

Image-Scan Video Segmentation Architecture and FPGA Implementation

T. Morimoto, H. Adachi, K. Yamaoka, K. Awane, T. Koide, and H. J. Mattausch

Reserch Center for Nanodevices and Systems, Hiroshima University, 1-4-2, Kagamiyama, Higashi-Hiroshima, 739-8527, Japan

Phone: +81-82-424-6265 Fax: +81-82-424-3499 e-mail: {koide, hjm}@sxsys.hiroshima-u.ac.jp

1. Introduction

At present, block-based image compression such as JPEG or MPEG-2 is the mainstream of image compression methods. If the compression ratio of each object in the image can be individually determined, more efficient image compression, so called object-based image compression [1], becomes possible. In addition, there are a variety of object-based image processing applications like object recognition, object tracking, or the new MPEG-7 standard [2]. In all of these applications, image segmentation is an indispensable first-step for object extraction.

Many image segmentation algorithms have already been proposed [3, 4]. However, the majority of these algorithms is designed for software implementation and has large complexity, so that real-time processing (30 fps) with a general-purpose processor is difficult. To satisfy real-time requirements, a pixel-based fully-parallel video segmentation LSI architecture for QVGA size images, which realizes over 300 fps video segmentation, have been reported [5]. However, this fully-parallel approach has much higher performance than required by standard video frame rate applications (30 fps) and consumes a relatively large area. Therefore, a reduction of the implementation area by relaxing the timing constraints is desirable. To overcome this, we have developed an image-scan video segmentation architecture for standard frame rate video applications and verified its sufficient performance with an FPGA-based prototype system for 80×60 video images (30 fps).

2. Image-Scan Video Segmentation Architecture

A conceptual diagram of the image-scan video segmentation architecture by region growing is shown in Fig. 1. An input image (e.g. 6×10 pixels) is divided into smaller image blocks (e.g. 6×2 pixels). The pixels in each block are processed in parallel according to the previously reported region-growing algorithm [5] with a small image-segmentation processing array of the block size and all blocks are sequentially scanned in a cyclic mode from top to bottom of the image. Between the processing steps of two blocks the processing results of the finished block are stored in a memory and the processing status of next block, obtained in the previous scan of the image, is loaded into the processing elements of the array. These storing and loading steps require a memory with very high access bandwidth which is realized by using on-chip memory with large word length. Naturally, the block size is variable, and a trade-off can be exploited to optimize processing time (video frame rate) versus hardware amount.

The image-scan approach is implemented with an architecture consisting of an embedded memory part and a processing part as shown in the block diagram of Fig. 2. The processing-part is further subdivided into *weight calculation circuit* and *image segmentation unit*. The weight calculation circuit calculates the connection weights from the pixel data (e.g. RGB values) of the input image, which stand for the similarity between neighboring pixels. The image segmentation unit is an $m \times 2$ array of image segmentation

elements (ISE), each corresponding to a processing element for one pixel. The image segmentation unit executes the scan-mode region-growing by sequential processing of the image blocks. For this purpose the previous pixel states and the connection weights are first loaded from the memories and after processing the new pixel states are stored again to enable continuation of the block processing during the next scan. Loading and storing operations interact with the memory part. In the image segmentation unit, a seed pixel (leader cell) for region-growing is first selected and then region-growing from this seed pixel is carried out with the ISE-array in parallel. After several scans over all blocks, when the current region cannot grown anymore, this region is defined as one segment and a region-identifying number (label) is assigned to all of its pixels. Then the next seed pixel is searched and used for a new region-growing process. In the described way, all pixels are classified into meaningful regions.

In the memory-part there are five kinds of memories, which are called *excitation flag*, *segmented flag*, *leader cell flag*, *connection-weight* and *label number memory*, respectively. The first 3 memories store the intermediate pixel states during the region-growing processes and together with the connection-weight memory are interconnected with the image segmentation block as shown in Fig. 2. All status and connection-weight data for segmentation processing of $m \times 2$ pixels are stored under the same address in each memory, so that they can be accessed easily and simultaneously in one clock cycle.

