Interactive Differential Evolution for Image Enhancement Application in Smart Phone

Myeong-Chun Lee and Sung-Bae Cho
Dept. of Computer Science
Yonsei University
Seoul, Korea
lmspring@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Abstract— **In this paper, we propose an automatic image enhancement tool for smart phone by using interactive differential evolution (IDE). From a remarkable progress of the camera sensor in mobile devices, people take pictures with their mobile phone instead of a digital camera. However, as they are not satisfied with their images in spite of the progress, they still want to edit their images by using mobile applications, which are usually complex and cause user fatigue, especially for beginners. To reduce it and make a simple interface, we exploit IDE, which is a kind of interactive evolutionary computation. Let the user IDE to evaluate the individuals. Because of the small parameters of the differential evolution (DE), we could make the tool simply and overcome the user fatigue. DE is also an efficient and fast evolutionary algorithm which uses the difference of the vectors. Subjective test shows the usefulness of the tool.**

Keywords: Differential evolution, Interactive differential evolution, Mobile image enhancement

I. INTRODUCTION

Usually, people are not satisfied with their images, which are taken from a digital camera or camera sensor built-in mobile devices. Because of this reason, people utilize image editing tools to make a clear and bright image. Additionally, the mobile camera sensor is getting smaller and more sophisticated [1].

From the progress of camera sensor, many people take pictures by using their mobile phone instead of digital camera and they enhance their images through a mobile image editing tool. However, there are some drawbacks. First, it is mostly difficult to learn the mobile image editing tools for beginners who cannot handle the mobile phone. Second, although the mobile phone is inappropriate for displaying many images on its screen at once, the commercial tools are mostly separated into many parts for generating their enhanced image. It is inconvenient for some people who have poor sight.

To solve the problem, we propose a mobile color image enhancement tool which only shows two images in a display by using Interactive Differential Evolution (IDE). IDE is a kind of Interactive Evolutionary Computation (IEC) that user evaluates the individuals for new generation. IDE alleviates the user fatigue which is the problem of IEC. There is an also reasonable result that IDE has better performance than

Interactive Genetic Algorithm (IGA) and tournament IGA [2]. A main concept of the differential evolution (DE) is to utilize vector difference. Because DE is a simple search method and performance of convergence is fast, it is more suited for mobile environment than desktop environment.

The objective of this paper is to enhance the color of images from the mobile phone and show the usefulness of the proposed image enhancement tool. Color image enhancement is a method which produces an output image that subjectively looks better than the original image by changing the intensity of the image pixel. To do that, we encode the pixel values of images as a chromosome for changing it. We conduct the subjective tests by using the results of the enhanced images and compared with other methods such as IGA and PEstudio that are a commercial image editing tool.

This paper is organized as follows. We explain the related methods which are developed in mobile environment in Section II. In Sections III and IV, we introduce an image enhancement tool, DE, and IDE. Section V describes the results of the experiment, and our conclusions are presented in Section VI.

II. RELATED WORKS

There are many methods to manipulate the images in the mobile environment. Color enhancement, cropping image, and noise reduction are the best-known methods. TABLE I shows some studies about image enhancement in mobile environment.

TABLE I.
Mobile image enhancement related study

Author	Year	Editing type	Method
Z. Jian et al.[3]	2005	Image merging	Cooperative editing
T.–M. Jung et al.[4]	2010	Image refinement	Interactive Evolutionary Computation
J.-H. Jung et al.[5]	2010	Skin tone and background change	Segmentation, boundary refinement, etc.
K. R. Babu et al.[6]	2010	Color enhancement, noise reduction	Mean and variance Adjustment
X. Yingen et al.[7]	2010	Image stitching	Seam finding, image blending

The selection of algorithms and development environment are important factors. Although the performance of mobile devices has increased remarkably, it is still lower than the desktop. Not only simple, but also good performance algorithm is needed to be used for image enhancement in a mobile environment. Most of those studies in TABLE II tried to enhance the color tone or reduce the noise. However, these are complex and need many parameters. T.-M Jung et al. separated a display into nine images to compute the evolution and most commercial tools are also utilizing the complex interface. It is inconvenient to check the detailed parts of the image because of the small screen of mobile phone.

