Applied Mathematics and Computation 267 (2015) 419–426

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Embedded platform for local image descriptor based object detection

Rafal Kapela *, Karol Gugala, Pawel Sniatala, Aleksandra Swietlicka, Krzysztof Kolanowski

Poznan University of Technology, Department of Computer Engineering, ul. Piotrowo 3A, 60-965 Poznan, Poland

ARTICLE INFO

Keywords:
Hardware accelerators
Image descriptors
Fast-Retina Keypoint
Hamming distance
FPGA
Zynq Z-7010

ABSTRACT

The article presents novel idea of a hardware accelerated image processing algorithm for embedded systems. The system is based on the well known Fast Retina Keypoint (FREAK) local image description algorithm. The solution utilizes Field Programmable Gate Array (FPGA) as a flexible module that is used to implement hardware acceleration of a given part of the image processing algorithm. The approach presented in this paper is slightly different. Since we are using very fast FREAK descriptor it is not our purpose to implement full feature extraction algorithm in hardware but just its most time-consuming part which is brute force matcher based on the Hamming distance. Moreover our goal was to design very flexible system so that the feature detection and extraction algorithm can be replaced without any interruption in the hardware accelerated part.

© 2015 Published by Elsevier Inc.

1. Introduction

The value of the hardware accelerators is rather unquestionable. They have been widely used in every discipline that even in the smallest extent has contact with electronics. The idea of the CPU leaving some of the time-consuming tasks to other unit that specializes in their processing is rather known and not novel. The problem however that is closely related to this field is that they have very narrow range of specializations (i.e., they are fast but not as flexible as CPU) keeps the designers of the hardware IP cores still busy. Some contradiction to this thesis may be the Compute Unified Device Architecture (CUDA) and other graphic-card based solutions but this solutions were never aiming at low-power designs. Quite the contrary – the assumption is made that the power issue is neglected by CUDA solutions. There are of course some mobile CUDA solutions like the recent K1 processor [1] but still it will never be as power efficient as specified Application-Specific Integrated Circuits (ASICs) which are the natural extension of the FPGA designs (i.e., the FPGA netlist can be easily applied to design low power application specific digital circuit). For this reason we do not really focus on the general purpose accelerators but the IP cores designed specifically for a given type of problem.

Due to the interdisciplinary character of this work the remaining part of the Introduction section consists of two paragraphs: one containing the state-of-the art in the local image processing algorithms whereas the latter describes the achievements in the FPGA image and video related applications.

* Corresponding author.
E-mail address: rafal.kapela@put.poznan.pl (R. Kapela).

http://dx.doi.org/10.1016/j.amc.2015.02.029
0096-3003/© 2015 Published by Elsevier Inc.

1.1. Local image descriptors

Local image descriptors are well known and efficient image description techniques since the development of Scale Invariant Feature Transform (SIFT) [2] and Speeded Up Robust Features (SURF) [3]. A very interesting, from hardware implementation point of view, branch of local image description techniques are the descriptors that produce binary patterns as a result. A good examples are Binary Robust Independent Elementary Features (BRIEF) [4], Fast Retina Keypoint (FREAK) [5] or Binary Robust Invariant Scalable Keypoints (BRISK) [6]. The fact, that these algorithms produce binary pattern that can be interpreted in terms of the Hamming distance makes them good candidates for description techniques that can be relatively easy accelerated in hardware. Up to date, multiple designs have been published [7–10]. The unquestionable advantage of all of them is the speed of processing the input image or video frame. This parameter depends on the architecture and obviously the chosen FPGA platform but for most presented systems it is on the level of about 30 ms which makes them capable of processing the 30 fps video in the real-time. The second dominant parameter for this kind of systems is the size of the frame they can describe during mentioned time and it varies from about 320×240 in [7] up to even full-HD video (1920×1080) in [10]. It is worth to mention that the latter system can process full-HD video at 60 fps.

1.2. FPGAs for hardware acceleration of image and video processing

FPGAs are widely considered as accelerators for compute-intensive applications. This is very true especially in the image and video processing fields. This is mainly due to the fact that FPGAs are very flexible in terms of the possible functionalities they may implement. In other words it is up to the designer what the architecture of the system will be. In [11] a good survey on the computing models for FPGA-based accelerators is presented. The authors of this survey claim that FPGAs are good tools for designing high-performance computers (HPC) because of their flexible micro-architecture. The two main features of FPGAs for HPC are highlighted: parallelism (up to 10,000 parallel threads for low-precision computations) and payload per computation. Both equally important but while first is quite obvious and well discussed the latter is usually omitted and needs clarification since it is the main difference that distinguishes hardware from software processing. Payload for computation stands for the fact that the most of the control processes/signaling is implemented into the logic. Thanks to this the designed system do not need to emulate overhead instructions (eg., in loop computations). The paper itself does not present any comparative results but is good inspiration and guide for changing the perspective needed for designing own efficient FPGA solutions. In [12] Lysakov and Shadrin present an FPGA-based hardware accelerator for high-performance data-stream processing. The system is interesting giving the fact that it works similarly to the well known video streaming software – gstreamer library [13] so that it defines the streaming pipeline within which each worker/stage has a separate task to perform. The presented system is capable of processing four independent data streams in parallel. A very similar project but intended for images is presented in [14]. The authors of this project keep the data-set of images on their hard drive and send them to the hardware accelerators implemented in FPGA for further processing. The real nature of the FPGA computing is presented in [15] where FPGA is partially reconfigured dynamically (i.e., parts of the FPGA design are changed on-line during the operating phase of the system) in order to perform different image processing tasks like filtering or binarization. The time needed for reconfiguration is in range of several tens of micro seconds and is usually neglected in comparison to the overall performance of the system. The results achieved by the FPGA system outperform CPU by 100 up to 1000 times. Lastly there are number of papers that present FPGA-based hardware accelerators in many projects like the evaluation of the communication interfaces [16], place and route [17] or even software runtime acceleration [18].

Following sections of the article show in more detail how the system was conceived from both – software and hardware perspectives. We start with a brief overview of the FREAK description algorithm, then in Section 3 we present the software/hardware implementation of the algorithm following with the presentation of the overall architecture of the system and the achieved results. The paper is concluded in the last section.

2. The FREAK algorithm

For the proper logical flow of the article and it is consistency of the this section presents the overview of the algorithm itself. As it was mentioned in the introduction section FREAK belongs to the local image description algorithm genre. It is undoubted advantages that make it applicable for the embedded platform implementations are that it is much faster than the SIFT and SURF algorithms (up to about 140 times [5]) and produces a binary pattern as an output which describes the given patch of the image.

The idea of the description itself is not new and exploits the approach in which the description is made based on the image area that is surrounding the extracted previously keypoint [19]. FREAK is biologically inspired algorithm that follows the hypothesis that the interest area of the image can be described as a difference of Gaussian (DoG) functions. This is very convenient since it allows to compare image regions only based on the filtered luminance signal without any time-consuming preprocessing steps like image enhancement or edge extraction. Given the image patch surrounding the key-point the descriptor is extracted based on the formula 1

$$F = \sum_{0 \leqslant a < N} 2^a T(P_a) \tag{1}$$

Download English Version:

https://daneshyari.com/en/article/6420303

Download Persian Version:

https://daneshyari.com/article/6420303

Daneshyari.com