

## Tuning of image parameters by interactive evolutionary computation

Jaksa, Rudolf

Department of Cybernetics and AI Technical University of Kosic

高木, 英行

Department of Art and Information Design, Kyushu Institute of Design

<https://hdl.handle.net/2324/4486567>

---

出版情報 : Proceedings of 2003 IEEE International Conference on Systems, Man, and Cybernetics (SMC'03), pp.492-497, 2003-10-08. IEEE

バージョン :

権利関係 : © 2003 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.





# Tuning of Image Parameters by Interactive Evolutionary Computation\*

Rudolf Jakša

Department of Cybernetics and AI  
Technical University of Košice  
Letná 9/B, 04120 Košice, Slovakia  
jaks@neuron.tuke.sk

Hideyuki Takagi

Faculty of Design  
Kyushu University  
4-9-1 Shiobaru, Minami-ku, Fukuoka, 815-8540, Japan  
takagi@design.kyushu-u.ac.jp

**Abstract** – We discuss the use of interactive evolutionary computation for designing and optimizing image enhancement filters with fewer parameters than normal filter model. We present a method to speed up this optimization by combining interactive and non-interactive evaluation.

**Keywords:** Interactive evolutionary computation, image enhancement, visual evaluation.

## 1 Introduction

Interactive evolutionary computation (IEC) is a method used to optimize target systems based on human subjective evaluation and has been applied in several fields such as the arts, engineering and education [6]. Image processing is one of these applications, and some research regarding its efficacy has been presented [3, 4, 5, 7]. This paper is a consecutive work to our papers [1, 2, 7], which focused on the IEC-based optimization of algebraic and complex parametric filters for image enhancement.

The objective of this paper is to design image filters with few tunable parameters than normal filter modeling and study the performance of IEC for optimizing them. Adjustment of brightness, contrast or gamma modifiers present a common type of such a filter.<sup>1</sup> We use an approach that is coherent with our previous work [1, 2, 7] and analyze results according to the past results obtained with different types of filters.

Contrast and brightness modification of an image specifies the optimized input-output characteristic of a filter for each of the RGB colors. This optimization is performed in the IEC process, which is based on the users' visual inspection and accordingly interactively designed filter. It is a suitable approach for image processing where the users are not signal processing experts.

We also present a method for combining interactive and non-interactive evaluation in IEC-based im-

age processing. Non-interactive histogram-based pre-evaluation of filter candidates is incorporated into IEC with an aim to reduce user fatigue.

This paper is structured as follows. First we describe the generic approach of IEC-based image processing and image filters in sections 2 and 3. Next, we focus specifically on filters with few parameters in section 4.1 and discuss experimental results for the particular IEC-based image processing of section 4.2. A method for combining interactive and non-interactive evaluation in IEC-based image processing is introduced in section 4.3.

## 2 IEC-based image processing

IEC is a form of Evolutionary Computation (EC) where a human is used instead of a fitness function. A basic schematic of an IEC-based image processing application is illustrated in Figure 1.

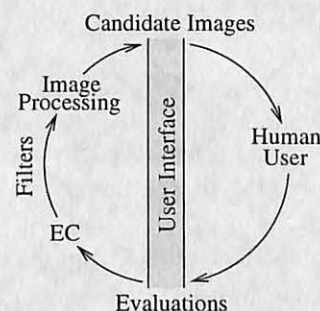


Figure 1: IEC-based image processing loop.

Interaction between a human user and a computer, indicated on the right and left sides of Figure 1 respectively, is conducted through a user interface. The user interface displays candidate images to the IEC user and collects the user evaluations of these images.

The computer part of IEC consists of an EC part and an image processing part. User-inputted evaluations are used by the EC to optimize individual image processing filters. The role of the image processing part is to generate the next generation of images using the image filters created by EC. This interactive optimization process loops iteratively.

\*0-7803-7952-7/03/\$17.00 © 2003 IEEE.