We utilize DE to overcome the above problems. Because the number of parameters of DE is small, the performance of consistent convergence to the global minimum is good, and it is possible to display just two images on a screen of the mobile phone.

III. IMAGE ENHANCEMENT TOOL

Fig. 1 shows the flow chart of the image enhancement tool by using IDE. The initialization means the selection of an image from a user. Each element of the image, which contains color, brightness, contrast, and gamma values, is encoded as a chromosome. The new image is generated after mutation and crossover. User has two options as follows. First, they choose an image which he wants to generate between two images. A new image is generated again from the user's choice after mutation and crossover. The other is to choose a final image from the user.
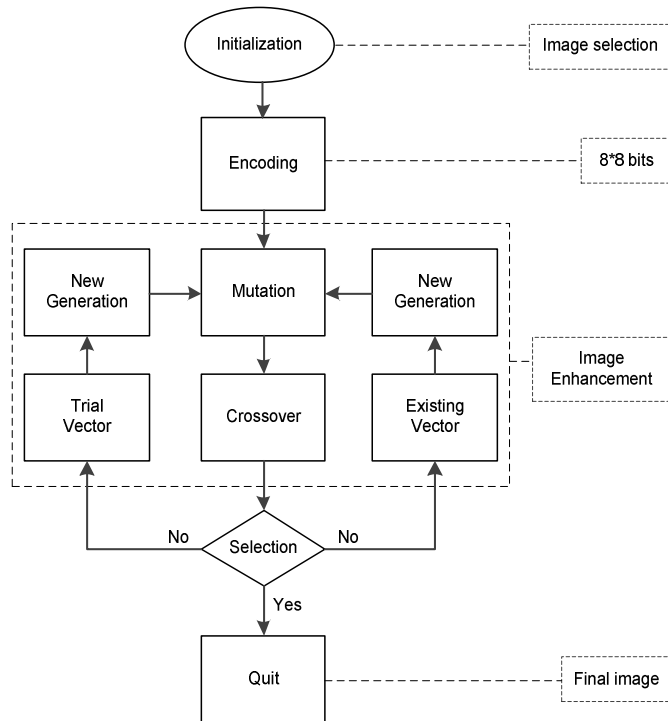

Fig. 1. Flow chart of the image enhancement tool.

As mentioned before, it is difficult to show many images on the screen in a mobile device. Due to this, we only show two images on a screen, like Fig. 2. There are two user options to evaluate the enhanced images and to choose a final image. User can choose what he wants to enhance between two images and if an image is chosen, the enhanced image will be shown to the other screen. Other menus, such as NEW, RESET, SAVE, and EXIT, are provided for interface.
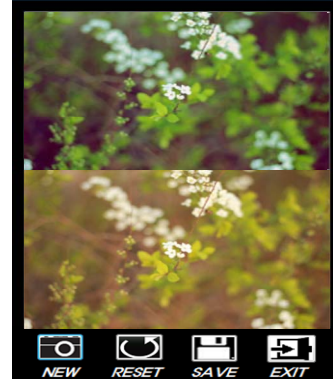

Fig. 2. Interface of the image enhancement tool. Two main options: choice of an image that is needed to enhance and choice of final image. Four buttons: new, reset, save, and exit.

