Extended Abstracts of the 2013 International Conference on Solid State Devices and Materials, Fukuoka, 2013, pp136-137

FPGA Implementation of 60-FPS QVGA-to-VGA Single-Image Super-Resolution

Satoshi Chikuda, Takanori Ohira, Yuki Sanada, Masaki Igarashi, Masayuki Ikebe, Tetsuya Asai, and Masato Motomura

Graduate School of Information Science and Technology, Hokkaido University, Kita 14, Nishi 9, Kita-ku, Sapporo, 060-0814 Japan.
Phone : +81-11-706-7689, Fax : +81-11-706-7890, E-mail : chikuda@impulse.ist.hokudai.ac.jp

Abstract

Recently, a novel algorithm of filter-based single-image super resolution (SR) has been proposed [1]. We here propose a hardware-oriented image-enlargement algorithm for the SR algorithm without using frame buffers, and exhibit novel circuits of the proposed enlargement algorithm and the SR algorithm, including extensive demonstrations on FPGA, aiming at the development of single-image SR module for practical embedded systems.

1. Introduction

Super resolution (SR) techniques, which increase resolution of images, are of importance to digital video providers who transcode existing low-resolution media (*e.g.*, DVD media) on high-resolution displays (*e.g.*, 4K displays). Recently, Gohshi *et al.* proposed a novel algorithm for single-image SR [1] where resolution of input images is increased by simple nonlinear filters. The algorithm is partly suitable for hardware implementation because it requires no iterations (and thus no frame buffers), while exhibiting drastic performance as compared with performances of conventional interpolation-based SR algorithms, by reproducing the frequency spectrum exceeding the Nyquist frequency. In the algorithm, the Lanczos filter was employed for "enlargement" of input images, however, upon the hardware implementation, the filter requires many floating operations on wide filter kernels [2]. Therefore, we propose a novel enlargement algorithm based on box filtering that requires integer operations only between a small number of line buffers, while keeping almost the same enlargement quality as Lanczos 2. Furthermore, we exhibit novel circuits of the proposed enlargement algorithm and Gohshi's SR algorithm, and show the experimental results on FPGA.

2. Enlargement Algorithm based on Box Filtering

Figure 1 shows concepts of our enlargement algorithm. An input image ($N \times N$; $N = 3$ in this example) is enlarged twice by upsampling with bilinear interpolation. Then, the enlarged image (4x; $4N \times 4N$) is given to both a box filter and a normalization units. The box filter performs burring to atteneute jaggies in the enlarged image [3]. Edge refinement of the box-filtered image is performed based on the normalized data (local max and min data). Finally, the output image is obtained by downsampling, and the resulting image size is $2N \times 2N$ (2x). It should be noticed that the model does not require any iterative operation.

Edges of box-filtered image are refined by conventional contrast enhancement based on normalization using maximum and minimum values in a local domain. Finally, the edge-refined image is downsampled, and the resulting image is obtained as 2x enlarged image.

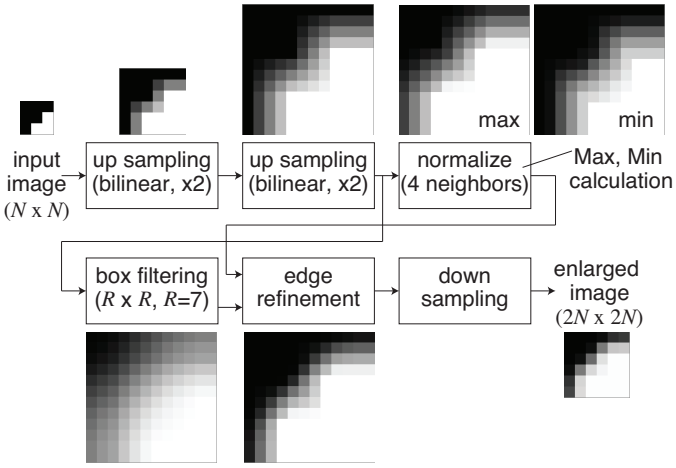

Fig. 1. Processing flow of proposed enlargement algorithm.

