Improved Image Halftoning Technique Using GAs with Concurrent Inter-block Evaluation

Emi Myodo, Hernán Aguirre, and Kiyoshi Tanaka

Dept. of Electrical and Electronic Engineering, Faculty of Engineering
Shinshu University, 4-17-1 Wakasato, Nagano 380-8553, Japan
{myodo,ahernan,ktanaka}@iplab.shinshu-u.ac.jp

Abstract. In this paper we propose a modified evaluation method to improve the performance of an image halftoning technique using GAs. We design the algorithm to avoid noise in the fitness function by evolving all image blocks concurrently, exploiting the inter-block correlation, and sharing information between neighbor image blocks. The effectiveness of the method when the population and image block size are reduced, and the configuration of selection and genetic operators are investigated in detail. Simulation results show that the proposed method can remarkably reduce the entire processing time to generate high quality bi-level halftone images.

1 Introduction

Recently the application of evolutionary algorithms (EAs) to real-world problems has been rapidly increasing. Signal processing is one of the areas in which methods using EAs are steadily being developed [1]. In this work, we focus on the image halftoning problem using genetic algorithms (GAs). In the halftoning problem a N-gray tone image must be portrayed as a n-gray tone image, where $n < N$. GAs have been used for halftoning in two ways. One approach seeks to evolve (or co-evolve) filters, which are applied to the input N-gray tone image to generate a halftone image [2,3,4]. The second approach searches directly for the optimum halftoning image having as a reference the input N-gray tone image. Our work fit in the latter approach.

Kobayashi et al. [5] first proposed a direct search GA based halftoning technique to generate bi-level halftone images. This technique divides the input images into non-overlapping blocks and uses a simple GA with a specialized two dimensional crossover to search the corresponding optimum binary patterns (high gray level precision and high spatial resolution). The method's major advantages are that (i) it can generate images with a specific desired combination of gray level precision and spatial resolution, and (ii) it generates bi-level halftone images with quality higher than traditional schemes such as ordered dithering, error diffusion, and so on [6]. However, the method uses a substantial amount of processing time and computer memory. Recently, Aguirre et al. [7] have proposed an improved GA (GA-SRM) to overcome the drawbacks of [5]. GA-SRM applies varying mutation parallel to crossover and background mutation, putting

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2724, pp. 2264–2276, 2003.

the operators in a cooperative-competitive stand with each other by subjecting their offspring to extinctive selection [8,9]. The halftoning technique using GA-SRM [7], compared to the conventional GA based technique [5], can achieve a 98% reduction in population size and a 70%-85% reduction in processing time while generating high quality bi-level halftone images. Additional related work can be found in [10], where the halftoning problem is treated as a multiobjective optimization problem to generate simultaneously halftone images with various combinations of gray level precision and special resolution. Also, in [11] the bi-level halftoning technique is extended to a multi-level halftoning technique using GAs. The gains in performance achieved by GA-SRM alone are substantial, yet the direct GA-SRM based halftoning method still requires much more processing time than traditional methods, such ordered dithering and error diffusion, and enhancing efficiency is required.

The GA based halftoning methods mentioned above evolve all image blocks independently from each other. A side effect of this is that the evaluation function becomes approximate for the pixels close to the boundaries between image blocks, which introduces false optima and delays the search. In this paper we improve further GA based halftoning with a modified evaluation method that contemplates inter-block correlation between neighbor blocks evolving all image blocks concurrently. The effectiveness of the method when the population and image block size are reduced, and the configuration of selection and genetic operators are investigated in detail. Simulation results verify that the modified scheme further accelerates the search speed to generate high quality halftone images, reducing processing time to less than $\frac{1}{100}$ of the time required by the conventional GA based halftoning method proposed in [5].

2 Image Halftoning Scheme Using GA

2.1 Individual Representation

An input image is first divided into non-overlapping blocks $\boldsymbol{D}$ consisting of $r \times r$ pixels to reduce the search space of solutions [5,7]. The GA uses an individual $\boldsymbol{x}$ with a $r \times r$ two dimensional representation for the chromosome. In the case of bi-level halftoning each element of the chromosome $x(i,j) \in \{0,1\}$. Figure 1 illustrates the division of the image in blocks and an example of individual $\boldsymbol{x}$ corresponding to a current block $\boldsymbol{D}$.