IV. METHOD

A. Encoding

We present each element of the images as a chromosome and it is shown in Fig. 3. An image usually consists of a lot of pixels or vectors. Each pixel has the information such as brightness, contrast, gamma, and color. Brightness information is a subjective human feeling from the colors of an object. Contrast is the difference of the visual feature to distinguish between object and background. Gamma information is to change the intensity of light nonlinearly. Color information is used for changing color balance. Brightness and contrast are encoded in 8bits. Gamma and color are encoded in 24 bits and each value of red, green, and blue is individually divided into 8bits.
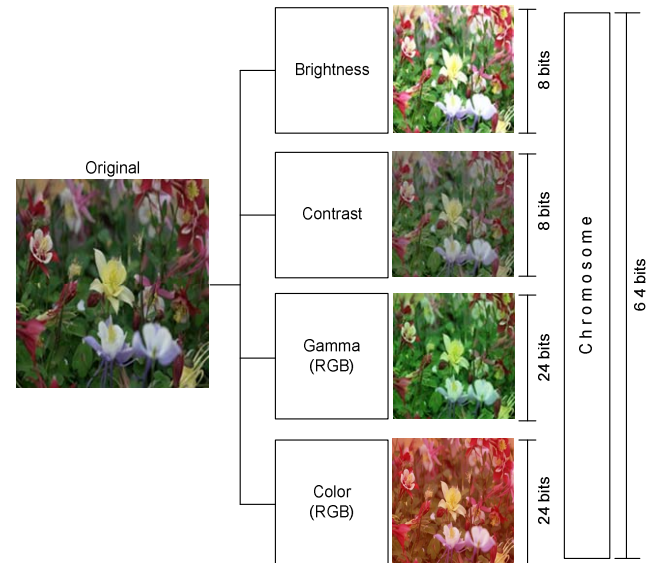

Fig. 3. A chromosome encoded by each element of an image.

B. Differential Evolution

Differential Evolution (DE) is a parallel direct search method which was introduced by Storn and Price in 1996. It has four major characteristics: First, DE was designed to be a stochastic direct search method to handle non-differentiable, nonlinear cost function. Second, DE utilizes the vector populations for parallelism. Third, self-organization is used as a minimization method. Fourth, performance evaluated through convergence property is good compared with algorithms such as Annealed Nelder and Mead strategy (ANM) and Adaptive Simulated Annealing (ASA) [8]. Since the advent of DE, many variants of the basic algorithm such as Trigonometric mutation, Arithmetic recombination, Opposition-Based DE, etc. are developed [9]. However, in this paper, we apply the original version. DE represents all individuals as NP D dimensional real-valued parameter vectors

$$x_{i,G}, i = 1,2,3, \ldots, NP \qquad (1)$$

where i is individuals and G is generations. Trial vector $m_{i,G}$ is generated by mutation and crossover operations. To calculate a mutant vector, three individual members are randomly chosen, and the difference of two vectors is multiplied by mutation control constant. The output is added to the other individual to generate the mutant vector. The calculation for a mutant operation is indicated as follows.

$$m_{i,G} = x_{r1,G} + F \cdot (x_{r2,G} - x_{r3,G}) \qquad (2)$$

where $x_{r1,G}$, $x_{r2,G}$, $x_{r3,G}$ are the randomly generated individuals and mutually different. $m_{i,G}$ is the mutant vector in generation G. F is the mutation control constant to suppress the amplification. In this work, each individual makes a filter. Fig. 4 shows how a mutant filter is generated. Three filters are randomly chosen and the difference of two filters is calculated. Constant value F is then applied to limit the amplification. After applied F value, Filter 1 is added for making a mutant filter.
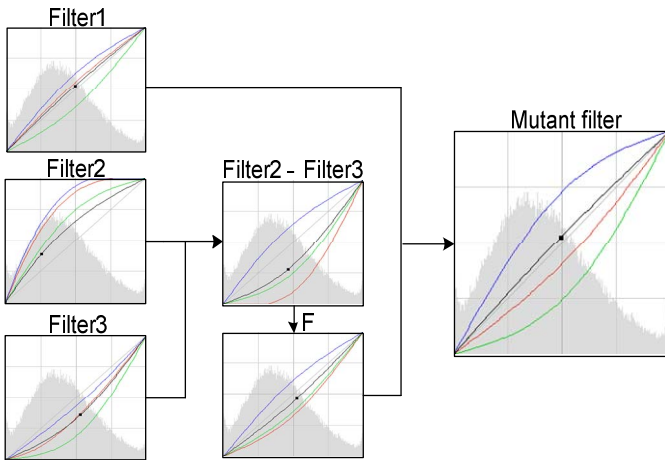