<sup>1</sup>We use term "image parameters" to refer to values of filters, where 'brightness' may be such a image parameter.

In the field of image processing, IEC is applicable in two partly different situations:

- a) if a user is looking for a particular filter or features, she/he can develop it or find it in this process, and
- b) if the user wants to find some new unknown features of a certain image – to explore the image – she/he can accomplish that through this process, as well.

The result in case a) is the last filter obtained in IEC session, while in case b) it is an interactive process as whole; case a) represents an exploitative behavior, while case b) the explorative one.

Genetic Algorithms (GA) and Genetic Programming (GP) with algebraic operations have been used for IEC-based filter design for parametric filters and for classical image processing filters. The tradeoff between using different variations on these methods and structures is not well studied yet. There are no established real-world applications of IEC-based image processing.

## 2.1 User interface

An important component of IEC is the Graphical User Interface (GUI), because the interaction between computer and user is an essential part of the IEC algorithm. This is an exception in EC area and also in wider context of computational intelligence.

In image processing applications, the most basic role of the GUI is to display sets of images to the user (see Figure 8). The impression of the GUI is dramatically different if it is used to display 10 images at once or to display 100 images at once. Because of the limited capacity of the computer display and the need to display several images in parallel, there are two trade-offs in design of GUI:

1. The trade-off between the ability to see details in the image and ability to see the overall appearance of image. This is balanced by usage of zooming and/or cropping of the image (displaying only a part of it).
2. The trade-off between the ability see details in the single image and ability to grasp its relation to other displayed images. If we reduce the size of population too much, the optimization performance may be limited. If we display only a portion of population at once, the ability of user to compare particular individuals may be limited, as she/he must recall non-displayed images.

Another role of GUI is to collect users' evaluations. This may be configured in several ways too. The user may search and pick out interesting images, select and evaluate best images, precisely evaluate all images, etc. Furthermore, there is a difference between continuous and discrete evaluation, as well as between 3-value and 5-value ones. If the user is looking for any interesting

features of the image and wants to see a lot of different views of the image, he/she will be frustrated by the need to precisely evaluate every single image. If the user knows the target filter behavior, she/he may want to precisely evaluate slightly different images in order to fine tune the resulting filter.

User interface handles two information flows: one from computer to human and the other from human to computer. The total bandwidth of the information flow through the user interface is then the sum of these two flows. We may assume that a higher bandwidth means a deeper interaction. The information exchanged in interactions may, however, differ dramatically in quality. If using more expressive image presentations, the user evaluations may be more accurate and may lead to faster convergence of the search. To measure quality of presented information, a sum of consequent future evaluations might be used.

## 3 Image Processing Filters

### 3.1 Algebraic Filters

Algebraic filters [7] compute the output pixel value using input pixel information and an arbitrary mathematical function to describe the input/output relation. To evolve such a filter the natural choice is to use GP methods, since they can build the structure of the function and also optimize its parameters. However, it's possible to build such a filter with predefined structure type using a simple GA for its optimization.

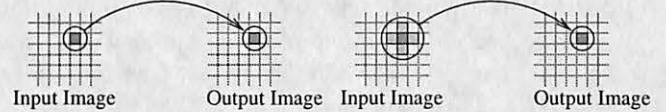


Figure 2: One-to-one and several-to-one pixel mapping filter.

The following mathematical function is an example of an algebraic filter as discussed:

$$r_{out} = 100 \sin(r_{in}/20) + 50 \log(g_{in}) - 120 + r_{in}/b_{in} \quad (1)$$

Here,  $r_{out}$  is the intensity of the output pixel in its red channel and  $r_{in}$ ,  $g_{in}$  and  $b_{in}$  are its input intensities in the red, green and blue channels. This filter realizes a one-to-one mapping of pixels in an image. To be able to achieve edge detection or other operations which rely on relationships between different pixels in an input image, we must include several input pixels in the filter function.

