CNP: An FPGA-based Processor for Convolutional Networks

Clément Farabet[1] Cyril Poulet[1] Jefferson Y. Han[2]
Yann LeCun[1]
[1] Courant Institute of Mathematical Sciences, New York University
715 Broadway, New York, NY 10003
[2] Perceptive Pixel Inc.
cfarabet@nyu.edu
http://www.cs.nyu.edu/~yann

Convolutional Networks (ConvNets [5, 6]) are biologically-inspired hierarchical architectures that can be trained to perform a variety of detection, recognition and segmentation tasks. ConvNets have a feed-forward architecture consisting of multiple linear convolution filters interspersed with point-wise non-linear squashing functions. Because they can easily be trained for a wide variety of tasks (OCR [6], face/person detection [3, 8], object recognition [9], robot navigation [7, 4]), ConvNets have many potential applications in micro-robots and other embedded vision systems that require low cost and high-speed implementations.

Pre-trained ConvNets are algorithmically simple, with low requirements for memory bandwidth and arithmetic precision. Hence, several hardware implementations have been proposed in the past. The first one was the ANNA chip, a mixed high-end, analog-digital processor that could compute 64 simultaneous 8×8 convolutions at a peak rate of 4.10^9 multiply-accumulate operations per second [1]. Subsequently, Cloutier et al. proposed an FPGA (Field Programmable Gate Array) implementation of ConvNets [2], but fitting it into the limited-capacity FPGAs of the time required the use of extremely low-accuracy arithmetic.

We present a complete vision/recognition system on a single DSP-oriented FPGA. The design requires no external hardware, other than a few memory chips, and has been integrated onto a small 7×8 cm printed circuit board. It can be used to implement any vision based system in which the bulk of the computation is spent on convolutions with small-size kernels. The design is specifically geared towards ConvNets, but can be used for many similar architectures based on local filter banks and classifiers.

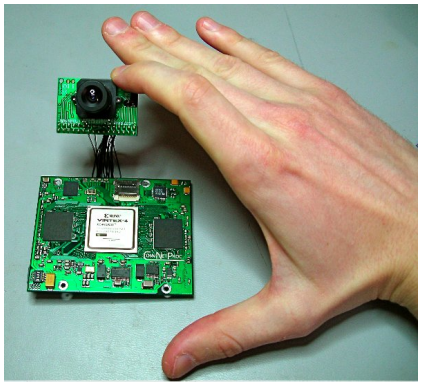

Figure 1: Our complete vision system.

In usual hardware designs, flexibility is left aside to maximize the efficiency of the system. Instead, the system described here is a programmable ConvNet Processor (CNP), which can be thought of

as a RISC (Reduced Instruction Set Computer) processor, with an instruction set that matches the elementary operations of a ConvNet. While these elementary operations are highly optimized, and make extensive use of the parallelism inherent in the hardware, implementing a particular ConvNet simply consists in reprogramming the software layer of the system, and does not require to reconfigure the logic circuits in the FPGA

A ConvNet face detection system was implemented and tested. Face detection on a 512×384 frame takes $100ms$ (10 frames per second), which corresponds to an average performance of 3.4×10^9 connections per second for this 340 million connection network (peak performance is 9.8×10^9 multiplications per second with a single hardware 7×7 convolution). The design can be used for low-power, lightweight embedded vision systems for micro-UAVs and other small robots.

Figure 2: Output of the system: (left) with, and (right) without non maximum suppression.

References

[1] B. Boser, E. Sackinger, J. Bromley, Y. LeCun, and L. Jackel. An analog neural network processor with programmable topology. *IEEE Journal of Solid-State Circuits*, 26(12):2017–2025, December 1991.

[2] Jocelyn Cloutier, Eric Cosatto, Steven Pigeon, Francis Boyer, and Patrice Y. Simard. Vip: An fpga-based processor for image processing and neural networks. In *Fifth International Conference on Microelectronics for Neural Networks and Fuzzy Systems (MicroNeuro'96)*, pages 330–336, Lausanne, Switzerland, 1996.

[3] C. Garcia and M. Delakis. Convolutional face finder: A neural architecture for fast and robust face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1408–1423, 2004.

[4] Raia Hadsell, Ayse Erkan, Pierre Sermanet, Jan Ben, Koray Kavukcuoglu, Urs Muller, and Yann LeCun. A multi-range vision strategy for autonomous offroad navigation. In *Proc. Robotics and Applications (RA'07)*, 2007.

[5] Y. LeCun. Generalization and network design strategies. In R. Pfeifer, Z. Schreter, F. Fogelman, and L. Steels, editors, *Connectionism in Perspective*, Zurich, Switzerland, 1989. Elsevier. an extended version was published as a technical report of the University of Toronto.

[6] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998.

[7] Y. LeCun, U. Muller, J. Ben, E. Cosatto, and B. Flepp. Off-road obstacle avoidance through end-to-end learning. In *Advances in Neural Information Processing Systems (NIPS 2005)*. MIT Press, 2005.

[8] M. Osadchy, Y. LeCun, and M. Miller. Synergistic face detection and pose estimation with energy-based models. *Journal of Machine Learning Research*, 8:1197–1215, May 2007.

[9] Marc'Aurelio Ranzato, Fu-Jie Huang, Y-Lan Boureau, and Yann LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *Proc. Computer Vision and Pattern Recognition Conference (CVPR'07)*. IEEE Press, 2007.

Topic: Visual processing and pattern recognition