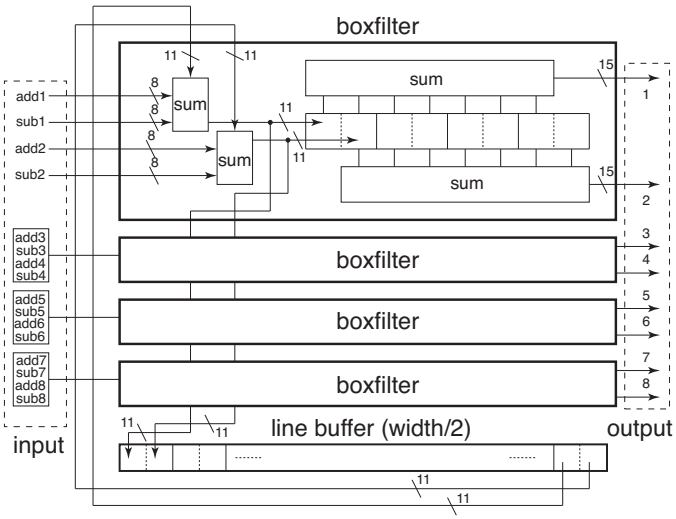

Fig. 2. Schematics of proposed parallel box filter circuit.

3. FPGA Implementation

Our enlargement circuit consists of conventional up-sampling blocks for 4x bilinear enlargement, 7x7 box-filter blocks for the burring, local max/min blocks for conventional edge refinement, and conventional down-sampling blocks. Figure 2 illustrates the proposed 7x7 box-filter block where sub_i and add_i represent the sum of pixel values being used (added to or subtracted form the line buffer [4]) to update the convoluted pixel value, and outputs 1-8 represent the parallel

TABLE I
FPGA Implementation and Performance Summary

Input Res. (MAX)	Output Res. (MAX)	Depth	ALUT&ALM counts	Register counts	FPGA clock	VSYNC
320x240	640x480	8-bit gray	21,126	33,648	90 MHz	60 Hz

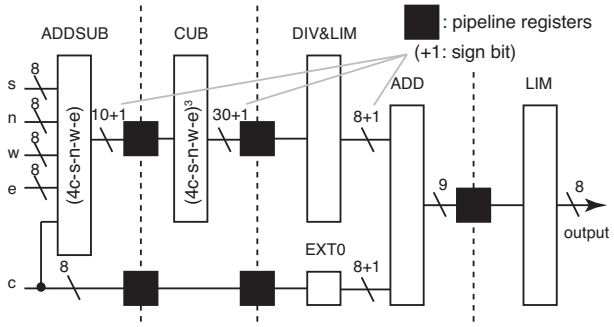

Fig. 3. Super resolution filter based on Gohshi's model.

output streams.

The enlarged streams are given to SR kernel decoders [4] that extract north (n), south (s), east (e), west (w), and center (c) pixel values from the input stream being synchronous to a pixel-data transfer clock. The circuit also implements pixel counters to detect the vertical (s-n) and horizontal (e-w) boundaries (obeying the Neumann boundary). The extracted pixel values (s, n, w, e, c) are given to a pipelined SR filter circuit (Fig. 3), where ADDSUB module detects spatial edges, CUB module enhances the edges, DIV&LIM module compresses the enhanced edge and limits the compressed edges, ADD module sums the limited-and-compressed edges and sign-extended c values, and LIM module limits the summed value within the output bit width (8).