An algebraic filter function for implementing a several-to-one pixel mapping may take the following form:

$$r_{out} = 0.2r_{-1,0} + r_{0,0} + 0.1g_{0,-1} + 0.2g_{0,0} - 0.1g_{0,1} \quad (2)$$

Where  $r_{-1,0}$  is the red channel intensity of the pixel at the position  $(-1,0)$ , (left) of the current pixel.



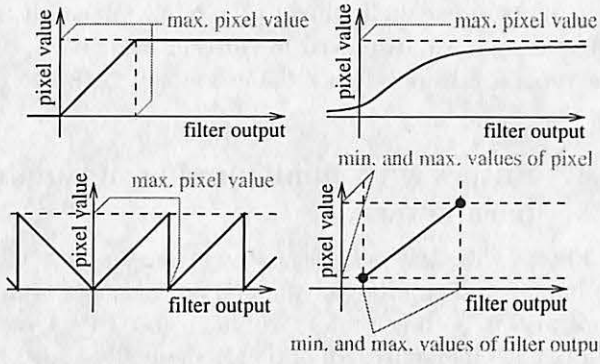


Figure 3: Functions to limit the pixel value and auto-scaling method.

An image's maximal pixel intensity value is usually limited to 8, 12, 16, etc. bits per pixel. When using an arbitrary mathematical function to compute pixel value, this maximal pixel value may be easily exceeded. To address this problem, we must limit the output of the filter again to the dynamic range of image. We may use limiting functions to do this, such as those shown in Figure 3. All these functions are nonlinear and may have a dramatic impact on the function of the filter.

To overcome this nonlinear behavior, an auto-scale method may be used instead of these limiting functions. However, the auto-scale method is specific to a given filter and may be image specific and this may prohibit the use of given filter with other images and/or may affect the dynamics of the IEC search process. Also, the dynamic range of the output image will be fixed when using the auto-scale method; the brightest and darkest pixels will be always of the same intensity.

Why not employ an IEC optimization method to also optimize the limiting function? In the ideal situation, the output of the filter function will never exceed the maximal value of pixel, and thus we wouldn't need a limiting function. However, the initial filters in IEC image processing are not optimal (usually random) and the optimal solution is found through the optimization process. Therefore, if we use filters which are able to exceed the permitted range, in order for the process to proceed correctly we must explicitly limit the output pixel values to the correct range.

### 3.2 Parametric Filters

Parametric filters with fixed structure are based on parametric functions, and IEC is used only to optimize their parameters. If the number of parameters is fixed, a simple GA algorithm will be sufficient for the optimization of such a filter. An example of this type of filter might be one consisting of three functions for adjusting pixel intensities in each of the red, green and blue (R,G,B) channels, where every of these functions is a linear spline function with a fixed number of 'joints'.

Filters in Figure 4 are severely limited in function compare to the algebraic filter of the type in equa-

tion (1). Filters in equation (1) are of the type:

$$r_{out} = f_r(r_{in}, g_{in}, b_{in}) \quad (3a)$$

$$g_{out} = f_g(r_{in}, g_{in}, b_{in}) \quad (3b)$$

$$b_{out} = f_b(r_{in}, g_{in}, b_{in}) \quad (3c)$$

Spline-based RGB example filters (Fig. 4) are of type:

$$r_{out} = f_r(r_{in}) \quad g_{out} = f_g(g_{in}) \quad b_{out} = f_b(b_{in}) \quad (4)$$

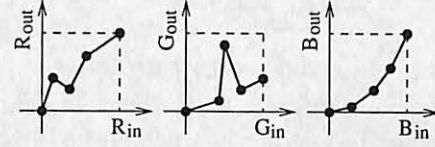


Figure 4: Linear spline based RGB filters.

Note, however, that while the 'simplicity' of this filter type will limit it in function, it will also reduce the search space for the IEC and may accelerate the search. Furthermore, this type of limitation of filter abilities might be useful in that it already prohibits the IEC from changing the image too much.

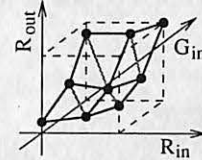


Figure 5: Two-dimensional linear spline filter.