Fig. 4. Calculation of a mutant filter. F is constant value to limit the amplification of the difference between filter2 and filter3.

After the mutant filter is generated, crossover operation is conducted. Equation (3) is a calculation that makes a trial vector from the target and mutant vectors. If *randb(j)* which means the evaluation of a uniform random number generator with outcome is lower than crossover rate (CR) or *j* is equal to *rnbr(i)* which means randomly chosen index, target vector is represented as a trial vector. Otherwise, mutant vector is represented as a trial vector.

$$u_{ji,G} = \begin{cases} m_{ji,G} & \text{if}(randb(j) \le CR) \text{ or } j = rnbr(i) \\ x_{ji,G} & \text{if}(randb(j) > CR) \text{ or } j \ne rnbr(i) \end{cases}$$
$$when \ j = 1, 2, \ldots, n \qquad (3)$$

An example of crossover operation is shown in Fig. 5. Trial filter is determined according to the value of *randb* and CR rate.
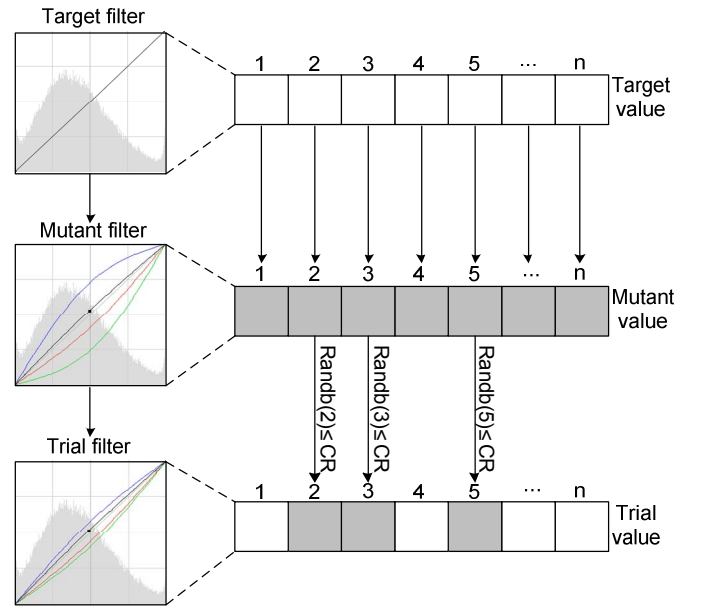

Fig. 5. Crossover process. A trial filter is generated from the target filter and mutant filter.

After the crossover operation, a selection is performed as equation (4). The generated trial vector has to be evaluated whether it becomes a new individual of next generation through a cost function. If output of the cost function applied through the trial vector is lower than that of the other cost function applied through the target vector, trial vector is selected. Otherwise, target vector is selected.

$$x_{i,G+1} = \begin{cases} u_{i,G+1} & if \ f(u_{i,G}) \le f(x_{i,G}) \\ x_{i,G} & Otherwise \end{cases} \qquad (4)$$

However, we give the right of choice what filter will choose to users, as shown in Fig. 6. This method will be explained in next section.
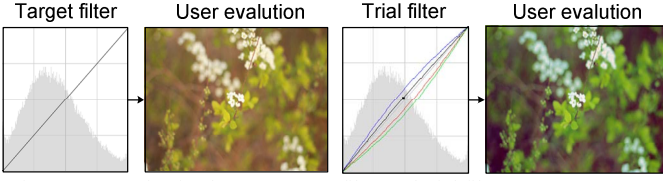
Fig. 6. User evaluation between target filter and trial filter.