2.2 Evaluation

We evaluate chromosomes with two kinds of evaluation criteria. (i) One is high gray level precision (local mean gray levels close to the original image), and (ii) the other is high spatial resolution (appropriate contrast near edges) [5]. The bi-level image halftoning technique calculates a gray level precision error by

$$E_m = \sum_{(i,j)\in D} \frac{1}{r^2} |g(i,j) - \hat{g}(i,j)| \tag{1}$$

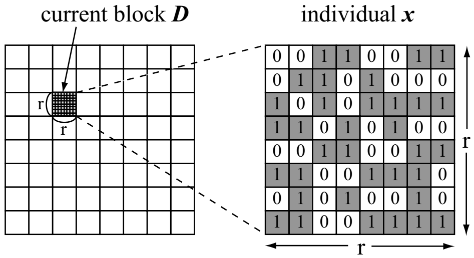


Fig. 1. Image division and individual representation($r = 8$)

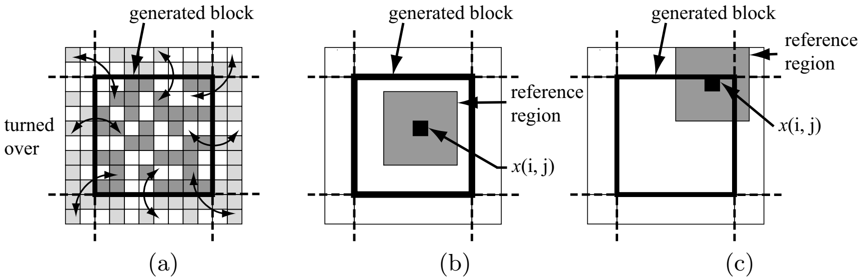


Fig. 2. (a) shows a current generated block $\boldsymbol{x}$ with its binary pattern copied around block boundaries for gray level estimation. (b) and (c) show examples of generated pixels and their reference region to calculate gray level estimation.

where $g(i,j)$ $(i,j{=}0,1,\cdots,r\text{-}1)$ is the gray level of the (i,j)-th pixel in the input image block, and $\hat{g}(i,j)$ is the estimated gray level associated to the (i,j)-th pixel of the generated halftone block $(x(i,j))$. To obtain $\hat{g}(i,j)$ a reference region around $x(i,j)$ is convoluted by a gaussian filter that models the correlation among pixels. In order to reduce discontinuity around block boundaries, the pixel pattern of $\boldsymbol{x}$ is copied around the boundary regions, as shown in Fig. 2a, and used to calculate the gray level estimation $\hat{g}(i,j)$. Figure 2b and Fig. 2c illustrate two examples of $x(i,j)$ and its reference region to calculate $\hat{g}(i,j)$. In Fig. 2b the reference region lies within the generated block while in Fig. 2c it exceeds the block boundaries and includes also the copied pixels.

In order to preserve the edge information of the input image well, the spatial resolution error in the bi-level image halftoning technique is calculated by

$$E_c = \sum_{(i,j)\in D} \frac{1}{r^2} |G(i,j) - B(i,j)| \tag{2}$$

$$G(i,j) = g(i,j) - \bar{g}(i,j)$$

$$B(i,j) = (x(i,j) - \frac{1}{2})N$$

where $G(i,j)$ is the difference between the gray level $g(i,j)$ of the (i,j)-th pixel in the input image block and its neighboring local mean value $\bar{g}(i,j)$.

The two errors E_m and E_c are combined into one single objective function as

$$E = \alpha_m E_m + \alpha_c E_c \tag{3}$$

where α_m and α_c are weighting parameters of E_m and E_c, respectively. The chromosome's fitness is assigned by

$$F = E_{max} - E \tag{4}$$

where E_{max} is the error associated with the worst chromosome in a population. The GA is used to search for optimum compromise between (i) and (ii) with the above fitness function.