4. Experimental Results

We implemented the proposed circuits on a commercial FPGA (MMS Co., Ltd., PowerMedusa, MU300-DVI, Altera Stratix II). The circuits shown in Figs. 2 and 3 were coded by VHDL, and were synthesized and place-and-routed by Quartus II. An input VGA image (640x480) on a laptop PC was given to the input DVI port of the FPGA board. The board captured 320x240 pixels (QGVA) in the input VGA images at 60 fps, and enlarged the QVGA images to VGA resolution. The enlarged images were processed by SR filter circuits on the same board. The processed SR images were exhibited on a separate monitor through the output DVI port. Then, the processed SR images were transmitted to PC via commercial DVI/HDMI capturing products (Elgato Game Capture HD). Figure 4 shows our experimental sets. The input and processed SR images are shown in Fig. 5 top left (200x200 of 320x240) and right (400x400 of 640x480), respectively. The image was flatten while the edges were clearly kept (Fig. 5 right). Table I summarizes specification and performance of the SR circuits on FPGA. All the line buffers were implemented by FFs of the FPGA. The the number of registers listed in Tab. I includes registers in both primary circuits and line buffers.

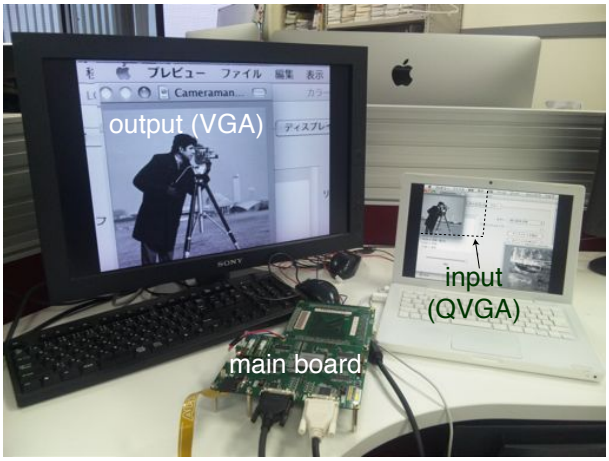

Fig. 4. Experimental sets.

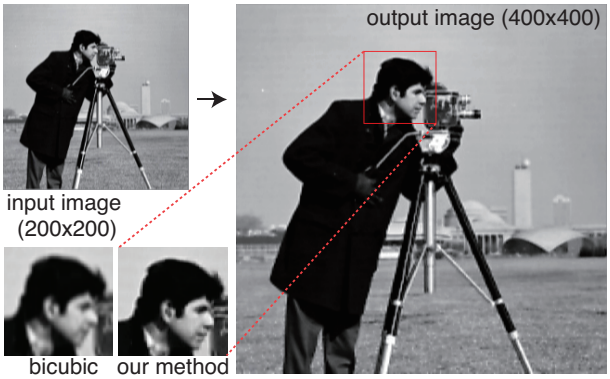

Fig. 5. Demonstration of proposed super resolution filter (experimental).

5. Summary

We implemented an algorithm of single-image super resolution (SR) [1] on FPGA where a novel hardware-oriented enlargement algorithm was employed. Although the proposed architecture has not been optimized well, one may further reduce the number of line buffers, by considering interfaces between the enlargement and SR blocks. Line buffers in the kernel decoder may be shared by an output line buffer in the last stage of the enlargement circuit.

References

[1] S. Gohshi, "A new signal processing method for video —Reproduce the frequency spectrum exceeding the Nyquist frequency—," Proc. 3rd Multimedia Systems Conf., pp. 47-52, 2012.

[2] http://en.wikipedia.org/wiki/Lanczos_resampling

[3] M.J. McDonnell, "Box-filtering techniques," Computer Graphics and Image Processing, **17**(1), pp. 65-70, 1981.

[4] Sanada Y., Ohira T., Chikuda S., Igarashi M., Ikebe M., Asai T., and Motomura M., "FPGA implementation of single-image super resolution based on frame-bufferless box filtering," 2013 RISP International Workshop on Nonlinear Circuits, Communications and Signal Processing, The Island of Hawaii, USA (Mar. 4-7, 2013).