To overcome the limitation of the given simple filter we might build a more capable parametric filter from modules of the type  $r_{out} = f_r(r_{in}, g_{in})$ , as in Figure 5. Finally, to build a parametric filter fully compatible with those in equation (3) we must use three-dimensional spline functions, though such a 3D parametric filter may still not be fully equivalent to the algebraic filter in equation (1). Linear spline functions have a rough shape compared to functions, such as  $x^2$  or  $\sin(x)$ , which may be used in an algebraic filter. To smooth the shape of the spline, cubic splines may be used or the number of 'joints' may be increased. Instead of splines, other functions may be used for parametric filters as well, such as Taylor series or Fourier series functions.

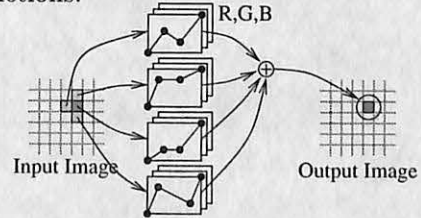


Figure 6: Multiple pixel spline filter.

The structure in Figure 6 represents a several-to-one pixel parametric filter. However, the sum in Figure 6 is not general enough, so for a truly general case we have to use multidimensional splines. Therefore, for 5 pixels the in input image and 3 channels (red, green, blue), we have to use a 15-dimensional spline function.

### 3.3 Structured Filters

Structured filters are filters built from a set of parametric or algebraic filters. This allows us to achieve the characteristics of algebraic filters while using parametric filters as building blocks. See Figure 7 for example.

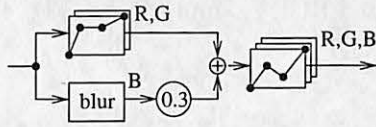


Figure 7: Structured filter.

This type of filter mimics the conventional approach, where image processing is accomplished by combining several predefined types of filters. Thus, the advantage of this type of filter is its similarity to traditional approaches, a property that might be welcomed by many potential users. Furthermore, the dynamics of the IEC search may be controlled when using structured filters by enabling or disabling particular building blocks.

## 4 Experiments and discussion

The schematic shown in Figure 8 illustrates the setup used in our experiments.

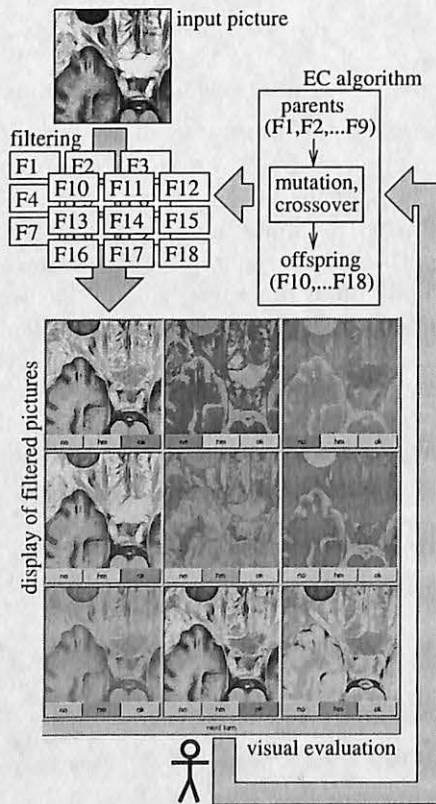


Figure 8: Experimental setup of IEC image processing application.

An input image is filtered by an initial set of random filters (F1,F2,...,F9), and nine filtered images are displayed for evaluation. These images are then evaluated by the user, and an EC generates a new set of filters (F10,F11,...,F18) based on that evaluation. These

new filters are again applied to the input image, the filtered images are displayed for further evaluation, and the process is iterated until the user is satisfied with the result.

