Real – time system for shape extraction from an image

Laurenţiu M. Ionescu, Alin Gh. Mazăre, Vasilica A. Berechet, Ioan Liţă

Electronics, Computers and Electrical Engineering
University of Pitesti
Pitesti, Romania
laurentiu.ionescu@upit.ro

Adrian-Ioan I. Liţă

Applied Electronics
Politehnica University of Bucharest
Bucharest, Romania
ioan.lita@upit.ro

Abstract— **This paper presents a system of "image sensor" for recognition and counting of shapes present in an image. Result information is provided as identified object, position relative to the lower left corner of the image and the number of objects detected in a certain type of image. In other words, the system functions as a sensor, the image is the entry and the items extracted from it represent the output data. Experiments aimed basic shapes: rectangle, triangle, circle, but the work can be extended to other more complex forms. The system was implemented on a Zynq SOC 7000 (Xilinx) and can be used in many applications involving pattern recognition in real time.**

Keywords— *Real time, shape recognition, FPGA*

I. INTRODUCTION

There is much information that can be extracted from images. Segmentation, morphological operators, labeling and feature extraction of objects of interest represent the typical stages of working with images. Usually, the image processing domain belongs to software applications. The technological development of ASICs and FPGAs has enabled the development of integrated applications that process images and return data like sensors [1]. Their advantage over standard microprocessor based algorithms is that they allow a real time transmission of results associated with images [2]. One possibility, exploited in several scientific papers, is to extract information from images captured by a video camera and use in applications. By implementing the algorithm on DSP, reconfigurable circuits or ASICs [3] we can get real-time information extracted from images.

One research direction is on-site image processing, happening directly in the camera, while transmitting to the base station only desired events, not live video. In order for this purpose to be achievable, the central computer that runs image processing algorithms needs to be replaced by a much smaller integrated system capable of running the same algorithms, in real time for one camera.

The evolution of the technology to build integrated systems made running more and more complex algorithms possible in real time. Image and video processing are not an exception of this. One decade already passed since image processing algorithms were first implemented in reconfigurable [4], both algorithms related to improving the quality of images and identifying objects in images (image processing level 2) and especially those related to the spectral decomposition of the image for analysis of the frequency components [5]. Since the implementation of functional blocks to the implementation of parallel processing architectures with real-time response all it took was only a step. The architectures are based on FPGA circuits - and sometimes also on DSP modules - allowing the implementation of processing blocks which can work in parallel, i.e. integrated facial recognition system [6]. Another important step in implementing dedicated image processing circuits was using artificial intelligence, especially neural networks for pattern recognition of objects in the image [7]. Some IC solutions made in this period (early 2000s) such as adaptive spectral histogram equalization system [8] are currently used in modern cameras. In 2006 one of the first application developed using image analysis to identify certain events related to the movement was integrated into a system of "virtual motion sensor" [9]. Integrated image analysis tools were used not only for images in the visible spectrum but also in ultrasound such as those presented in [10] and [11]. Identifying the analysis of 2D images has applications in many fields, with low environmental impact [12].

Our structure is shown in the figure below. Image acquisition block is the first module. There are two possible channels, in our case. The first is given by an analog acquisition mode, when the camera provides analog signal (SVIDEO for example) and the second is the typical digital serial interfaces: HDMI or USB. From any of the sources, the processor takes the digital serial data and assembles data packets which are then transmitted to FPGA. The interface between processor and FPGA is a high speed serial interface that allows internal transmission of information pixel by pixel. Next, there is an image segmentation block based on a function of the brightness threshold – assuming that the object of interest are separated by lighter colors against darker background. At this stage, the image arranged as a bit array is subject to the extraction of the objects. The extraction procedure is performed using an association technique: templates of the objects are in an associative memory and slices of the picture will be compared with those templates. The comparison is in Hamming distances space - in conclusion will identify not only identical shapes but most close to those sought.

