SOFTWARE IMPLEMENTATION AND HARDWARE ACCELERATION OF RETINAL VESSEL SEGMENTATION

EMBC'17 JeJu, Korea

L. Cavinato, I. Fidone, M. Bacis, E. Del Sozzo, G. C. Durelli, M. D. Santambrogio
{lara.cavinato, irene.fidone, marco.bacis} @mail.polimi.it,
{emanuele.delsozzo, gianlucacarlo.durelli, marco.santambrogio}@polimi.it
Politecnico di Milano
Dipartimento di Elettronica, Informazione e Bioingegneria

Rationale & Contribution

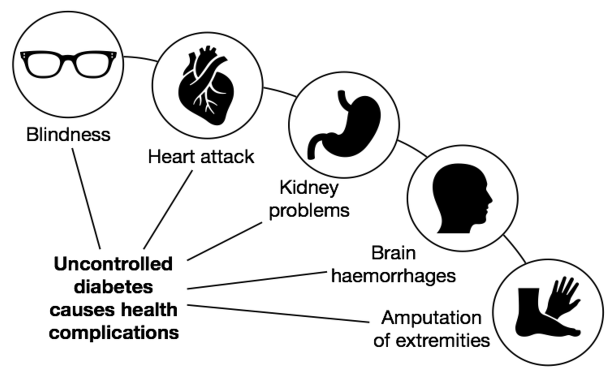
Screening tests are an effective tool for the diagnosis and prevention of several diseases. Unfortunately, in order to have an **early diagnosis**, the huge number of collected samples has to be processed **faster** than before. In particular this issue concerns image processing procedures, as they have a **high computational complexity**, which is not satisfied by modern software architectures.

Solution: Field Programmable Gate Arrays (**FPGAs**) can be used to accelerate partially or entirely the computation.

Case study: Retinal vessels Segmentation for **Diabetic Retinopathy screening tests**.

Context Definition

Diabetic Retinopathy (DR) is a complication of diabetes, it is caused by blood metabolic decompensation which brings to irregular micro-vascularization.

