A Low Complexity and Lossless Frame Memory Compression for Display Devices

Teresa Liew Bao Yng, Byung-Gook Lee and Hoon Yoo

Abstract — *In this paper, we introduce a new low complexity and lossless algorithm based on subband decomposition with the modified Hadamard transform (MHT) and adaptive Golomb-Rice (AGR) coding for display devices. The main goal of the proposed method is to reduce memory requirement for display devices. A basic unit of the proposed method is a line of the image so that the method is processed line by line. Also, MHT and AGR coding are utilized to achieve low complexity. The major improvement of the method is from the use of AGR and subband processing compared with exiting methods which are similar to the method in terms of complexity and applications. Simulation results show that the algorithm achieves a superior compression performance to the existing methods. In addition, the proposed method is hardware-friendly and could be easily implemented in any display devices*[1].

Index Terms — Frame recompression, lossless compression, low complexity, displays devices.

I. INTRODUCTION

Research in compression techniques has stemmed from the ever-increasing need for efficient data transmission, storage and utilization of hardware resources. Uncompressed graphics, audio and video data require considerable storage capacity and transmission bandwidth. The purpose of image compression is to save storage space and to reduce transmission time for image data. It aims at achieving a high compression ratio (CR) while preserving good fidelity of decoded images.

In general, image compression algorithms are divided into two categories: lossless and lossy compression. Compression techniques belonging to the first category have the main characteristic that the image involved can be perfectly reconstructed from the compressed file. Thus, they have no information loss in the compression and decompression process. On the other hand, lossy compression methods discard some part of image data which is likely undetectable loss in image quality by the human visual system. Basically, to achieve better image quality, image compression algorithms have been developed based on the image data characteristics or target applications. For example, image and video are applied to totally different image compression algorithms, respectively. Moreover, gray level images and bi-level images are also applied to different methods. This is the reason that many image compression techniques are developed in the literature. Also, this situation can be possible in target applications which utilize or handle images differently.

Recently, LCD devices for a high quality display are widely used in applications including display panels of mobile devices like digital TV panel, PC screen panel, portable DVD player, digital still camera, etc. Most display systems consist of three main devices which are a display timing controller, a display driver that includes a frame buffer and a display panel. As image processing technology develops, larger and larger images are integrated into the LCD devices. Thus, the much more frame buffer is required. However, the display devices are now very sensitive to cost. Therefore, to reduce the cost, it is beneficial to reduce the required amount of storage devices while preserving good quality of the image on the display devices.

A number of researches have been proposed concerning on lossless compression with low complexity [1-5, 8]. Among these techniques, some of them are related with the compression of frame memory for video decoders [1-3]. For example, the method in [1] reduced the memory consumption at the sacrifice of the image quality using a quantization process. This method is a hardware-friendly algorithm for the MPEG-2 video decoder. The method in [2] is a lossless compression for reducing the frame memory of the H.264 video decoder. It employs DPCM and Huffman coding to keep the implementation easy. The method in [3] is also for the H.264 decoder and utilizes the architecture of the H.264 decoder. The method in [8] is a standard compression method for lossless and near lossless image compression. It is relatively a low complexity algorithm compared with the standards such as JPEG, JPEG2000. However, it is not considered to be a low complexity method for frame memory reduction in video coders or LCD devices because it requires many multipliers in hardware implementation and it does not offer the random accessibility in the compressed bitstream.

In this paper, we focus on a low complexity and lossless frame memory compression and decompression algorithm for LCD devices. As mentioned above, frame memory reduction for LCD devices is required and its hardware implementation should be concerned. Moreover, the image quality of LCD devices is important factor so lossy compression may be

[1] This work was supported in part by Dongseo University and in part by the post BK program.

Teresa Liew Bao Yng is currently a master student in Dongseo University, Busan, South Korea (e-mail: teresalby@dit.dongseo.ac.kr).

Byung-Gook Lee is with the division of computer and information engineering,, Dongseo University, Busan, South Korea (e-mail: lbg@dongseo.ac.kr)

Hoon Yoo is the corresponding author with the division of computer and information engineering, Dongseo University, Busan, South Korea (corresponding e-mail: hunie@dongseo.ac.kr).