II. System presentation

This section presents the system which extracts objects from images. The section contains two modules: one dedicated to presenting the system as an architecture and the second to presenting the used circuits.

A. System architecture

A system block diagram is presented in figure 1. The system contains an analog acquisition module which reads SVIDEO HD 480p format from the camera. The acquisition module allows a 10bit acquisition at a frequency of 54MHz, based on an IC from Analog Devices: ADV7183B.

Generated serial data sent over a Two Wire Interface to the processor is converted into pixels (raw data) for the other processing modules to use. From this processor the data can also be sent as images on a HDMI interface or over USB (720p 1280x720 format).

Not considering the format, the processor fetches the data packets and sends them to a segmentation module. The segmentation is based on a comparison of each pixel's brightness with a threshold while classifying the colors in two classes, forming a 2D monochrome matrix.

Further, the matrix is compared in more associative memory blocks. Specifically, we use 100 associative memory level 1 blocks to compare and identify patterns of images from 100 parts of the image (the image is like a puzzle of 100 slices). The outcome of each of the 100 memory blocks is fed to the input of an associative level 2 memory (results 4-bit per picture - so we have 16 classes of images for each slice). This level is responsible for identification of the object (or objects) from the image according to a template and sending the match as outputs. An important feature of our system consists of achieving milliseconds response time of the system from receiving image data output to providing output data, making the system suitable to analyze the motion pictures.
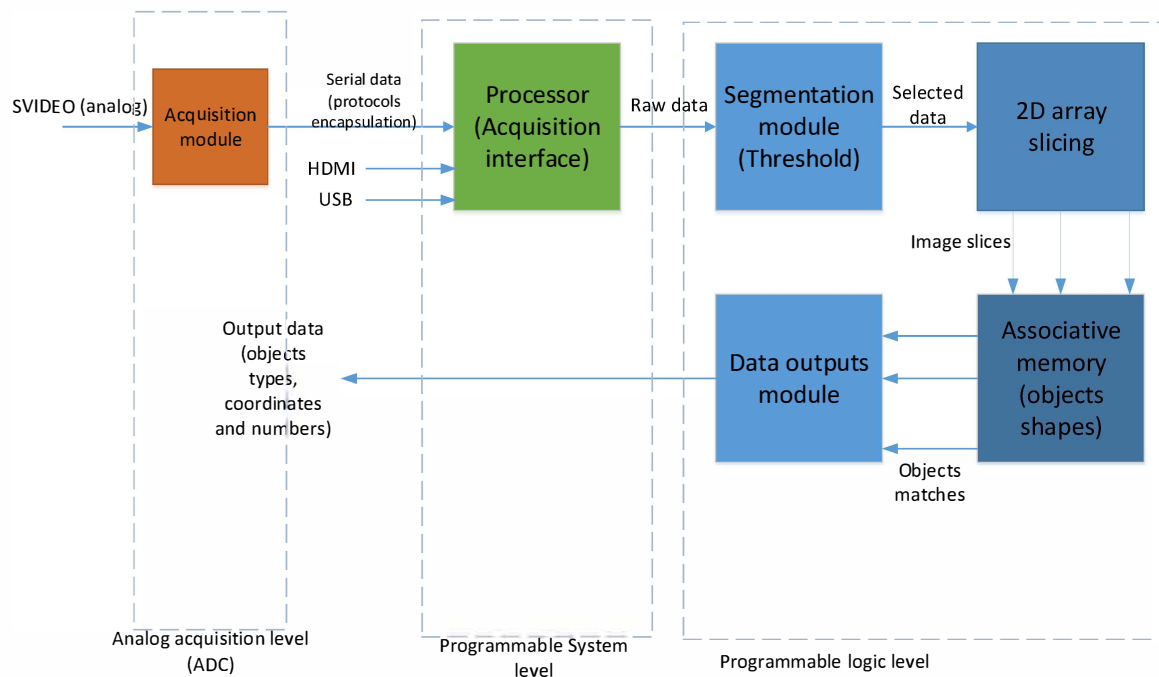