C. Interactive Differential Evolution

IDE is a kind of IEC method in which the user evaluates the fitness function and is based on differential evolution. Most of the previous studies about IDE [10][11][12] tried to overcome user fatigue, which is usually occurred on IGA. New individual between trial and target vectors is selected from the user. Fig. 7 shows a pseudo code of DE. To materialize as two options, we substitute the condition statements to user evaluation. From these alterations, user can choose what they will make an enhanced image between original and enhanced images. The stop condition is also determined from user choice for a final image.

```
// X : Target vector
// M : Mutant vector
// U : Trial vector
// G : Generation
// NP : Parameter vector

Initialization X(0) ← {X1(0),…,XNP(0)}
G ← 0
while stop_condition() do     // change to user selection
    for i=1 to NP
        Mi ← mutation(X(G))
        Ui ← crossover(Xi(G), Mi)
        if f(Ui) <f(Xi(G))        // change to user selection
            Xi(G+1) ← Ui
        else                      // change to user selection
            Xi(G+1) ← Xi(G)
        end if
    end for
    G ← G+1
end while
```

Fig. 7. Pseudo code of DE. The condition statements and loop statement are changed to user selection for making an enhanced image.

V. EXPERIMENT

We conducted five experiments and twenty people participated in the experiment to evaluate the user satisfaction and fatigue degree. Thirty four images were used for evaluation and comparison.

First, we used a landscape image to confirm how the image is generated in each generation. Fig. 8 shows the result of the test. The phenotype shows the generated image in each generation and genotype shows the value of each element in the image filter. Final image was generated in eleventh generation.
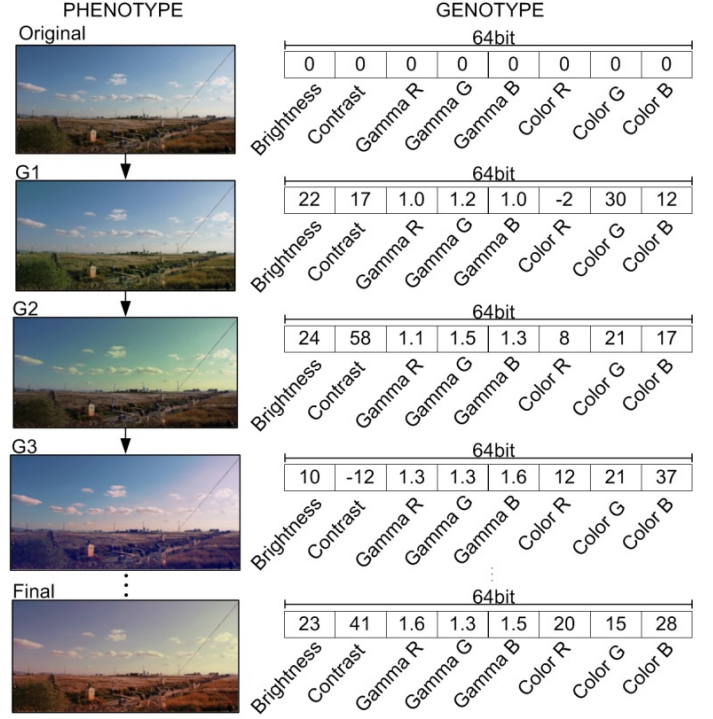

Fig. 8. Phenotype and Genotype result for a landscape image. Total elements of filter in genotype consist of 64bits.