Contributed Paper
Manuscript received July 15, 2008

avoidable. Thus, we introduce a low complexity and lossless compression algorithm for LCD devices. The proposed method provides a line by line processing since the frame memory in LCD devices is accessed line by line. To accomplish the proposed method, we employ the modified Hadamard transform (MHT) and an adaptive Golomb Rice coder for every line data of an input image. The proposed method gives a substantial gain compared with the existing methods [1-2] and keeps the computation complexity low.

The rest of this paper describes the details of the proposed method. In section II, the new low complexity and lossless compression algorithm is briefly introduced and the detailed architecture of the proposed algorithm is described. In section III, the simulation and results are presented. Conclusions and future work are mentioned in section IV.

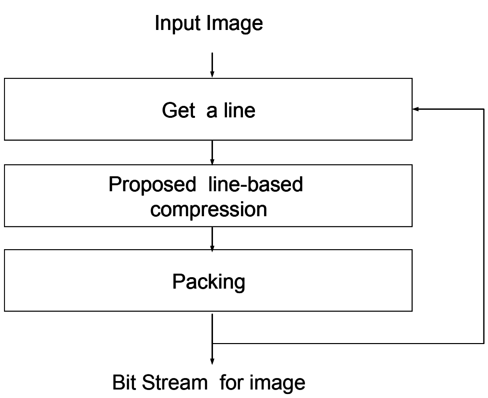


Fig. 1. Basic concept of the proposed line-by-line method.

II. PROPOSED COMPRESSION ALGORITHM

In this section, we present a new low complexity and lossless frame memory compression algorithm for display devices. From the next subsection, the proposed method and its architecture will be introduced.

A. Overview of Proposed Algorithm

Fig. 1 and 2 show the proposed algorithm and the compression process details are as follows. A line of the input image is the input of the algorithm and the final output is a compressed bitstream segment. As the algorithm is shown in Fig. 2, an input line is divided into blocks of length 8. And then the 8-point modified Hadamard transform (MHT) is applied to each block and it provides the DC and AC components for each block. After MHT, all DC components are grouped together for the second MHT. The DC components from the second MHT are directly packed into bitstream. All AC components are coded by the proposed Adaptive Golomb-Rice (AGR) coder. Before all AC components are to be coded by AGR, they should be remapped by the Rice mapping in Fig. 2. Then the packing process produces the compressed bit segments. In the following subchapters, the proposed algorithm is discussed in detail.

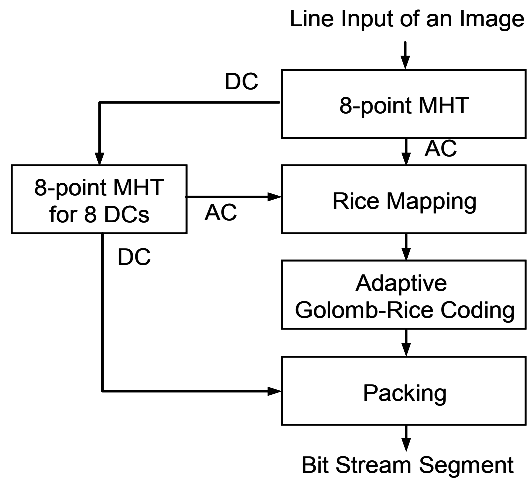


Fig. 2. Generic Diagram of the Proposed Method.

B. Image Input

Basically, the proposed algorithm uses a line-by-line input image, instead of using block by block image input which is normally used in most of image compression methods. The purpose of using a line by line process is because our main idea of this paper is to apply our proposed algorithm into display devices where a line by line process is much easier to be accepted than a block by block process.

Fig. 1 shows the basic process of the proposed line by line algorithm. Moreover, Fig. 2 indicates that a line image is divided into blocks of length 8. Therefore, the basic input unit for the proposed method is a block of length 8. For example, if the input image is 512x512, the number of input lines is 512 and each line is divided into 64 (=512/8) blocks. The reason we need a 8-point block as an input is because MHT is carried out in the unit of 8 pixels in the proposed method. If MHT is changed in the unit, the size of input blocks should be changed.

