Real-time implementation of contextual image processing operations for 4K video stream in Zynq UltraScale+ MPSoC – a demo

Marcin Kowalczyk
AGH University of Science
and Technology Krakow, Poland
E-mail: kowalczyk@agh.edu.pl

Dominika Przewlocka
AGH University of Science
and Technology Krakow, Poland
E-mail: dprze@agh.edu.pl

Tomasz Kryjak, *Member IEEE*
AGH University of Science
and Technology Krakow, Poland
E-mail: tomasz.kryjak@agh.edu.pl

***Abstract*—In this demo a hardware implementation of selected contextual based image pre-processing modules for a 4K@60 fps video stream in a Zynq UltraScale+ MPSoC is presented. Results obtained with Verilog, Vivado HLS and SDSoC with xfOpenCV librarty are presented and compared. Moreover, the scheme for implementing contextual based operations for a video stream in the format of 2 and 4 pixels per clock and challenges related to the pipelined implementation of processing such data are described. The following operations are presented: simple averaging (box filter), Gaussian filter, edge detection using the Sobel and Canny methods, median filter and morphological erosion and dilation operations. All designed modules support the processing of a 3840 x 2160 @ 60 fps video stream.**

***Index Terms*—FPGA, Zynq SoC, image pre-processing, contextual based filtering, real-time processing, Sobel, Canny, Vivado HLS, SDSoC, xfOpenCV**

I. Introduction

The implementation of real-time image processing, analysis and recognition systems using reconfigurable devices has been present in scientific research and industrial applications for almost 30 years. During the period the used resolution changed from 320×240, through 640×480, 1280×720 (High Definition – HD) to 1920×1080 (Full High Definition – FHD). The next step in the development of vision systems is the 3840×2160 pixels @ 60 fps video stream (4K, Ultra High Definition – UHD). In relation to FHD, there is a fourfold increase in the number of pixels and therefore also the pixel clock value increases to slightly less than 600 MHz. This is the "limit" value for today available FPGA devices.

Therefore, it is not longer possible to implement vision system in the well-known serial-pipeline scheme (one pixel is processed in one clock cycle). The solution is a parallel-pipeline scheme, where in one clock cycle more than one pixel is processed. For 4K video stream this equals 2 or 4 pixels per clock (ppc). This allows to lower the pixel clock from approximately 600 MHz to 300 MHz or 150 MHz respectively, which is an "acceptable" value. It should be noted that the parallel-pipeline processing scheme has a fairly significant impact on the implementation of all broadly understood contextual operations – from rather simple filtrations discussed in this paper to such algorithms as connected component labelling, optical flow, stereovision or HOG features computation.

The work presented in this paper was supported by the National Science Centre project no. 2016/23/D/ST6/01389

In this demo the challenges associated with a fully pipelined implementation of contextual operations for a 4K video stream are presented. Three approaches were considered: HDL (Hardware Description Language – implementation in Verilog), HLS (High Level Synthesis – using the Vivado HLS tool) and using the xfOpenCV library and the Xilinx SDSoC tool. The demo setup consists of a HDMI 2.0 signal source (PC or camera), a ZCU102 evaluation board and a monitor.

II. Implementing contextual based filtering for 4K video stream

For a video stream in 2 ppc or 4 ppc format, two or four pixels are available simultaneously at the input of the contextual operation module. To process the data in real time, 2 or 4 pixels should also be provided at the output (after a certain number of clock cycles – latency). Therefore, two or four contextual operations should be carried out in parallel. The following considerations are presented for a 3×3 context, but quite straightforward they can be generalized to $N \times N$ (assuming that N is odd).

The context generation scheme for the 2 ppc format is shown in the Figure 1. It is similar to the well-known delay line or circular buffer scheme, with the difference that a single "element" represents two consecutive pixels from the input image. The 3×3 context consists of 18 pixels, of which 12 are used for calculations – they form the two considered contexts. With parallel access to contexts, any operation can be performed (for example, Gauss filtering) – in a manner known from 1 ppc processing. Finally, at the output 2 new pixel values are provided. For the 4 ppc case, the solution is analogous, the context contains 36 pixels of which 18 are used and 4 pixels are at the output.

Passing a video stream thorough the used Zynq SoC device involves the use of the following modules: Video PHY Controller, HDMI 1.4/2.0 Receiver Subsystem, HDMI 1.4/2.0 Transmitter Subsystem. This modules allow to read video stream into the FPGA logic and write video stream to output.