2.3 Genetic Operators and Selection

The improved GA-SRM for the halftoning problem [7] is based on a model of GA that puts genetic operators with complementary roles in a cooperative-competitive stand with each other [8,9]. The main features of the model are (i) two genetic operators to create offspring: Self-Reproduction with Mutation (SRM) that puts emphasis on mutation, and Crossover and Mutation (CM) that puts emphasis on recombination, (ii) an extinctive selection mechanism, and (iii) an adaptive mutation schedule that varies SRM's mutation rates from high to low values based on SRM's own contribution to the population. In [7], CM uses the same two dimensional crossover as [5] and SRM is provided with an Adaptive Dynamic Block (ADB) mutation schedule, in which the size of mutation block is dynamically adjusted. Extinctive selection is implemented with (μ, λ) proportional selection.

3 Proposed Method

3.1 Problems

As indicated in 2.2, the GA based halftoning schemes in [5] and [7] evolve the image blocks independently from each other, copy the binary pattern of the current block around the boundary regions, and use that information to evaluate the quality of the generated block. Due to the expected high correlation between neighboring pixels in an image, the pixels copied around the boundaries of the generated block aim to reduce discontinuities between blocks. However, these pixels are not the true information of the generated neighbor blocks. Although mathematically the same fitness function is used for every pixel, from an information standpoint the conventional GA based halftoning method induce a kind of approximate fitness function [12] for the pixels close to the boundary regions, which introduces false optima. This can mislead the algorithm and greatly affect its search speed. Note that if the area of image block is reduced the number of pixels evaluated with the approximate function (e.g. Fig. 2c) will increase while the number of pixels evaluated with the true fitness function (e.g. Fig. 2b) will decrease, negatively affecting the quality of the generated image and delaying processing time. The noise introduced by the approximated function is a real obstacle for further reduction of processing time.

3.2 Modified Evaluation Method

To have a fitness function that models the halftoning problem with higher fidelity, we make use of the inter-block correlation between neighbor blocks in the evaluation, linking each block with its neighbor blocks and sharing some genetic information between them. A GA is allocated to each block and each GA evolves its population of possible solutions concurrently. In this process the best individuals $\boldsymbol{x}^{*(t-1)}_{u,v}$ in the neighbor populations are generationally referred and used to calculate the fitness values for individuals $\boldsymbol{x}^{(t)}_{k,l}$ in the current population, as shown in Fig. 3. With this procedure of information sharing between populations we can supplement incomplete information in the evaluation process of [5,7] expecting that it would contribute to reduce processing time, improve the image quality around block boundaries, and allow further reductions of block size. Parallel implementations can be realized with the required number of processing units, linking at most 8 neighbor units. In this work the parallel GA is simulated as concurrent processes in a serial machine.

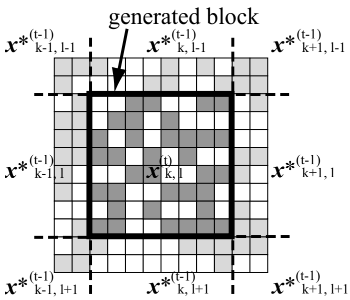


Fig. 3. A current block and connected neighbor blocks for gray level estimation

4 Results and Discussion

4.1 Experimental Setup

In this paper, we apply the proposed method to a canonical GA (cGA) [5] and GA-SRM [7]. To test the alogorithms we use SIDBA's benchmark images in our simulation. Unless stated otherwise, results presented here are for image "Lenna". The size of the original image is 256×256 pixels with $N = 256$ gray levels and the generated images are bi-level halftone images ($n = 2$). The image block size is $r \times r = 16 \times 16$ and the population size is 200. The weighting

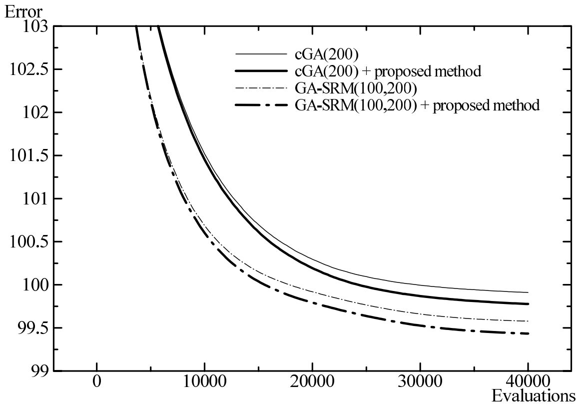


Fig. 4. Performance by cGA(200) and GA-SRM(100,200) with proposed and conventional method