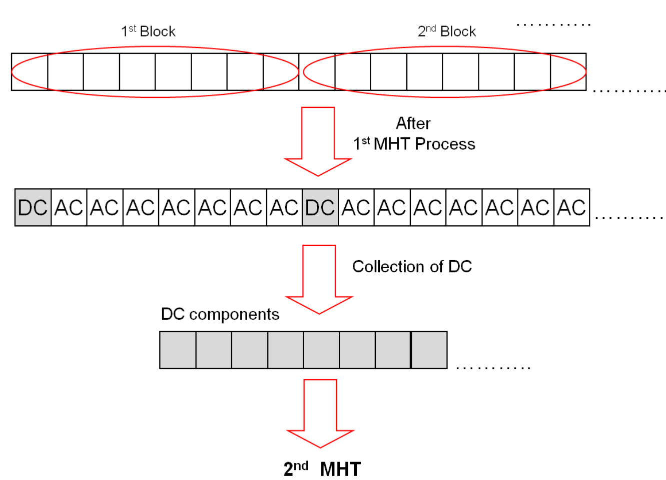


Fig. 3. Details of subband decomposition of the proposed algorithm

C. Modified Hadamard Transform

As a decorrelation process, we employ the modified Hadamard transform (MHT) [6] which is proposed by one of authors of the paper. The MHT can be composed of adders and 1-bit shift right operations, which are very useful in hardware implementation. Fig. 4 illustrates the signal diagram of the 8-point MHT, where dashed lines represent sign-inverting and SR stands for 1-bit shift right.

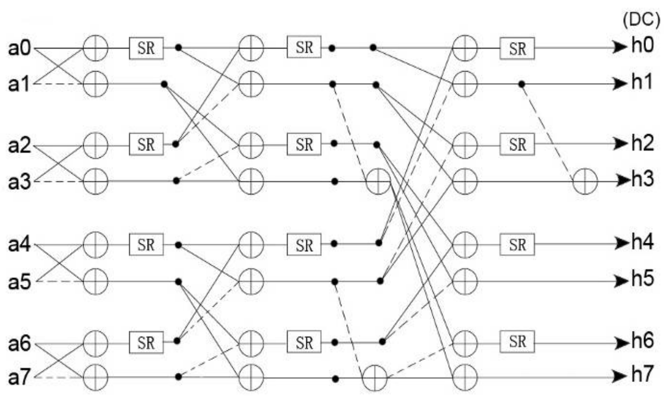


Fig. 4. Signal flow of 8-point MHT. Dashed lines represent sign-inverting and SR stands for 1-bit shift right.

D. Subband Decomposition

To increase the efficiency of image compression, we utilize the MHT for subband decomposition. This is illustrated in Fig. 3 and 5. As is shown in Fig. 3, we simply assemble DC components together and apply the second MHT to the DC data. Also, this is considered to be two-level decomposition as illustrated in Fig. 5.

It has been shown that DC components are still highly correlated. Thus the second MHT for DC data provides further decorrelation between them. According to simulation results we discussed in chapter III, the subband decomposition with the MHT shows its efficacy.

E. Rice Mapping

A Golomb-Rice coder accepts only a non-negative number as its input. However, the transform coefficients can be a negative number. Thus, it is required to convert the transform coefficients into a non-negative number. This is usually accomplished by a mapping as follows:

$$v = \begin{cases} 2c, \; c \geq 0 \\ -2c-1, \; c < 0 \end{cases} \tag{1}$$

where c represents the transform coefficients and v denotes the value of the input of the Golomb-Rice coder.

The inverse mapping from the value v back to c is simple because the LSB of v indicates the sign of c and the magnitude of c is simply obtained by the 1-bit shift right operation with respect to v.

F. Adaptive Golomb-Rice Coding

We employ adaptive Golomb-Rice (AGR) coder for an entropy coder. Adaptive Golomb-Rice coding (also called dynamic Golomb-Rice coding) is an adaptive coding technique based on Golomb-Rice coding.

Golomb-Rice (GR) coders have been used widely in modern compression systems such as JPEG-LS [8] and H.264 [3]. The motivation is that GR coders are optimal or nearly optimal for integer sources with two-sided geometric distributions [10]. In other words, GR coders approximate optimal Huffman coders for such sources, with minimal loss in efficiency. The main advantages of GR coders are that they require less computational power than Huffman coders and no ROM memory for encoding tables and can be easily implemented in hardware systems. Moreover, GR coders can be easily utilized in an adaptive fashion whereas adaptive Huffman coders requires very large amount of computational power.