### 4.1 Filters with small number of tunable parameters

Filters with less tunable parameters are convenient to handle and suitable for manual adjustment. On the contrary, it is difficult for a human user to set many parameters manually. An option to manually adjust the filter is therefore helpful in reducing user fatigue in IEC and, having fewer free parameters for EC optimization, also constrains the filter characteristic and may speed up optimization process.

We will discuss two types of filters with a fewer number of tunable parameters than conventional filters:

1. contrast/brightness based filter, which allows the for the adjustment of the contrast and brightness of the image and thus has two free parameters;
2. linear spline based filter consisting of two linear parts and thus having four free parameters, and single linear function with two free parameters.

These filters can be used in two variations:

- a) for intensity-based operation, where the same input-output relation is in parallel applied to all color components of image,
- b) in coloring mode, where a different filter of the same type is applied to every color component of image. This allows for the 'coloring' of gray-scale images or for manipulating the color tone of a color image.

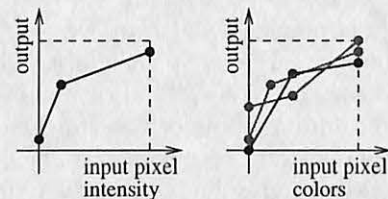


Figure 9: Example of linear spline filter with two linear parts for intensity operation (with 4 free parameters) and for coloring operation (with 12 free parameters).

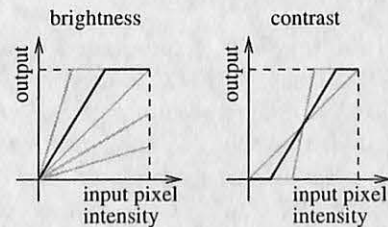


Figure 10: Example of contrast/brightness based filter in intensity mode; parameters of filter determine orientation of line which intersect begin of coordinate system or center of working square area, respectively.



## 4.2 Enhancing brightness/contrast of photographs

We compare two scenarios for enhancing the brightness/contrast of a photograph to illustrate their performance. In the first approach, we use IEC-based optimization of filter parameters, and in the second, we use conventional EC optimization of filter parameters according to a histogram evaluation. Equal distribution of pixel intensity is sometimes a useful strategy for a good looking photograph, and a pixel histogram along with intensity is a good tool to monitor the distribution. To evaluate a measure of the picture quality, we use the width of histogram and the average distance of the number of equal pixels from average number of equal pixels, as shown in Figure 11.

The user is involved in these two scenarios in different ways: in the IEC case, she/he interactively numerically evaluates candidate images; in the histogram-based EC optimization case, she/he designs evaluation rule.

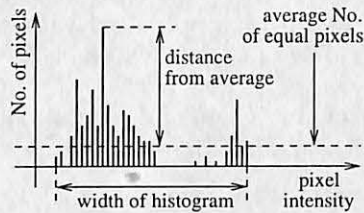


Figure 11: Evaluation of the histogram based on its width and average distance from average value.

The optimization results are illustrated in Figure 13, and the original photograph with corresponding histogram is in Figure 12. The chosen photograph has a narrow histogram, which is easy for us to see improvements. Filters used in these examples are 3-point splines, i.e. with 12 parameters. Both optimization results with IEC and EC were obtained within about 15 minutes. The best EC optimized ones were chosen among several runs and are shown in the figures. Figure 14 was obtained using 4-point splines (with 18 parameters), which have higher capability to control contrast and colors. However, the contrast of Figure 14 looked too stronger than that of Figure 13 on a computer display and looked worse than the latter figures.

Both approaches are applicable for the given task and produce results in reasonable time. The IEC approach achieves slightly better color tones and picture appearance. This can be expected because IEC systematically incorporates users' aesthetic preferences into optimization. The advantage of the histogram-evaluated EC based approach is its ability to be used automatically once the evaluation rule is designed. Although this mode of optimization may need more generations (or also more time) to achieve acceptable results, it can be done conveniently off-line.