3. FPGA-Based Video Segmentation System

For the verification of the proposed architecture, we have constructed an FPGA-based real-time image segmentation prototype system with a video camera and a display, as shown in Fig. 3. The segmentation quality and remaining insufficiencies of the proposed architecture can be analyzed more quickly by using this system and in particular the visual inspection is more practical than circuit simulation.

The input-image size to the FPGA-implemented image-scan, region growing segmentation circuit is 80×60 pixels (30 fps). The usage of logic elements and on-chip internal memory of the main FPGA (EP1S60 [6]) is about 48% and 2%, respectively, which includes the complete image-scan image segmentation circuit, three memory synchronizers, an address generator, and an active memory selector. The specification of the segmentation system is summarized in Table I. Figure 4 shows some segmentation examples which compare simulated segmentation results by MATLAB and segmentation results obtained with the FPGA-based prototype system. From these results it can be seen that correct segmentation results are achieved with the developed prototype system.

The trade-off characteristic of the proposed image-scan architecture with respect to the processing time and the number of processing elements is shown in Fig. 5. From this graph, it can be seen that about 640 (320×2) ISEs, operated at 12 MHz clock frequency, are required for

the real-time image segmentation (30 fps) of QVGA size images. We have estimated that the resources of the latest FPGA generation (e.g. Altera EP2S180 [6]) are sufficient to implement this image-scan realization of QVGA-size image segmentation. An ASIC implementation is of course also possible by simply using the Verilog-HDL source of the FPGA implementation for the synthesis of the ASIC layout.

4. Conclusion

An image-scan architecture for video segmentation by region growing has been presented and the effectiveness of the proposed architecture has been confirmed by an FPGA-based prototype system for 80×60 pixels video images (30 fps). FPGA-based segmentation of QVGA-size images is estimated to be possible with the latest FPGA generation at the low clock frequency of 12 MHz. The next development steps include the realization and performance evaluation of an ASIC implementation of the proposed image-scan

architecture.

Acknowledgments

Part of this work was supported by a the 21st Century COE program "Nanoelectronics for Tera-bit Information Processing," a Grant-in-Aid for young Scientists (B) (No.16700184), Ministry of Education, Culture, Sports, Science and Technology, Japanese Government and a Grant-in-Aid for JSPS Fellows, 1650741, 2005.

References

- [1] T. M. Strat, Proc. of Work Shop and Exhibition on MPEG-4, pp. 53 - 57, 2001.
- [2] The homepage of the Moving Picture Experts Groups, 2006, URL: <http://www.chiariglione.org/mpeg/>.
- [3] J.C. Russ, "The Image Processing Handbook," pp. 371 - 429, CRC PRESS, 1999.
- [4] B. Jähne, "Digital Image Processing, 5th revised and extended edition," pp. 427 - 440, Springer, 2002.
- [5] T. Morimoto et al., IEE Proc. Cir. Dev. & Sys., vol. 152 (6), pp. 579-589, 2005.
- [6] Altera Corporation, 2006, URL: <http://www.altera.com/>.

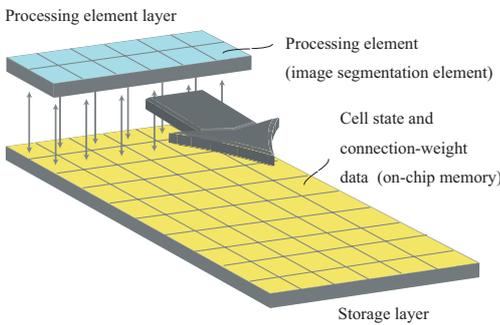


Fig. 1 Conceptual diagram of image-scan video segmentation. High access-bandwidth is needed between the processing element layer and storage layer for real-time image processing. In this example, 6×2 processing elements are used for segmentation of a 6×10 pixel image.