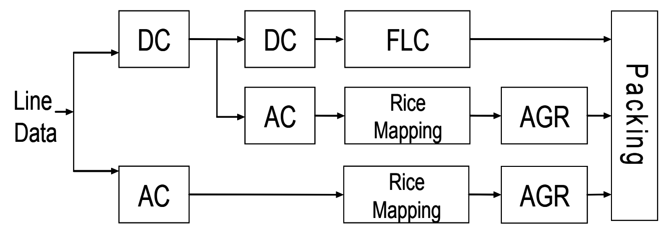
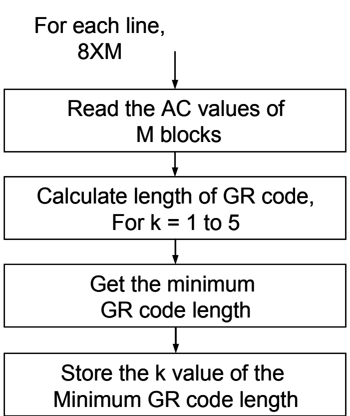
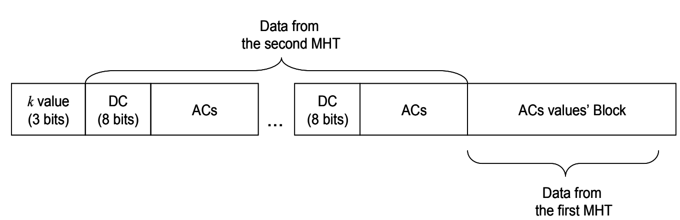


Fig. 5. Subband decomposition and the entropy coding in the proposed method. DC data are packed by the FLC (fixed length coder) and AC data are mapped by the Rice mapping and coded by the AGR coder.



(a)



(b)

Fig. 6. (a) AGR coding flow of the proposed method and (b)Bit packing format for each line data.

The GR scheme is a generalization of the earlier scheme of the Rice coding although each scheme was invented independently. Specifically, the Rice coding is equivalent to the Golomb coding when the tunable parameter is a power of two. This makes Rice codes convenient for use on a computer since multiplication and division by 2 can be implemented using a bit shift operation, which can be performed extremely quickly. Similarly, calculating the corresponding remainder can be achieved by a simple bit mask operation, which is considered to be a very fast operator. Consequently, Rice coding is used as an entropy encoding stage in a number of lossless and lossy image compression methods.

Fig. 6(a) shows the flow of the adaptive Golomb-Rice coding for each input line of the proposed algorithm. The proposed method uses the AGR coding at each line of the input image. In other words, each line of the image has different *k* value to get the maximum compression performance. To determine the optimal parameter *k*, the lengths of GR coded bits for all AC components are summed up. The length of the GR code for a given value *v* is calculated by

$$length_{Golomb} = k + 1 + v / 2^k \, . \tag{2}$$

Then the best *k* value is determined based on the minimum sum of the GR code lengths. For a small value, a smaller *k* leads to a smaller Golomb-Rice code length. As value increases, a larger *k* may produce a smaller code length. Thus, the choice of *k* depends on value.

G. Packing

The AGR codes for AC components are packed as bit-segment as shown in Fig. 6(b). The best *k* value for each line of the input image is coded by the 3-bit fixed length coding (FLC) and stored in the leftmost position. Next, the DC and AC components from the second MHT process are stored with the 8-bit FLC and the AGR coder, respectively. The process is repeated for the remaining blocks of the 2nd MHT process. Once all the DC and AC components of the 2nd MHT process are packed in the bitstream, the remaining AC components from the first MHT process are coded and appended. The resulting format is illustrated in Fig. 6(b).

III. Simulation and Discussion

To show the validity of the proposed algorithm, two existing methods of low complexity and lossless compression are compared with our algorithm. The existing methods are the differential of adjacent pixel (DAP) with Huffman coding as proposed in [3] and the modified Hadamard transform (MHT) with Golomb-Rice coding as proposed in [1]. However, the algorithm which used the MHT and GR coding as proposed in [1] is a lossy compression, in order to make the algorithm proposed in [1] be lossless compression, the quantization process in the proposed algorithm is omitted in our simulation.

First, we simulate the proposed algorithm with only one MHT and AGR coding. After that, we apply the subband decomposition with the MHT and AGR coding to show that the efficacy of each module in the proposed method. Then, we compare the results with the reference algorithms. Also, we analyze computational complexity and memory requirement for the reference algorithms and the proposed algorithm. The comparison results of the performance of each algorithm are shown in the Table I. In order to obtain an enough empirical evaluation of the proposed low complexity and lossless compression algorithm, we use a set of well-known images for experiments. In our experiment, we use twelve 512x512 well-known images which are listed in the first column of Table I. The average of the compression ratio for each algorithm is calculated. As a measure of the performance of compression, we define a compression percentage as

$$Compression\,Percentage = 100(1 - \frac{compressed\ file\,size}{original\ file\,size})\% \tag{3}$$

TABLE I.
COMPARISON RESULTS OF REFERENCE ALGORITHMS AND PROPOSED ALGORITHMS.