Table 1. Comparison of number of evaluations

method	cGA(200)	GA-SRM(100,200)
conventional	1.000T	0.510T
proposed	0.695T	0.430T

parameters in Eq. (3) are set to $\alpha_m = 0.2$ and to $\alpha_c = 0.8$. In the case of the canonical GA the crossover probability is set to $p_c = 1.0$ and mutation probability to $p_m = 0.001$, similar to [5]. In our simulation we consider the result (error value of Eq. (3)) achieved by cGA with this settings after $T = 40,000$ evaluations as a reference value for image quality [5]. In the case of GA-SRM, for CM the crossover and mutation probabilities are set to $p_c = 1.0$ and $p_m^{(CM)} = 0.001$, respectively. For SRM, ADB mutation schedule with bit swapping mutation is used [7]. The balance for offspring creation is $\lambda_{CM} : \lambda_{SRM} = 1 : 1$, and the ratio between the number of offspring and the number of parents is $\mu : \lambda = 1 : 2$.

4.2 Effects in Conventional Schemes

First, we show the effect of the proposed method in cGA and GA-SRM with large populations. Figure 4 plots the error reduction over the generations by cGA(200), which denotes a canonical GA with a population size of 200 individuals, and by GA-SRM(100,200), which denotes GA-SRM with a parent population of $\mu = 100$ and an offspring populaion of $\lambda = 200$. From Fig. 4 we can see that both cGA and GA-SRM can achieve higher image quality with the proposed method than with the conventional evaluation method. Table 1 shows the number of evaluations needed to reach the reference value for image quality. We can reduce the number of evaluations about 31% in cGA and 16% in GA-SRM. Note that GA-SRM even with the conventional evaluation method is faster than cGA with the proposed evaluation method.

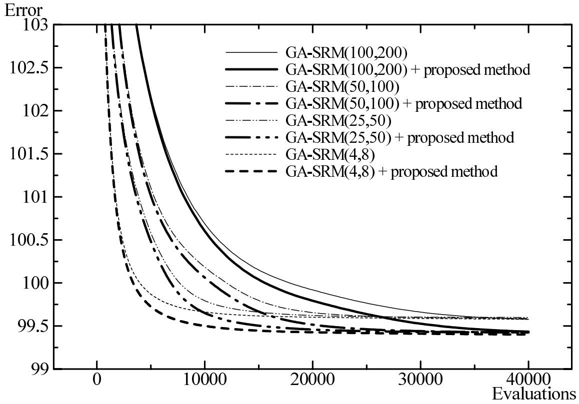


Fig. 5. Performance by GA-SRM with proposed and conventional method in different population sizes

Table 2. Effect of population size reduction

method	GA-SRM (100, 200)	(50, 100)	(25, 50)	(4, 8)
conventional	0.510T	0.330T	0.211T	0.115T
proposed	0.430T	0.290T	0.185T	0.094T

4.3 Effect in Population Size Reduction

Second, since memory is an important issue in this application we study the effect on performance of reducing the population size. Figure 5 plots the error transition over the evaluations by GA-SRM with the conventional and proposed method. From Fig. 5 we can see that GA-SRM with the proposed method using smaller population sizes accelerates the search speed without deteriorating the final image quality. Table 2 shows the number of evaluations needed to reach the reference value for image quality.