We estimated the number of generations when users choose the final image. This experiment was conducted for measuring user fatigue. Ten images (landscape, object, and portrait) were used for the experiment. As can be seen in Fig. 9, most final images are chosen within ten generations except the image (b). This result confirms that the users did not suffer from fatigue and they can make enhanced images without difficulty.
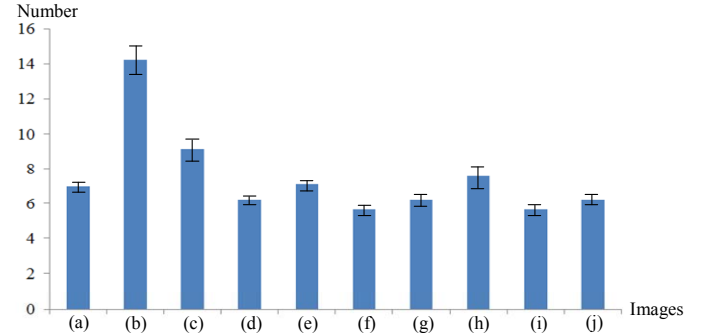

Fig. 9. Average number of generations until final image was generated. x axis: Images for experiment, y axis: Number of generations. (a)~(d): landscape, (e)~(h): Object, (i),(j): portrait.

Because most pixels in the image (b) as shown in Fig.10 were consist of unicolor, it was difficult to distinguish what color of the image was enhanced. It required more generations than the others. However, this phenomenon occurred in other tools and the expert tool such as Photoshop is usually needed to edit such images. Since the average number of generations is still under fifteen, it is the still sufficient number to satisfy the users.
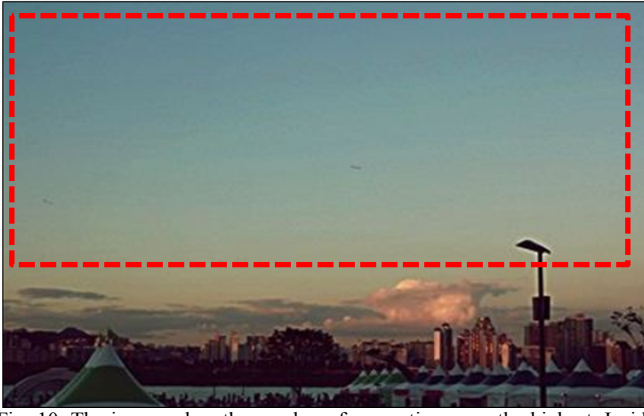
Fig. 10. The image when the number of generations was the highest. Inside of the dotted red line shows that most of pixels are consist of similar color.

We conducted additional experiment by using six images (flower, portrait, glass, lock, and landscape) and the enhanced images were compared with the originals and other two images generated by IGA and PEstudio. The PEstudio is a commercial application program in mobile environment [13]. The enhanced images are shown in Fig. 11 and the total number of images compared each other were twenty four.

(a)   (b)   (c)   (d)

Fig. 11. Five images are used for the experiment. Column (a): Original, Column (b): Results of IDE, Column (c): Results of IGA, Column (d): Results of PEstudio.

Twenty people were shown the enhanced and the original images for evaluation. The range of scores is from 0 to 100 and average score of each image is shown in Fig. 12.
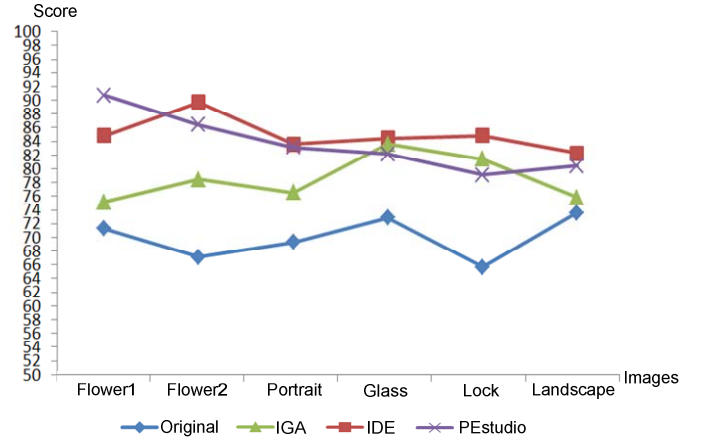

Fig. 12. The result of user satisfaction. Generally, IDE receives the highest score except Flower1 image.