Image Name	DAP+ Huffman [1]	MHT+GR [2]	MHT+AGR (Proposed Algorithm 1)	2MHT+ AGR (Proposed Algorithm 2)
Lena	31.6%	35.5%	41.4%	47.3%
Pepper	27.7%	27.7%	35.2%	40.2%
Airplane	37.5%	29.7%	35.5%	41.4%
Target	45.3%	19.5%	21.9%	27.7%
Baboon	8.2%	10.2%	16.8%	23.0%
Elaine	21.9%	26.6%	33.6%	40.2%
Barbara	18.8%	23.8%	31.3%	36.3%
Zelda	32.4%	37.5%	46.5%	53.1%
Man	18.75%	20.3%	26.6%	31.6%
Houses	17.2%	12.5%	19.9%	21.9%
Clown	32.4%	31.6%	40.2%	42.2%
Lighthouse	20.7%	20.3%	27.3%	29.3%
Average	**26.0%**	**24.6%**	**31.4%**	**36.2%**

Table I indicates that the two proposed methods are superior to the two existing methods in terms of coding efficiency. We can observe an average of 5% increment in terms of the compression percentage for the proposed algorithm 1 as compared to both the reference algorithms. Besides, Table I shows us 10.2% of improvement in the compression percentage of the proposed algorithm 2 as compared to the reference algorithm 1 and 11.6% of improvement in the compression percentage of the proposed algorithm as compared to the reference algorithm 2.

Although Table I shows the performance of these algorithms on the same set of images, such evaluation may not

TABLE II.
COMPUTATIONAL COMPLEXITY AND MEMORY SIZE REQUIREMENT ANALYSIS FOR REFERENCE ALGORITHMS AND PROPOSED ALGORITHMS

Module	Reference Algorithm 1&2			Proposed Algorithm 1&2		
	Technique	Computational Complexity (Add or Sub operation/pixel)	Memory Requirement	Technique	Computational Complexity (Add or Sub operation/pixel)	Memory Requirement
Decorrelation	DAP	1	8 line buffer	1st MHT	6	1 line buffer
	MHT	6	1 line buffer	2ed MHT	6/8	1 line buffer
Entropy Coder	Huffman Coder	Memory Access Operation	Depend on hardware (ROM required)	AGR	6	N/A
	GR Coder	N/A	N/A	AGR	6	N/A

justify the actual capability of methods in real-life application. There are a few factors that have major influence on the outcome of the assessment of these algorithms. For example, image or content dependency is one of the factors that affect the performance outcome of each algorithm. From Table I, we can observe that the performance of all algorithms depends noticeably on the image content.

Finally, from Table I, the average compression ratio for each algorithm is calculated. It can be seen that the proposed algorithm 2 gives better compression ratio compared to the proposed algorithm 1 where the proposed algorithm 2 achieved a 10.2% and 11.6% improvement respectively in the compression percentage as compared with both the reference algorithms.

Table II shows the analysis of computational complexity and memory requirement for the reference algorithms and the proposed algorithms. From Table II, the computational costs required by the proposed algorithms increase as compared to the reference algorithms; however, the reference algorithm 1 requires more memory as compared to our algorithms. Especially, the reference algorithm requires ROM for Huffman coders. The use of ROM may be avoidable in the area of system on chip. The exact time required for each operation is hardware dependent.

As the entropy process of the reference algorithm 1, memory access operation may require complex design and implementation. Besides, the more memories are required, the more costly the algorithm will be. The dependence of the computational cost for the reference and proposed algorithms are based on the memory size required.

Therefore, the reference algorithm 1 is more costly because it requires a lot of memories as compared to our proposed algorithms. Our algorithms require only add, sub and shift operations which are simple to be implemented in hardware. As for reference algorithm 2, our proposed algorithm 2 outperform the reference algorithm 2 by 11.6% in the compression percentage and only need extra 6 operations per pixel which do not influence much on the operation complexity.