When comparing the optimization of 2-parameter, 12-parameter and more complex filters, both approaches

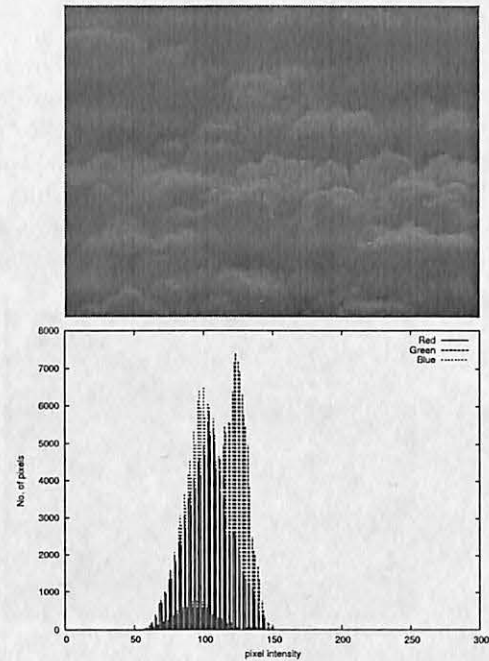


Figure 12: Original photograph with corresponding histogram.

increase in difficulty for increasingly complex filters, but also allow better results to be obtained. Direct manual adjustment of filter parameters in more complex filters is less effective. Conversely, in initial phase of IEC search, manual adjustment of simple filters is an effective tool.

In the task of enhancing the contrast/brightness of photographs, IEC is superior to EC with a histogram based evaluation because it is able to achieve superior picture quality.

## 4.3 Combination of interactive and non-interactive evaluation

Incorporation of the previously discussed histogram-based evaluation into IEC-based image processing can bring some of the convenience of off-line processing into IEC, though there is no guarantee as to the quality of results, as discussed in the previous section. This guarantee may not be provided in off-line processing but can only be provided through interactive user evaluations. The combination of them is simple; an evaluation rule is used to pre-set evaluation values of displayed images in GUI, and these values are displayed to a user with corresponding images. The user then evaluate only images whose pre-set evaluation values differ from those of the user's visual inspection. If the evaluation rule matches to user's preferences even if in part, it can reduce user's fatigue by reducing the number of evaluation.

If the complexity of optimization filters increases, it becomes difficult to use histogram based pre-evaluation. Thanks, however, to the constrained character of contrast/brightness adjustments, improvement by adopting histogram based pre-evaluations in this task is considerable.

## 5 Conclusions

Modeling filters with fewer parameters allows effective direct manipulation of parameters and pre-evaluation in IEC-based image processing. These are effective tools for reducing user fatigue. In addition, IEC is superior to EC with manual design of evaluation rules because of the more direct incorporation of the user's aesthetic preferences.