Fig. 1 Real –time image analysis system – block diagram

B. System components

As mentioned, the basic requirement is the real time response, as well as a sensor, so that the system be considered an intelligent sensor. For this we used a system on chip from Xilinx: Zynq 7000 (Z7010). The system contains everything needed to implement an intelligent image sensor. Thus, excluding the analog acquisition module needed if using analog cameras (which is based on the AD converter from Analog Devices - ADV7183B), everything represented in Figure 1 is implemented in the Zynq 7000: first of all an ARM Cortex A9 processor (866MHz) with two caches L1 and L2. It runs an operating system based on Ubuntu (Xilinux) containing integrated all stacks of protocols for HDMI and USB, necessary for reading images and converting them in

raw format digitally. On the other hand the image processing (shown in blue) should be run of parallel modules for high speed. Here we use an Artix 7 FPGA integrated into the same chip with the CPU. Thus, for an average FPGA price, all required processing stage for an intelligent image sensor is available.

III. Experimental results

Experiments were performed with image resolution of 1280 x 720 (720p). By segmenting an image is converted into a monochrome array, light color areas are extracted. The array with a bit depth of color is used for associative comparison with patterns. A comparison is get "match" for certain areas.

These are converted into coordinates of the object. They are transmitted along with the number of match-ins for an object.

Image divided in 36 slices (6 x 6)

Patterns for level 1 Associative Memory network

Input for level 2 associative memory: result from original image

Line 1: V0H0, V3H3, V3H3, V0H1,V3H3,V3H3
Line 2: V3H3,V0H0,V0H1,V0H1,V3H3,V3H3
Line 3: V3H3,V3H3,V3H0,V3H3,V3H3, V3H3
Line 4: V3H3,V0H1,V3H3,V0H0,V3H3, V3H3
Line 5: V0H1,V0H1,V3H3,V3H3,V0H0,V3H3
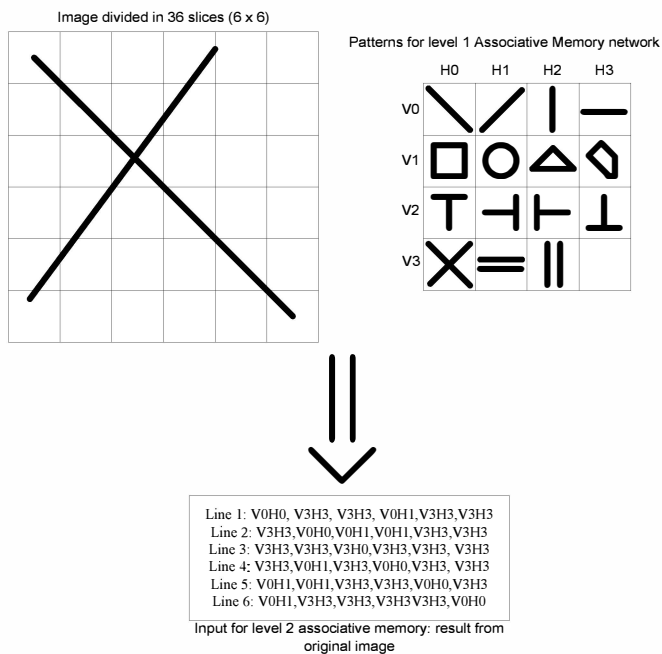Line 6: V0H1,V3H3,V3H3,V3H3,V3H3,V0H0

Fig. 2 Example of multi-level processing using associative memory: original image (up-left) divided in 36 slices, each slice is associated with level 1 patterns (up – right) and the results is inputs for 2 level memory

As can be seen from Figure 2, the system enables the identification of templates on two levels. First, based on an image captured by the camera, is a division of it into slices. Each slice represents a binary input in an associative memory - as stated, there are a total of 100 associative memory blocks that can take slices with resolutions of 20 x 10 pixels (pixel is a bit) so the entry is 200 bits. On the other hand, the memories have a 4-bit output which can therefore encode 16 classes of symbols.