As can be seen, three tools received higher scores than original, and especially IDE received the highest score of all images except Flower1. This is a meaningful result because IDE even received higher scores than the commercial tool (PE studio). However, biased colors periodically arose, as the right image on Fig. 13.
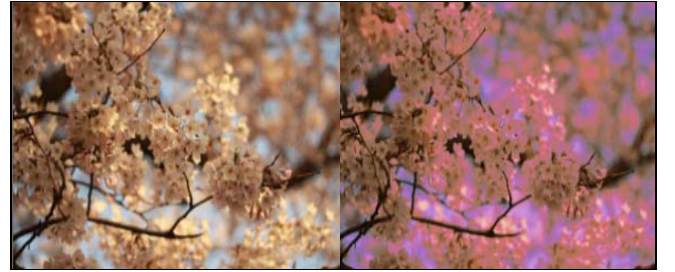

Fig. 13. Color bias phenomenon of Flower1.

To reduce this phenomenon, we adjusted the crossover rate. Fig. 14 shows the number of color bias phenomenon during 30 generations. Color biased number is gradually decreased as crossover rate was decreased. 0.3 is the best-fit crossover rate because there was an only little change for enhanced image when the crossover was 0.1.
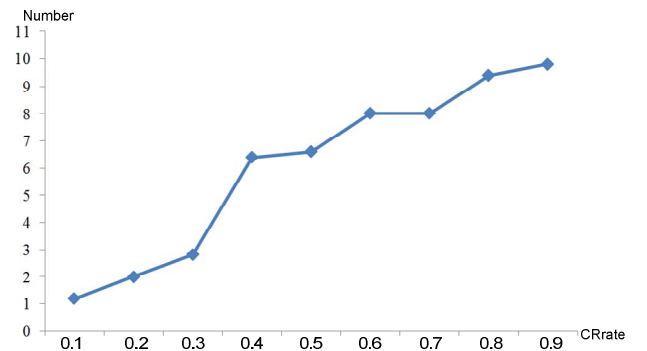

Fig. 14. The result of number of color bias phenomenon from crossover rate. x axis: crossover rate, y axis: The number of occurrences of color bias phenomenon.

We conducted SUS test to find merits and demerits of the tool. PEstudio and IGA are also conducted for the comparison. Twenty people were participated in the experiment and they indicate the degree of agreement or disagreement to the statement on a 5-point scale. The questionnaire is shown in TABLE II. It covers a variety of aspects of system usability, such as the need for support, training, and complexity, and thus have a high level of face validity for measuring usability of a system. Since even numbers of the questions are negative, the score of even number questions has to be recorded contrariwise[14].

TABLE II
System Usability Scale Questionnaire

#	Quesiton
Q1	I think that I would like to use this svstem frequently
Q2	I found the system unnecessarily complex
Q3	I thought the system was easy to use
Q4	I think that I would need the support of technical person to be able to use this system
Q5	I found the various function in this system were well integrated
Q6	I thought there was too much inconsistency in this system
Q7	I would imagine that most people would learn to use this system quickly
Q8	I found the system very cumbersome to use
Q9	I felt very confident using this system
Q10	I needed to learn a lot of things before I could get going out with this system

