Hardware versus software for real-time video processing

The research efforts presented in this paper fit in a larger research project, Painvision [1,2], that aims at recognizing pain in real-time in demented elderly suffering from Alzheimer's disease by means of a camera system. Typically demented elderly in the last stage of their disease, are unable to communicate verbally [3]. A camera based system can monitor these patient's facial expressions 24 hours a day and 7 days a week. If a patient becomes uncomfortable, the camera based system can alarm a caregiver immediately.

One of the Painvision challenges is to recognize facial pain expressions out of the video stream in real-time. Focusing on patients, this is not only a Painvision challenge as can be seen in similar research done by Becouze et al.[4]. They developed an image processing algorithm to measure the degree of facial grimacing in patients using a single digital camera. This algorithm can process 2.5 frames per second (fps) using a MATLAB® implementation, which is not real-time as they want to process 30 fps. Our real-time constraint is set at a processing speed of less than 50ms for one image (which is 20 fps). This is considered to be a standard timing constraint.

By experimentation we identified the normalization step as the most computationally intensive part of the pain recognition algorithm. This step outputs frontal images of the patient's face on a predefined mask, independent of the pose and distance of the patient from the camera. This size and pose normalization makes the machine learning techniques more robust and perform better later on the expression recognition algorithm steps.

This normalization step is based on a dense affine transform (warp) which is pixel based, and therefore very computationally intensive. The current implementation in MATLAB® [5] cannot be performed in real-time.

In this paper we present four different implementations of the affine transformation, and compare these implementations based on their processing time. Moreover, the goal of this research was to investigate which implementation is suitable for a real-time application.

The first implementation discussed is the MATLAB® implementation. MATLAB® is a high-level programming environment and is commonly used to develop and test algorithms. However, no real-time processing times should be expected [4,6]. Significant faster processing times can be expected by the C++ implementations using Open source Computer Vision (OpenCV) [7-9], or the Open source Graphics Library (OpenGL) [10-12]. The last implementation is a hardware based alternative. A Field Programmable Gate Array (FPGA), is used to implement the algorithm onto hardware. We decided to program this device using a high-level modular design strategy, based on a Simulink® model [5]. In combination with Xilinx System Generator [13] a Simulink® model can be automatically translated into the appropriate Hardware Description Language (HDL) code required to configure the FPGA.

Measurements indicate that a hardware implementation is the fastest (1.1ms), followed by the OpenGL implementation (1.3ms). Moreover, the OpenCV implementation is considerably faster compared to MATLAB® (24.4ms and 153.7ms respectively). As MATLAB® is our reference implementation, we calculated the mean squared error of each output image compared with the MATLAB® output image. With a mean squared error between 3.7 and 4.2 we can conclude that all three alternative implementations produce a comparable output image.

References

[1] Painvision. In http://www.painvision.be, accessed on September 2009.

[2] Bonroy B, Schiepers, Leysens, Miljkovic D, Wils M, De Maesschalck L, Quanten S, Triau E, Exadaktylos V, Berckmans D, Vanrumste B. Acquiring a Dataset of Labeled Video Images Showing Discomfort in Demented Elderly. Telemedecine journal and e-health 2009; 15(4): 370-378.

[3] Reisberg B, Ferris SH, de Leon MJ, Crook T. The Global Deterioration Scale (GDS) for assesment of primary degenerative dementia. American Journal of Psychiatry 1982; 139: 1136-9.

[4] Becouze P, Hann CE, Chase JG, Shaw GM. Measuring facial grimacing for quantifying patient agitation in critical care. Computer Methods an Programs in Biomedicine 2007; 87: 138-147.

[5] The MathWorks^{TM}. In http://www.mathworks.com/, accessed on September 2009.

[6] Kiran M, Kan MW, Lim MK, Liang KM, Lai WK. Implementing image processing algorithms using ‘Hardware in the loop’ approach for Xilinx FPGA. International Conference on Electronic Design 2008; 1-6.

[7] Qingcang Y, Cheng HH, Cheng WW, Xiaodong Z. Interactive open architecture computer vision. 15th IEEE International Conference on Tools with Artificial Intelligence 2003; 406-410.

[8] Bradski G, Kaehler A. Learing OpenCV Computer Vision with the OpenCV Library, First edition. Sebastopol: O’Reilly, 2009.

[9] OpenCV. In http://sf.net/projects/opencvlibrary. Accessed on September 2009

[10] Paltashev T, Govind N, Abla G. Simulation of hardware support for OpenGL graphics architecture. Information Technology: Coding and Computing, 2000; 295-300.

[11] Soferman Z, Blythe D, John NW. Advanced graphics behind medical virtual reality: evolution of algorithms, hardware, and software interfaces. Proceedings of the IEEE 1998; 86(3): 531-554.

[12] OpenGL. In http://www.opengl.org/. Accessed on September 2009.

[13] Xilinx. In http://www.xilinx.com/tools/sysgen.htm. Accessed on September 2009.

Title	Hardware versus software for real-time video processing
Authors	*Bert Bonroy[1,2] MSc*, Pieter Schiepers[2] MSc, Filip Anthoni[1] MSc, Greet Leysens[1] RN MSc, Dragana Miljkovic[2] MSc, Maartje Wils[3] RN, Lieven De Maesschalck[1] RN MSc, Eric Triau[3] MD, Vasileios Exadaktylos[2] MSc PhD, Daniel Berckmans[2] MSc, PhD, Bart Vanrumste[1,4] MSc PhD
Affiliations	[1] MOBILAB, K.H. Kempen University College, Kleinhoefstraat 4, 2440 Geel, Belgium [2] M3-BIORES, Katholieke Universiteit Leuven, Kasteelpark Arenberg 30 - bus 2456, 3001 Heverlee, Belgium [3] De Wingerd, Living and Care Centre, Wingerdstraat 14, 3000 Leuven, Belgium [4]ESAT – SCD, Catholic University of Leuven, Kasteelpark Arenberg 10, 3001 Heverlee, Belgium
Corresponding author	*Bert Bonroy MSc* MOBILAB K.H. Kempen University College Kleinhoefstraat 4, 2440 Geel Belgium Phone: +32 14 56 23 10 Fax: +32 14 58 48 59
Email addresses	*bert.bonroy@khk.be*