation: Fusion of the capabilities of EC optimization and human evaluation," Proceedings of the IEEE, vol.89, no.9, pp.1275–1296 (2001).

[18] Tokuda, Y., Hashino, H., et al., "Image quality enhancement support system by gamma correction using interactive evolutionary computation," IEEE Int. Conf. on Systems, Man and Cybernetics (SMC2007), Montrial, Canada, pp.2906–2910 (2007).

[19] Vasicek, Z. and Bidlo, M., "Evolutionary design of robust noise-specific image filters," 2011 IEEE Congress of Evolutionary Computation (CEC2011), New Orleans, USA, pp.269–276 (June, 2011).

[20] Zheng, Z.-R., Liu, Z.-C., Liu, X.-Y., and Du, P., " Genetic algorithm-based image preprocessing for volume rendering optimization," 2009 IEEE Int. Symposium on IT in Medicine and Education (ITME2009), Jinan, China, pp.389–393 (Aug., 2009).