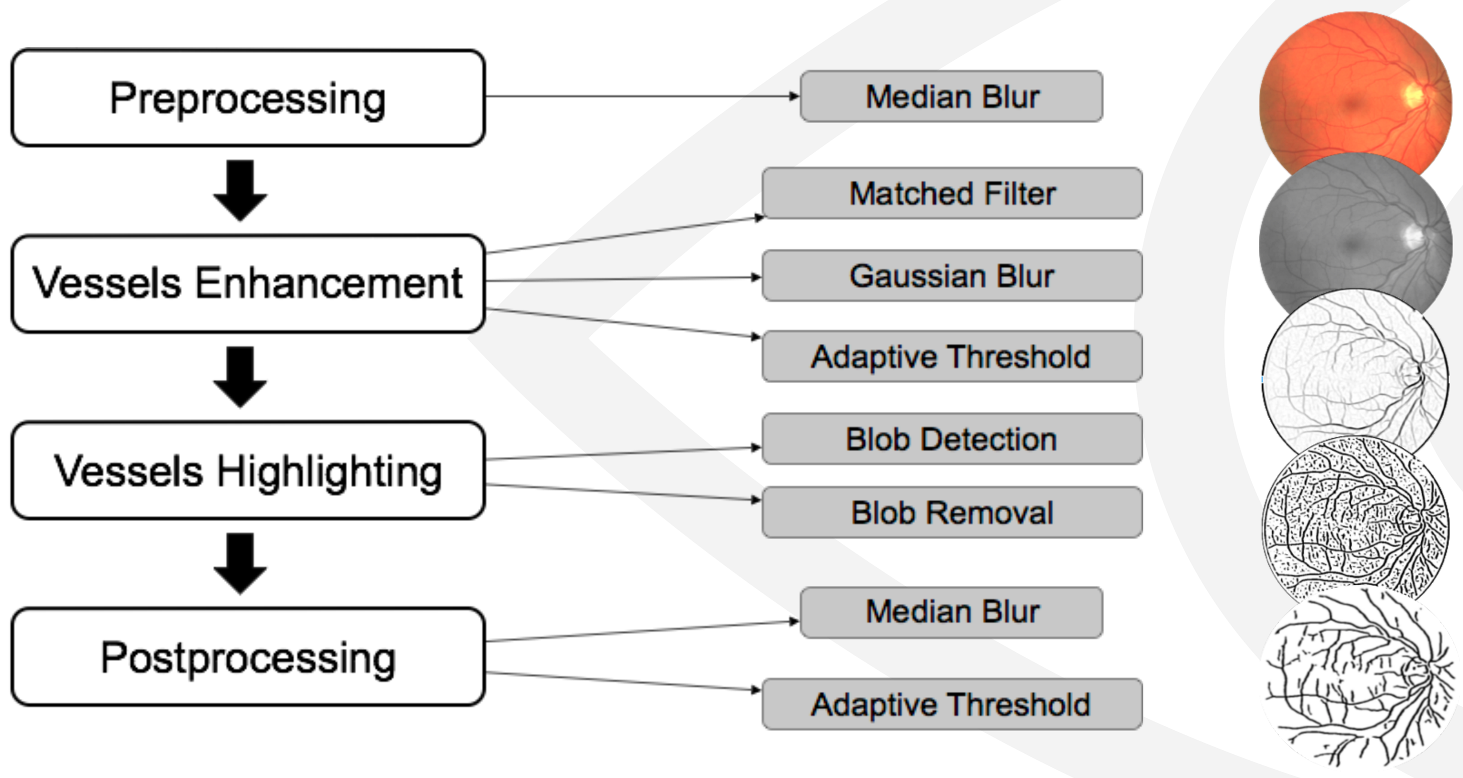


Segmentation is a technique that allows the partitioning of a digital image in two or more areas. In digital image processing, it enables the region of interest detection.

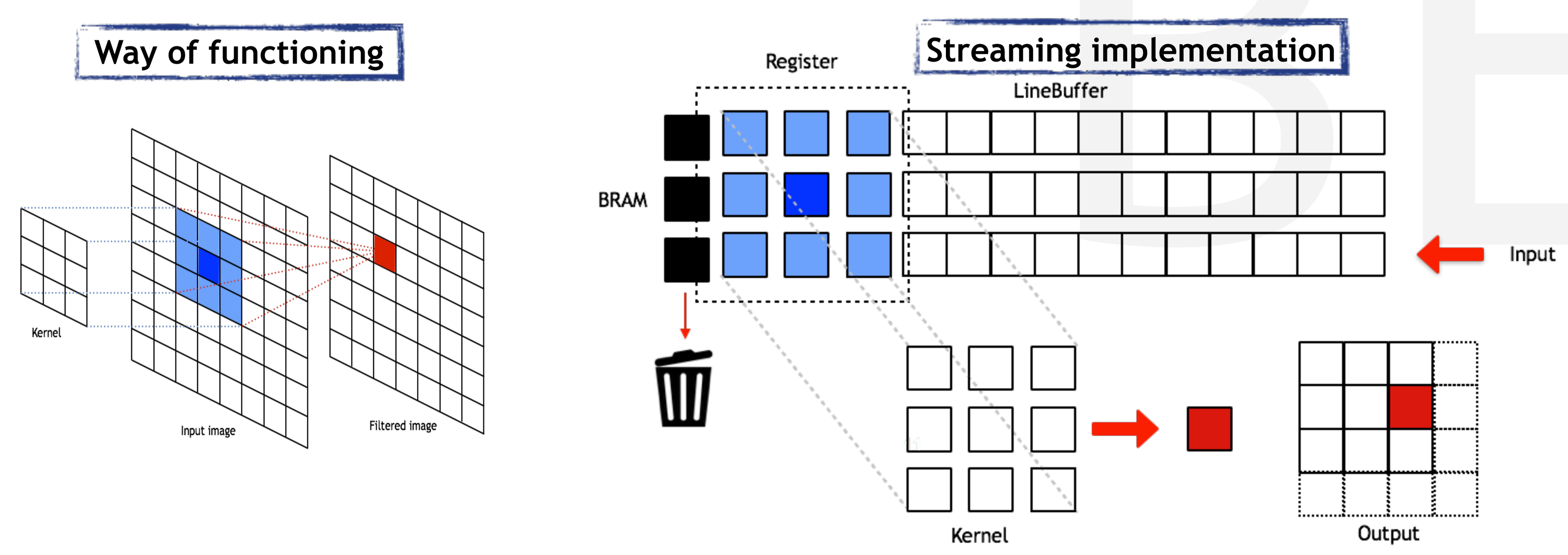
DRIVE [1] and **STARE** [2] Databases were used to evaluate our work.