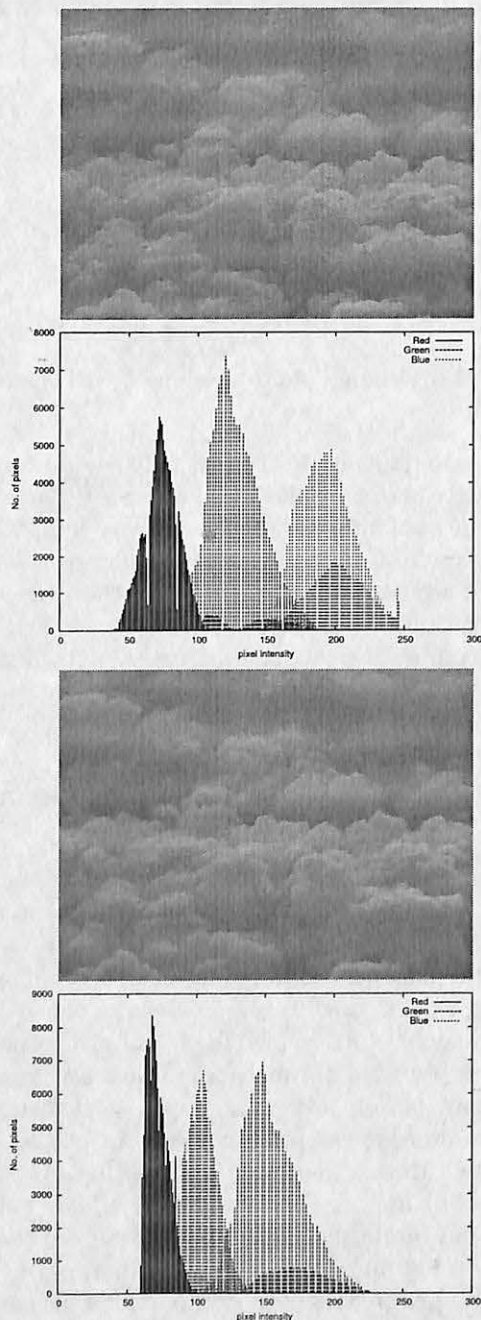


Figure 13: Example of photograph optimized by EC with histogram-based evaluation and by IEC.

## References

[1] R. Jakša, S. Nakano and H. Takagi, "Image Filter Design with Interactive Evolutionary Computa-

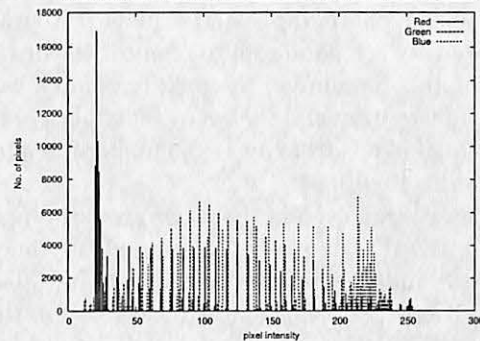
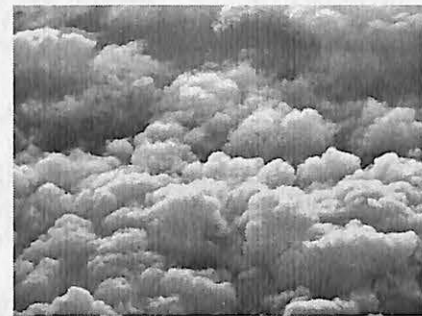


Figure 14: Photograph optimized with more complex filter, which allows more radical changes.

- tion," Proc. Int. Conf. on Computational Cybernetics (ICCC2003), Siófok, Hungary, Aug. 2003.
- [2] R. Jakša and H. Takagi, "Analysis and Evaluation for Interactive Evolutionary Computation-Based Image Processing," Proceedings of MPS Symposium, IPSJ Symposium Series vol. 2003, Kyoto, Japan, pp.243–250, Jan. 2003.
- [3] T. Mutoh, N. Komagata and Y. Ueda, "An experimental study for automatically generating image filter sequence by using simulated breeding," Proc. Workshop on Interactive Evolutionary Computation, Fukuoka, Japan, pp. 7–12, Mar. 1998, (in Japanese).
- [4] K. Otoba, K. Tanaka and M. Hitafuji, "Image processing and interactive selection with Java based on genetic algorithms," Proc. 3rd IFAC/CIGR Workshop on Artificial Intelligence in Agriculture, Makuhari, Japan, pp. 83–88, Apr. 1998.
- [5] R. Poli and S. Cagnoni, "Genetic programming with user-driven selection: Experiments on the evolution of algorithms for image enhancement," Proc. 2nd Annual Conference on Genetic Programming, Palo Alto, CA, USA, pp. 269–277, 1997.
- [6] H. Takagi, "Interactive Evolutionary Computation: Fusion of the Capacities of EC Optimization and Human Evaluation," Proceedings of the IEEE, vol.89, no.9, pp.1275–1296, Sep. 2001.
- [7] H. Takagi and N. Hayashida, "Interactive EC-based Signal Processing," 4th Asia-Pacific Conference on Simulated Evolution and Learning (SEAL2002), Singapore, pp. 375–379, Nov. 2002.