Fig. 15 is the result of SUS test and IDE was compared with IGA, and PEstudio. The score has a range of 0 to 100 and average score on each question is recorded. Except the questions 5 and 6, IDE was received the higher score than PEstudio. This result verifies the usefulness of IDE. The questions 5 and 6, those were about various features. It cannot help receiving the lower score than PEstudio since our goal of this paper was to make a tool simply.
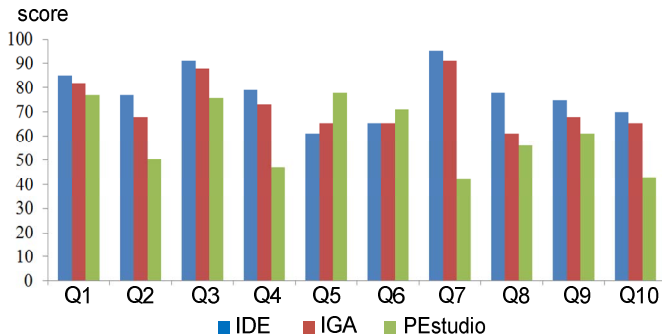

Fig. 15. Result of SUS test. IDE generally received the highest score except questions 5 and 6.

VI. CONCLUSION

We have proposed a color enhancement tool by using IDE and it has been developed in mobile environment. Because IDE uses a small number of parameters for generation, we showed only two images on a screen to make color enhancement images. Not only is it simple and convenient, but it also relieves the user fatigue. We have also proved the usefulness of the tool from the user satisfaction experiment and found the best-fit crossover rate to reduce color bias of the images.

REFERENCE

[1] R. Wei, "Motivations for using the mobile phone for mass communications and entertainment", *Telematics and Informatics*, vol. 25, no. 1, pp. 36-46. 2008.
[2] H. Takagi and D. Pallez, "Paired comparisons-based interactive differential evolution," *World Cong. on Nature & Biologically Inspired Computing*, pp. 475-480, 2009.
[3] Z. Jian, L. Qing, L. Xiang, and W. Liu, "A cooperative image editing tool over mobile phones," *Conf. on Multimedia Modeling*, pp. 264-270, 2005.
[4] T.-M. Jung, Y.-S. Lee, and S.-B. Cho, "Mobile interface for adaptive image refinement using interactive evolutionary computing," *IEEE Cong. on Evolutionary Computation,* pp. 1-7, 2010.
[5] J.-H. Lee, J. Heo, and Y.-S. Ho, "Skin tone enhancement and background change for mobile phone," *IEEE Trans. on. Consumer Electronics*, vol. 56, no. 4, pp. 2719-2726, 2010.
[6] K. R. Babu and K. V. N. Sunitha, "A new approach to enhance images of mobile phones with in-built digital cameras using mean and variance," *Int. Conf. on Advances in Computer Engineering* , pp.232-234, 2010.
[7] X. Yingen and K. Pulli, "Fast image stitching and editing for panorama painting on mobile phones," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 47-52, 2010.
[8] R. Storn and K. Price, "A simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization,* vol. 11, no. 4, pp. 341-359, 1997.
[9] S. Das and P. N. Suganthan, "Differential Evolution: A survey of the State-of-the-Art," *IEEE Trans. on Evolutionary Computation*, vol.15, no.1, pp. 4-31, 2011.
[10] M. Fukumoto, Y. Imamura, M. Inoue, and J. Imai, "Convergence of vectors in paired comparison-based interactive differential evolution for creating scent," *Int. Conf. on P2P, Parallel, Grid, Cloud and Internet Computing*, pp. 440-445, 2010.
[11] Z. S. Mohamad, A. Darvish, and S. Rahnamayan, "Eye illusion enhancement using interactive differential evolution," *IEEE Symposium on Differential Evolution*, pp. 1-7, 2011.
[12] R. Funaki and H. Takagi, "Application of gravity vectors and moving vectors for the acceleration of both differential evolution and interactive differential evolution," *Int. Conf. on Genetic and Evolutionary Computing (ICGEC)*, pp. 287-290, 2011.
[13] http://itunes.apple.com/us/app/photo-effect-studio/id440159265?mt=8
[14] J. Brooke. "SUS: a quick and dirty usability scale," *Usability Evaluation in Industry*, pp.189-194, 1996.