4.4 Effects in Block Size Reduction

Next, we study the effect of reducing the size of the image block fixing the population size to $(\mu, \lambda) = (4, 8)$ in GA-SRM. Here, the mutation probability for CM is set to $p_m^{(CM)} = \frac{1}{r \times r}$ [13], because this value for mutation rate causes better performance in combination with extinctive selection. Figure 6 plots the error reduction over the evaluations for “Lenna” and Table 3 shows the number of evaluations needed to reach the image quality reference value for “Lenna” and other benchmark images. Note that with the proposed method we can further accelerate the search by reducing the block size to be evolved and still keep high image quality. For example, in case of Lenna and $r \times r = 4 \times 4$ the proposed method needs only 240 evaluations to achieve the image quality reference value

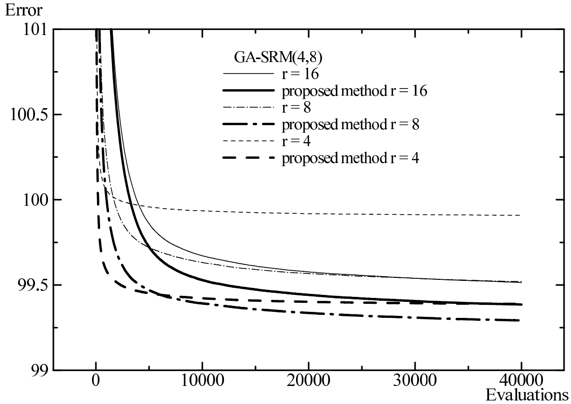


Fig. 6. Performance by GA-SRM(4,8) with proposed and conventional method using different block sizes

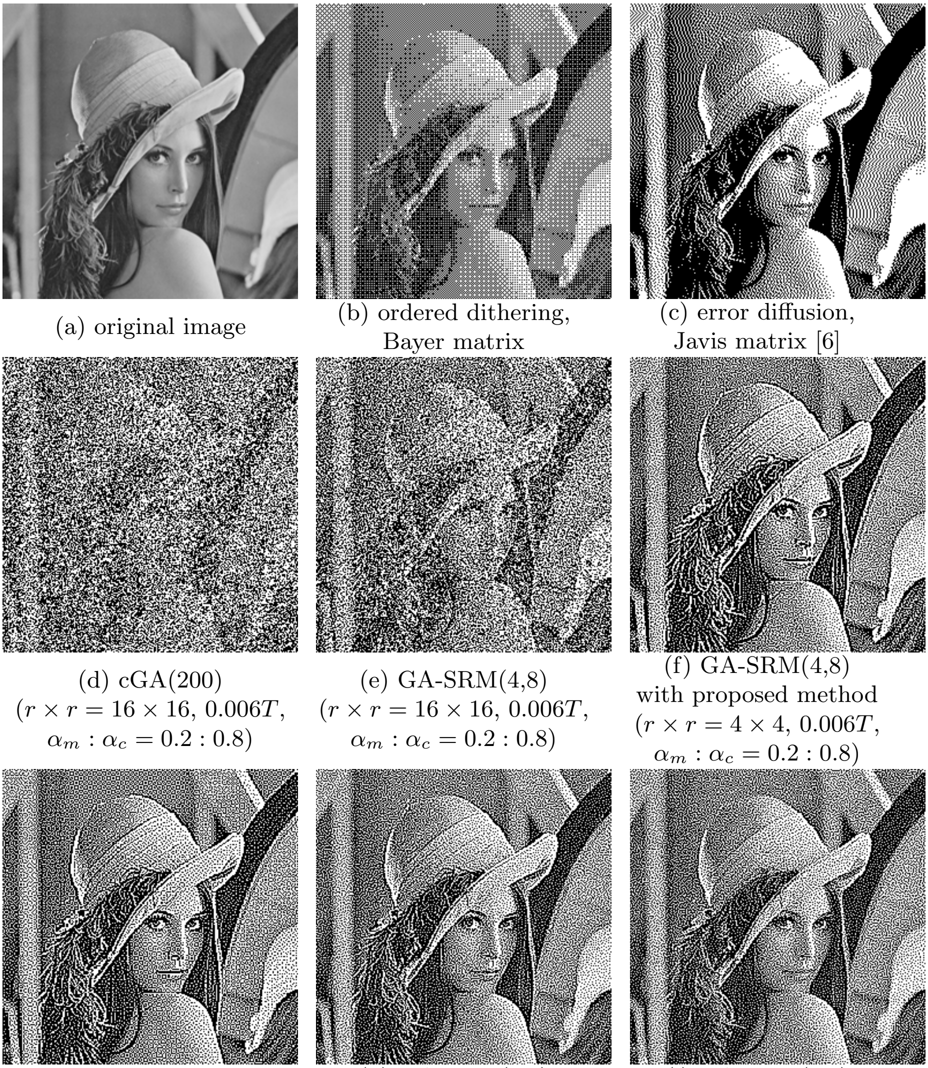
(the same image quality obtained by cGA after 40,000 evaluations)†, which means less than $\frac{1}{100}$ of the processing time compared with the original scheme [5]. Running software implementations of the algorithms in a Pentium IV processor (2GHz), it takes about 7 seconds to generate one image of 256 $\times$ 256 pixels that reaches the reference value for image quality. Processing time increases in proportion to image pixels. From Table 3 note also that we could consistently observe similar behavior for other benchmark images. On the other hand, the conventional method considerably delays the search and cannot achieve visually satisfactory images for various test images. As mentioned in 3.2, in this paper the parallel implementation of the GA is simulated as concurrent processes in a serial machine. A true parallel implementation will benefit from a further reduction in processing time due to the distribution of work to several processors.