For the experiments that occurred in this article we used 16 symbols shown in Figure 2, top right. Each memory takes the input and associates it with one of the 16 identified templates. To clarify, I represented the 16 templates in a table 4x4, each template having a coordinate both on vertically and horizontally represented Vx and Hx. For example, the first line of the picture has 6 slices so there will be 6 associative memory blocks that identify and will return the template code that identified (at the bottom of the figure represented as the pair VxHy).

In the level 2 associative memory the input will be fed from the values outputted by the level 1 associative memory

The solution was implemented on a system Zynq 7000 (manufactured by Xilinx). The circuit area allocated (meaning the logical resources including communication interface modules, is about 30%. As response times are shown in the table below.

Tab. 1 Time delay for system (estimated for programmable logic level)

Acquisition module (ADV7183B)	Acq. interf.	Seg.	Slicing	Comparison	Output packets module
~18 ns / pixel (54MHz/10bits)	125 ns / pixel (80MHz 3 ch., 10 bits)	~10 ns /pixel	<5 ns	< 20ns	~10 ns
Entire frame: 16.6ms	Entire frame: 115,2 ms	Entire frame: 921 us	100 slices depth max 500ns	100 comparisons max 2 us	22 bits size/ max 10 ns

The experiments were carried out with the figures contained triangles, circles, and rectangles arranged at random in the image. They were identified and counted up to 10 different items in each category. The detection rate of an object through associative algorithm is about 80%.
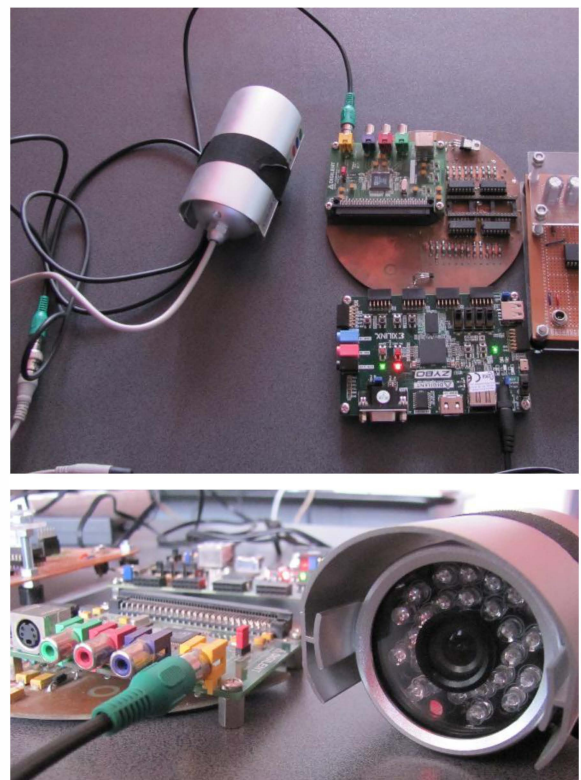
Fig. 3 Pictures with experimental system. Up: video camera to left, VDEC board with ADV7183B converter (top board) and interface for connection with Zybo board which is the main system with Zynq 7000 SoC. Bottom: front view of the system. Zybo board has 2 USB ports and 1 HDMI which can be used to connect camera with digital outputs.

There are works which deal with the takeover of image elements and using them as data for other applications. The novelty brought our work is related to the fact that we integrated the entire system: acquisition and shape detection on the same board. This considering that the circuit Zynq SoC are classified as average cost, suitable for general purpose applications. The system can be attached to already mounted video camera in a surveillance infrastructure.

IV. CONCLUSIONS

The system was used to identify images from a camera with a resolution of 1280 x 720 (720p HD). It was used to identify pictures of three classes: numbers, shapes and arrows. To identify them we used a network of associative memories where 16 figures were stored. The system had a response time of under 50 ms with correct response rate of 80%, performance comparable to other similar solutions.