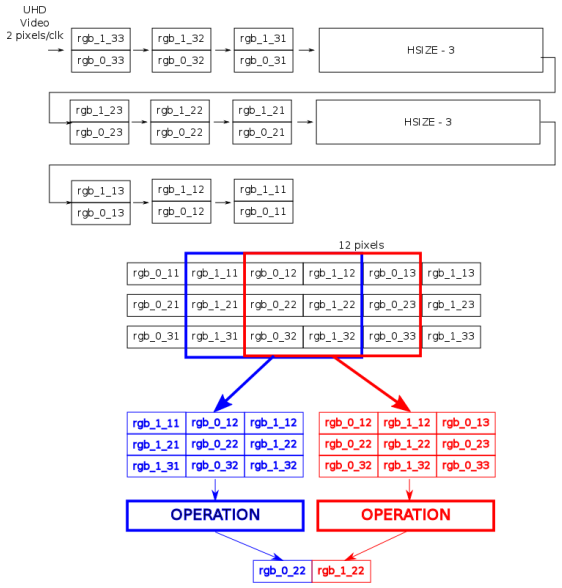

Fig. 1. 3×3 context generation for 2ppc video stream

TABLE I
RESOURCE UTILIZATION FOR THE FPGA PART OF THE WORKING SYSTEM WITH SOBEL FILTRATION ON ZYNQ ULTRASCALE+ MPSOC – 4 PPC FORMAT. ALL NUMBER TAKEN FROM THE POST PLACE REPORTS. THE NUMBER IN BRACKET SHOWS % OF RESOURCES AVALIABLE ON THE DEVICE.

Resource	Video pass-through	RGB2GRAY + Sobel
LUT	32025 (11.68%)	32529 (11.87%)
FF	39037 (7.12%)	39909 (7.28%)
BRAM	6 (0.66%)	8 (0.88%)
DSP	3 (0.12%)	23 (0.91%)

As a parameter of this block the number of pixels per clock can be set, as well as type of video interface (Native Video or AXI4-Stream).

Finally, the following modules have been implemented: box filter, gaussian filter, sobel filter, median filter, Canny edge detection, erosion, dilation and median for binary images.

FPGA resource utilization for a simple video pass-through and Sobel filtration with prior RGB to grayscale conversion is presented in Table I.

A. Vivado HLS

In order to evaluate the usefulness of Vivado HLS for 4K video processing two 3×3 contextual operations – Sobel and median – were implemented. The Xilinx Application Note 1167 was used as a reference [1]. Several types and functions from the *HLS Video Library* were utilized in order to create the mentioned modules. It was necessary to use 4 channels with a width of 8 bits in order to read and write data in the 4 ppc format. Therefore, custom HLS functions had to be implemented, as the *HLS Video Library* supports only computations for 1 ppc.

The resource usage for HLS-based Sobel modules is slightly higher than for the Verilog implementation. On the other hand, the HLS median filter is more efficient than the current Verilog version.

B. SDSoC

The SDSoC environment (Software Defined System on Chip) is in assumption a tool for creating hardware-software systems for Xilinx Zynq SoC based systems. One of its components, the reVision stack, provides a support for broadly understood embedded vision systems. It is based on the xfOpenCV library [2], a hardware implementation (in Vivado HLS) of selected functions from the OpenCV library [3] (mid 2018 about 45 functions). During experiments, the usefulness of this tool for 4K video stream processing was evaluated.

A thorough analysis of the Sobel module from the xfOpenCV was made. In the 1 ppc variant, it works with the same clock as the input data, which in this case is received in the 2 ppc format (i.e. 300 MHz). Thus, there is a reduction in the processing frequency and real-time operation, understood as processing the input stream without losing any information, is questionable in this case. In addition, the use of the mentioned buffering scheme leads to a visible delay between the input and output image.

Summing up. XfOpenCV and SDSoC have limited usefulness for systems with a pipeline data processing architecture, where the data is received directly from the video sensor. The use of buffering in external DDR memory is in this case is unnecessary, introduces delays and increases the energy consumption, as well as complexity of the system.

III. SUMMARY

In reprogrammable logic, a 4K@60fps video stream can be processed in 2 or 4 ppc format. The 2 ppc format requires that the modules operate at 300 MHz, which may be a challenge in the case of complex calculations. On the other hand, the use of resources is about twice less than for 4 ppc. The energy consumption for both solutions was tested on an example Sobel filtration. The differences were rather insignificant, however indicated that the 4 ppc solution is more energy efficient.

The implementation of contextual operations requires the use of a modified circular buffer and non-contextual ones such as colour space conversions, gamma correction, LUT coding, binarization to multiply to computational resources (x2 or x4).

The Vivado HLS tool allows to implement operations with more then one pixel per clock. However, it should be noted that the functions available in the HLS Video Library do not support other formats than 1 ppc. Therefore custom implementations are required. The vision system created in the SDSoC/xfOpenCV, due to the architecture that assumes buffering in external RAM, introduces visible delays to the video stream.

REFERENCES

[1] S. Neuendorffer, T. Li, and D. Wang, *Accelerating OpenCV Applications with Zynq-7000 All Programmable SoC using Vivado HLS Video Libraries*.
[2] Xilinx, *Xilinx OpenCV User Guide*.
[3] Opencv. [Online]. Available: https://opencv.org/ (last access 01.06.2018)