Figure 7 shows the original image "Lenna" and several generated images by traditional methods, such ordered dithering and error diffusion, and GA based halftoning methods with the conventional and proposed evaluation method.

4.5 Inter-block Information Sharing Gap

The effect on image quality and processing time of using a generational gap to perform the information sharing was also examined [14]. From our experiments we found that a *Gap* of 10 generations reduces substantially the frequency needed to share the inter-block information and still achieves a very significant reduction in processing time (only 1% of the processing time is needed compared to the original cGA method [5]). For values of *Gap* greater than 10 the search is delayed and no significant gain is observed reducing the number of times the information is updated.

† Here we compare average number of evaluation per pixel between different schemes, which is a good coincident with the actual processing time.



(a) original image

(b) ordered dithering, Bayer matrix

(c) error diffusion, Javis matrix [6]

(d) cGA(200) ($r \times r = 16 \times 16$, $0.006T$, $\alpha_m : \alpha_c = 0.2 : 0.8$)

(e) GA-SRM(4,8) ($r \times r = 16 \times 16$, $0.006T$, $\alpha_m : \alpha_c = 0.2 : 0.8$)

(f) GA-SRM(4,8) with proposed method ($r \times r = 4 \times 4$, $0.006T$, $\alpha_m : \alpha_c = 0.2 : 0.8$)

(g) GA-SRM(4,8) ($r \times r = 4 \times 4$, $1.000T$, $\alpha_m : \alpha_c = 0.2 : 0.8$)

(h) GA-SRM(4,8) with proposed method ($r \times r = 4 \times 4$, $0.006T$, $\alpha_m : \alpha_c = 0.5 : 0.5$)

(i) GA-SRM(4,8) with proposed method ($r \times r = 4 \times 4$, $0.006T$, $\alpha_m : \alpha_c = 0.9 : 0.1$)

Fig. 7. Original “Lenna” and output images

Table 3. Effect of block size reduction for "Lenna" and other benchmark images

image	method	GA-SRM(4,8)		
		16×16	8×8	4×4
Lenna (256×256)	conventional	0.112T	0.054T	0.84T
	proposed	0.090T	0.029T	0.006T
Aerial (256×256)	conventional	0.129T	0.062T	—
	proposed	0.095T	0.032T	0.006T
Airplane (256×256)	conventional	0.105T	0.056T	—
	proposed	0.088T	0.030T	0.005T
Girl (256×256)	conventional	0.108T	0.053T	—
	proposed	0.093T	0.027T	0.005T
Moon (256×256)	conventional	0.133T	0.059T	0.073T
	proposed	0.100T	0.031T	0.005T
Title (256×256)	conventional	0.127T	0.050T	0.018T
	proposed	0.117T	0.043T	0.010T
Woman (256×256)	conventional	0.120T	0.058T	0.130T
	proposed	0.092T	0.031T	0.006T