Obviously the system can be configured to detect other type of objects in images. Future research directions could be applications where this sensor is used. One area that could be used is the intelligent surveillance systems. The possibility of attaching a surveillance camera and identification in real time, the place of the objects in the image can be used successfully in a metropolitan network surveillance where data volume sent to the dispatcher would be big otherwise.

REFERENCES

[1] Sanchez, Jordi; Benet, Gines; Simo, Jose E., " Video Sensor Architecture for Surveillance Applications", IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS Volume: 22 Issue: 3 Pages: 537-547 Published: MAR 2014

[2] Kristensen, Fredrik; Hedberg, Hugo; Jiang, Hongtu; et al., " An embedded real-time surveillance system: Implementation and evaluation ", JOURNAL OF SIGNAL PROCESSING SYSTEMS FOR SIGNAL IMAGE AND VIDEO TECHNOLOGY Volume: 52 Issue: 1 Pages: 75-94 Published: JUL 2008

[3] Hedberg, Hugo; Kristensen, Fredrik; Owall, Viktor, " Low-complexity binary morphology architectures with flat rectangular structuring elements ", IEEE transactions on circuits and systems i-regular papers Volume: 55 Issue: 8 Pages: 2216-2225 Published: SEP 2008

[4] D. Crookes; K. Benkrid ; A. Bouridane; K. Alotaibi; A. Benkrid, "Design and implementation of a high level programming environment for FPGA-based image processing", IEE Proceedings - Vision, Image and Signal Processing, Volume 147, Issue 4, August 2000, p. 377 – 384

[5] Uzun, I.S., "FPGA implementations of fast fourier transforms for real-time signal and image processing", Field-Programmable Technology (FPT), 2003. Proceedings. 2003 IEEE International Conference on, 2003

[6] P. Ridao, J. Amat, "A new FPGA/DSP-based parallel architecture for real-time image processing", Real-Time Imaging 8(5):345-356 · October 2002

[7] Yang, F; Paindavoine, M, "Implementation of an RBF neural network on embedded systems: Real-time face tracking and identity verification", IEEE TRANSACTIONS ON NEURAL NETWORKS Volume: 14 Issue: 5 Pages: 1162-1175 Published: SEP 2003

[8] Reza, AM, "Realization of the Contrast Limited Adaptive Histogram Equalization (CLAHE) for real-time image enhancement", JOURNAL OF VLSI SIGNAL PROCESSING SYSTEMS FOR SIGNAL IMAGE AND VIDEO TECHNOLOGY Volume: 38 Issue: 1 Pages: 35-44 Published: AUG 2004

[9] Diaz, J; Ros, E; Pelayo, F, "FPGA-based real-time optical-flow system", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY Volume: 16 Issue: 2 Pages: 274-279 Published: FEB 2006

[10] Hernandez, A; Urena, J; Garcia, JJ; et al., "Ultrasonic ranging sensor using simultaneous emissions from different transducers", IEEE TRANSACTIONS ON ULTRASONICS FERROELECTRICS AND FREQUENCY CONTROL Volume: 51 Issue: 12 Pages: 1660-1670 Published: DEC 2004

[11] Hu, CH; Xu, XC; Cannata, JM; et al., "Development of a real-time, high-frequency ultrasound digital beamformer for high-frequency linear array transducers", IEEE TRANSACTIONS ON ULTRASONICS FERROELECTRICS AND FREQUENCY CONTROL Volume: 53 Issue: 2 Pages: 317-323 Published: FEB 2006

[12] I. E.Ceuca, A. Tulbure, S.Plesa, " Integrated monitoring system for tracking green energy production" 2011 IEEE 17th International Symposium for Design and Technology in Electronic Packaging (SIITME) , Octomber 20-23,2011, Tmisoara, Romania