Proposed Methodology

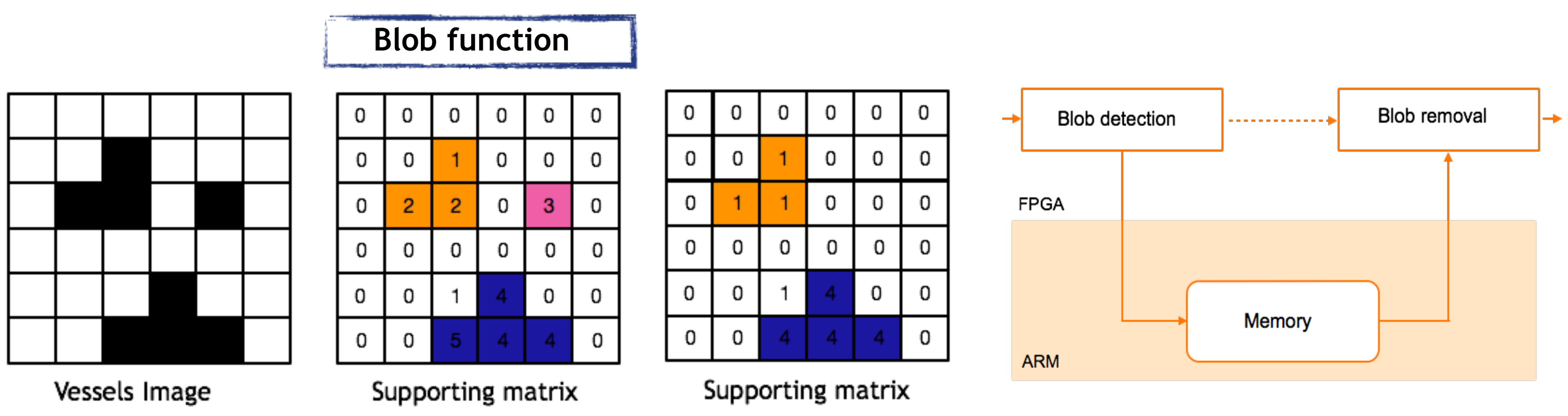
Algorithm: the proposed methodology consists in a pipeline of filters that process an input retinal image in order to detect the blood vessels. Each function corresponds to a hardware core.



Convolution is the main operator which join all the filters implementation, except for blob vessels structure highlighting step. It exploits:
- FPGA **BRAM** for storing image rows in a steaming architecture;
- FPGA **registers** to store kernels;
- **Pipelinig** and **unroll** pragmas to achieve maximum throughput.

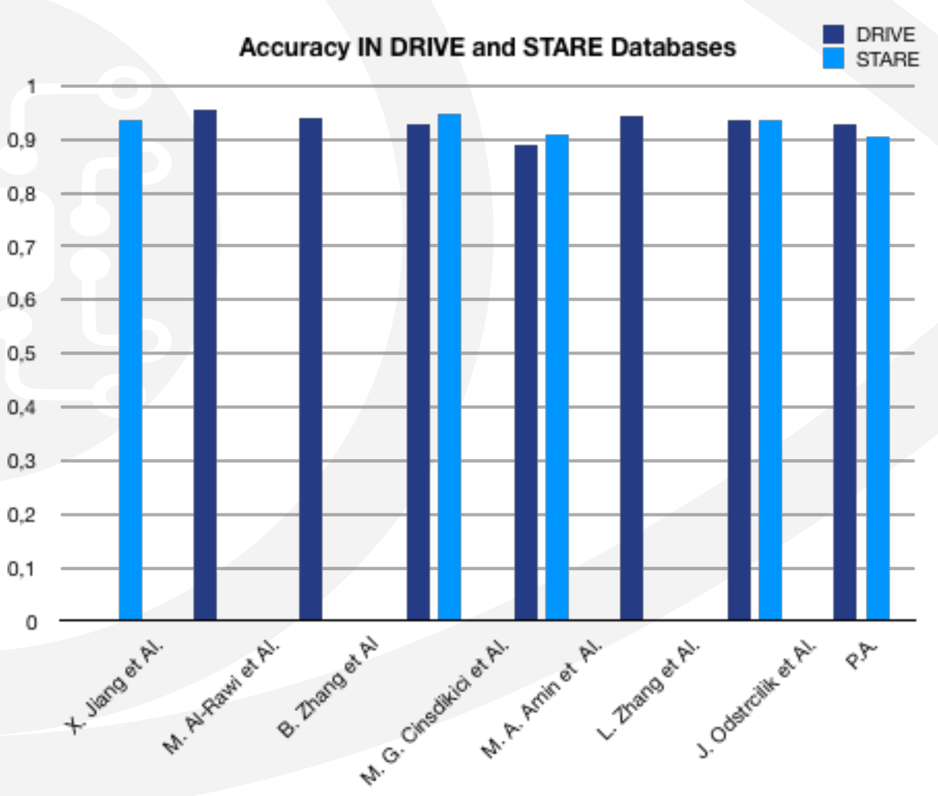


Blob detection and **blob removal** algorithms have been implemented differently. Since they are not linear filters, convolution can not been exploit. They basically consist on a **threshold** to single out vessels blob from noise. Here comes the need of use ARM memory to store the entire image.