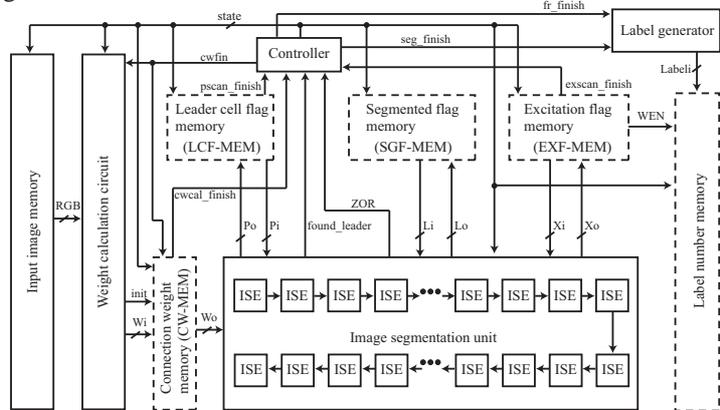


Fig. 2 Block diagram of the row-scan based image segmentation architecture. The processing part is indicated by full-line boxes and the memory part is indicated by dashed-line boxes.

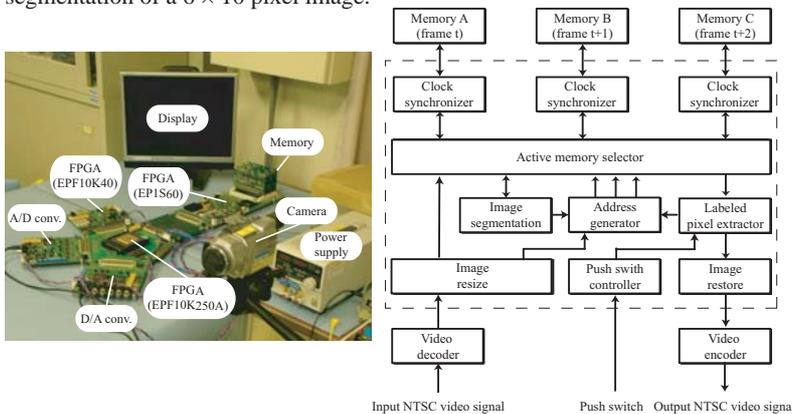


Fig. 3 Left picture shows our video segmentation prototype system. The block diagram of this segmentation evaluation system is shown in the right figure.

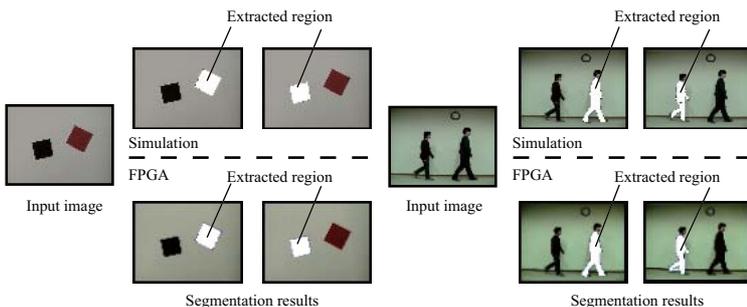


Fig. 4 Examples of segmentation with developed prototype system (FPGA) and MATLAB simulation.

Table I Specification of the developed system.

Input/output format	NTSC Y/C signal
Number of pixels (image segmentation)	4,800 pixels (80 x 60 pixels)
Number of image segmentation elements	160 ISE (80 x 2 ISE)
Clock frequency	12.27MHz
FPGA devices (Resource usage)	Altera EP1S60F1020C7 (27,929 / 57,120 LEs, 48%, 144,256 / 5,215,104 bits, 2%) Altera EPF10K250AGC599-3 (157 / 12,160 LCs, 1%) Altera EPF10K40RC208-4 (162 / 2304 LCs, 7%)

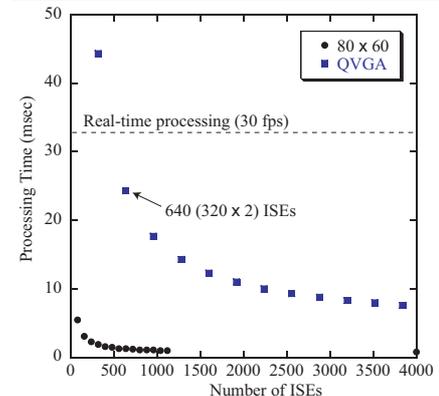


Fig. 5 Processing time and the number of processing elements of the proposed image-scan architecture at an operation frequency of 12 MHz.