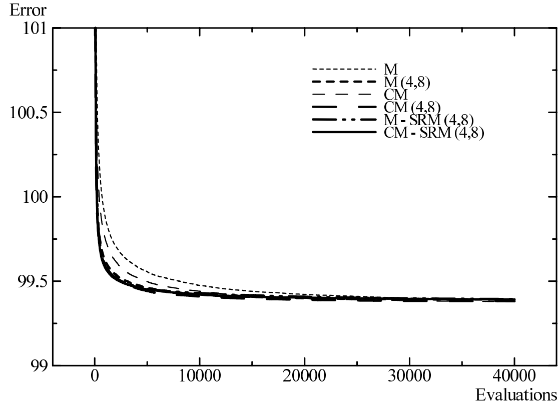
— : never achieved the reference fitness value by cGA(200) with T=40,000 evaluations

5 GA Configuration in Different Block Sizes

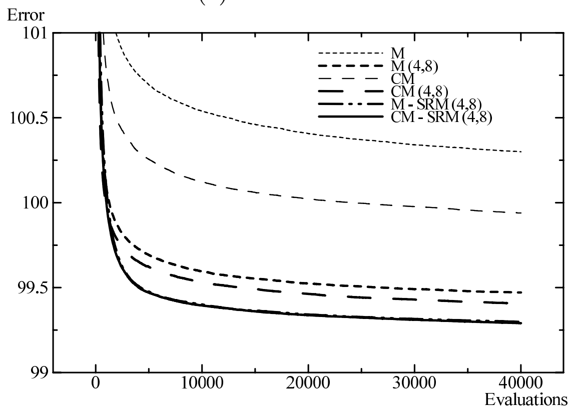
In 4.4 we showed that by using the proposed method the block size can be reduced from $r \times r = 16 \times 16$ to $r \times r = 4 \times 4$ to generate high quality bi-level halftone images. The block size determines the size of search space. Smaller blocks imply an easier optimization problem and simpler optimization methods could be used. However, smaller blocks would also require more hardware resources, especially if we want to have a true parallel implementation of the algorithm. On the other hand, bigger blocks would increase the difficulty of finding the optimum and enhanced optimization methods would be required. This trade-off is an important issue that should be considered during the implementation of this application.

In this section we study the configuration of selection and genetic operators in GA for different block sizes. Figure 8 shows results by several EAs set with an offspring population of 8. M indicates only mutation with $p_m = \frac{1}{r \times r}$, CM is crossover ($p_c = 1.0$) followed by background mutation ($p_m = \frac{1}{r \times r}$), and SRM is the adaptive varying mutation operator. If the values of (μ, λ) are indicated, e.g. (4,8), the algorithm uses (μ, λ) proportional selection. Otherwise, the algorithm uses proportional selection.

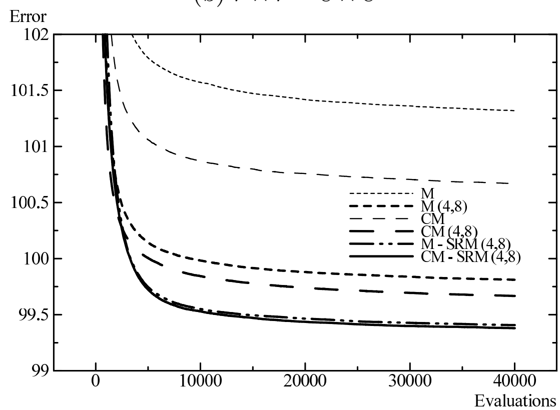
From these figures, we can see that GA-SRM (CM-SRM(4,8)) is the most reliable method in all block sizes to generate a high quality halftone image rapidly. But to make the implementation of this problem easier in certain block size, other methods can be used. For example, when halftone images are generated in block size $r \times r = 4 \times 4$, we can use M(4,8) method without disadvantages because the performance of M(4,8) is as good as CM-SRM(4,8) as shown in Fig. 8a.