Evaluation and Results

The Key Performance Indicators (**KPIs**): Accuracy and the latency of the implementation. The **accuracy** is the ratio between the correctly classified pixels count and the total number of pixels in the image. Then **latency** represents the average time to process a picture.



Work	Execution time DRIVE	STARE
X. Jiang et Al. [3]	-	19 s
M. Al-Rawi et Al. [4]	11 s	-
B. Zhang et Al. [5]	-	10 s
M. G. Cinsdikici et Al. [6]	35 s	-
M. A. Amin et Al. [7]	2 s	-
J. Odstrcilik et Al. [9]	3.22 s	4.07 s
P.A.	0.068 s	0.073 s

Implementation	Software	Hardware
Device	Intel Core i7	Zedboard
Execution time	0.06806 s	0.01041 s
Power consumption	26.929 Watt	4.749 Watt

The **software implementation** has been written in C++, using OpenCV [12] libraries. The test was run on a Intel Core i7-6700 CPU [13]. The architecture for the **hardware implementation** is an Avnet Zedboard [14] powered by a Xilinx Zynq-7000 All Programmable System on Chip.

Work	A. Nieto et Al. [10]	D. Koukounis et Al. [11]	P.A.
Device	Spartan 3	Spartan 6	Zedboard
Accuracy	0.9100	0.9007	0.9285
Execution time	1.4 s	0.03185 s	0.01041 s
Frequency	53 MHz	100 MHz	100 MHz

The possibility to have a very **low computation time**, allows to cut down on waiting times by providing a support for **real-time** diagnosis.

Future Work

Development of the implementation for the automatic detection of the diabetic retinopathy:
- Detection of the **micro-aneurisms** (non proliferative stage)
- Comparison of the vessels' structure (**neoangiogenesis**)
- Creation of an **embedded system** to perform screening tests

References

[1] M. Viergever, "DRIVE database." [Online]. Available: http://www.isi.uu.nl/Research/Databases/DRIVE/
[2] M. Goldbaum, "STARE database." [Online]. Available: http://www.ces.clemson.edu/ahoover/stare/
[3] X. Jiang and D. Mojon, "Adaptive local thresholding by verificationbased multithreshold probing with application to vessel detection in retinal images," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 1, pp. 131–137, 2003.
[4] M. Al-Rawi, M. Qutaishat, and M. Arrar, "An improved matched filter for blood vessel detection of digital retinal images," Computers inBiology and Medicine, vol. 37, no. 2, pp. 262–267, 2007.
[5] B. Zhang, L. Zhang, L. Zhang, and F. Karray, "Retinal vessel extraction by matched filter with first-order derivative of gaussian," Computers in biology and medicine, vol 40, no. 4, pp. 438–445, 2010.
[6] M. G. Cinsdikici and D. Aydın, "Detection of blood vessels in ophthalmoscope images using mf/ant (matched filter/ant colony) algorithm," Computer methods and programs in biomedicine, vol. 96, no.2,pp.85–95, 2009.
[7] M. A. Amin and H. Yan, "High speed detection of retinal blood vessels in fundus image using phase congruency," Soft Computing, vol. 15, no. 6, pp. 1217–1230, 2011.
[8] L. Zhang, M. Fisher, and W. Wang, "Retinal vessel segmentation using gabor filter and textons," Medical Image Understanding and Analysis (MIUA 2014), pp. 155–160, 2014.
[9] J. Odstrcilik, R. Kolar, A. Budai, J. Hornegger, J. Jan, J. Gazarek, T. Kubena, P. Cernosek, O. Svoboda, and E. Angelopoulou, "Retinal vessel segmentation by improved matched filtering: evaluation on a new high-resolution fundus image database," IET Image Processing, vol. 7, no. 4, pp. 373–383, 2013.
[10] A. Nieto, V. M. Brea, and D. L. Vilari˜no, "Fpga-accelerated retinal vessel-tree extraction," in Field Programmable Logic and Applications,2009.
[11] D. Koukounis, C. Tttofis, and T. Theocharides, "Hardware acceleration of retinal blood vasculature segmentation," in Proceedings of the 23rd ACM international conference on Great lakes symposium on VLSI. ACM, 2013, pp. 113–118.
[12] OpenCV library." [Online]. Available: http://opencv.org/
[13] "Intel Core i7-6700." [Online]. Available: http://ark.intel.com/it/products/88196/Intel-Core-i7-6700-Processor-8M-Cache-up-to-400-GHz
[14] "Zedboard." [Online]. Available: http://zedboard.org/product/zedboard