IV. CONCLUSION

In this paper, a new low complexity and lossless frame compression algorithm has been presented. We have selected two of the low-complexity and lossless algorithms for comparison. The main purpose of the proposed compression algorithm is to get the lowest complexity and lossless compression algorithm for LCD devices. To achieve the aim, we proposed subband decomposition with the modified Hadamard transform and adaptive Golomb-Rice coding. To evaluate the proposed method, we have compared the algorithms using twelve well-know images and calculated the average compression ratio to obtain a fairer and more effective evaluation. The simulation results indicate that the proposed algorithm provides better compression ratio compared to the reference algorithms and it can be applicable to most of the hardware devices with no image quality degradation and little hardware modification. Besides, the new proposed algorithm does not need any memory operation to store data so in this case our proposed method is able to minimize the cost for hardware implementation. In future, the proposed algorithm can be applied to real hardware implementation so that real world testing can be formed to get the exact complexity performance and exact time required for the proposed algorithm. From the simulation results, future research may be directed to improve the proposed algorithm by using the color buffer compression as proposed by [4] or other low complexity algorithms for better compression performance.

REFERENCES

[1] T. Y. Lee, A new frame-recompression algorithm and its hardware design for MPEG-2 video decoders, *IEEE Transactions on Circuits and System for Video Technology* Vol. 13, No.6, , pages 529-534, June 2003.

[2] T. Song and T. Shimamaoto, "Reference frame data compression method for H.264/AVC," *IEICE Electronics Express Vol. 4, No.3,* , pages 212-126, February 2007.

[3] Y. Lee, C.-E. Rhee and H.-J. Lee, "A new frame recompression algorithm integrated with H.264 video compression," *Proc. IEEE International Symposium on Circuits and System*, pages 1621-1624, May 2007

[4] J. Rasmusson, J. Hasselgren and T. Akenine-Moller, "Exact and error-bounded approximate color buffer compression and decompression," *Proc. the 22nd ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*, pages 41-48, August 2007

[5] T. L. B. Yng, B.-G. Lee and H. Yoo, "Low complexity, lossless frame memory compression using modified Hadamard transform and adaptive Golomb-Rice coding," *Proc. IADIS Computer Graphics and Visualization 2008*, July 2008, Amsterdam, Neterlands. Submitted for publication.

[6] H. Yoo, J. M. Jo and J. C. Jeong., "A hierarchical lossless Image compression based on modified Hadamard Transform," *Proc. 10th Workshop on Image Processing and Understanding*, pages 516-520, 1998

[7] I. Furukawa, M. Nomura, and S. Ono, "Hierarchical coding of super high definition images with subband+multistage VQ", *Proceeding ICASSP*, Toronto, Canada, pages 765-768, 1991

[8] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Trans. Image Processing*, vol. 9, pages 1309-1324, Aug 2000

[9] H. S. Malvar, "Fast progressive wavelet coding", *Proc. Data Compression Conf.* , Snowbird, UT, pages 336-343, March 1999

[10] N. Merhav, G. Seroussi, and M. J. Weinberger, "Optimal prefix codes for source with two-sided geometric distributions," *IEEE Trans. On Information Theory*, Vol. 46, pages 121-135, Jannuary 2000

Teresa Liew Bao Yng received the B.S. degree in Information Technology from Multimedia University, Malaysia in 2004. She is currently studying for Master Degree in Engineering in Dongseo University, Busan, South Korea. Her research interests are in image processing, image compression, interlacing and deinterlacing.

Byung-Gook Lee received his BS degree in mathematics from Yonsei University, Korea, in 1987, and his MS and PhD degrees in applied mathematics from Korea Advanced Institute of Science and Technology (KAIST) in 1989 and 1993, respectively. He worked at the DACOM Corp. R&D Center as a senior engineer from March 1993 to February 1995. He has been working at the Dongseo University since 1995. He is currently an assistant professor in the Division of Computer Information Engineering at Dongseo University, Korea. His research interests include computer graphics, computer aided geometric design, and image processing.

Hoon Yoo received his BS, MS, and Ph. D degree in electronic communications engineering from Hanyang University, Seoul, Korea, in 1997, 1999, and 2003 respectively. From 2003 to 2005, he was with Samsung Electronics, Korea where he was involved in the world's first development of the satellite DMB (digital multimedia broadcasting) and the terrestrial DMB on mobile phones. He submitted 23 domestic or international patents in the period of the development. Since 2005, as an assistant professor, he has been with Dongseo University, Busan, Korea. His research interests are in the area of image interpolation, integral imaging, image compression, digital signal processing.