(a) $r \times r = 4 \times 4$



(b) $r \times r = 8 \times 8$



(c) $r \times r = 16 \times 16$

Fig. 8. Configuration of selection and genetic operators in GA for different block sizes

6 Conclusions

In this paper we have presented a modified evaluation method to improve the performance of an image halftoning technique using GAs. The proposed algorithm avoid noise in the fitness function by evolving all image blocks concurrently, exploiting the inter-block correlation, and sharing information between neighbor image blocks. The effectiveness of the method when the population and image block size are reduced, and the configuration of selection and genetic operators were investigated in detail. We could verify that the proposed method can remarkably reduce the processing time to less than $\frac{1}{100}$ of the time required by the conventional GA based halftoning method without deteriorating high image quality. The proposed method also generates images with less noise around block boundaries.

As future work, we are planning to extend this method to multi-level and color halftoning using multiobjective optimization techniques [10].

References

1. K. F. Man, K. S. Tang, S. Kwong and W. A. Halong, *Genetic Algorithms for Control and Signal Processing*, Springer-Verlag, 1997.
2. J. Newbern and V. M. Bove, Jr., "Generation of Blue Noise Arrays by Genetic Algorithm", *Proc. SPIE Human Vision and Electronic Imaging II*, vol.3016, pp.441–450, 1997.
3. J. T. Alander, T. Mantere, and T. Pyylampi, "Threshold Matrix Generation for Digital Halftoning by Genetic Algorithm Optimization", *Proc. SPIE Intelligent Systems and Advanced Manufacturing: Intelligent Robots and Computer Vision XVII: Algorithms, Techniques, and Active Vision*, vol.3522, pp.204–212, 1998.
4. J. T. Alander, T. Mantere, and T. Pyylampi, Digital Halftoning Optimization via Genetic Algorithms for Ink Jet Machine, *Developments in Computational Mechanics with High Performance Computing*, CIVIL-COMP Press, Edinburg, UK, pp.211–216, 1999.
5. N. Kobayashi and H. Saito, "Halftoning Technique Using Genetic Algorithm", *Proc. IEEE ICASSP'94*, Vol.5, pp.105–108, 1994.
6. R. Ulichney, *Digital Halftoning*, MIT Press, Cambridge, 1987.
7. H. Aguirre, K. Tanaka and T. Sugimura, "Accelerated Halftoning Technique Using Improved Genetic Algorithm with Tiny Populations", *Proc. IEEE Intl. Conf. on Systems, Man, and Cybernetics*, pp.905–910, 1999.
8. H. Aguirre, K. Tanaka and T. Sugimura, "Cooperative Model for Genetic Operators to Improve GAs", *Proc. IEEE Intl. Conf. on Information, Intelligence and Systems*, pp.98–106, 1999.
9. H. Aguirre, K. Tanaka, T. Sugimura and S. Oshita, "Cooperative-Competitive Model for Genetic Operators: Contributions of Extinctive Selection and Parallel Genetic Operators", *Proc. Late Breaking Papers at Genetic and Evolutionary Computation Conference 2000*, pp.6–14, 2000.
10. H. Aguirre, K. Tanaka and T. Sugimura, "Halftone Image Generation with Improved Multiobjective Genetic Algorithm", *Proc. First Intl. Conf. on Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Springer, vol.1993, pp.501–515, 2001.

11. T. Umemura, H. Aguirre, K. Tanaka,"Multi-level Halftone Image Generation with Genetic Algorithms," *Proc. WSEAS Multiconference on Applied and Theoretical Mathematics*, pp.202–207, 2001.
12. Y. Jin, M. Olhofer, and B. Sendhoff, "A Framework for Evolutionary Optimization With Approximate Fitness Functions", *IEEE Trans. Evolutionary Computation*, vol.6, no.5, pp.481–494, 2002.
13. T. Bäck, "Optimal mutation rates in genetic search", *Proc. 5th Int'l Conf. on Genetic Altorithms*, Morgan Kaufmann, pp.2–8, 1993.
14. E. Myodo, H. Aguirre, K. Tanaka,"Improved Image Halftoning Scheme Using GAs with a Modified Evaluation Method " *IEEE-EURASIP Int'l Workshop on Signal and Image Processing*, to appear, 2003.