Journal of Circuits, Systems, and Computers
Vol. 12, No. 6 (2003) 769–781

World Scientific
www.worldscientific.com

THE NEW FRAMEWORK OF APPLICATIONS: THE ALADDIN SYSTEM

ÁKOS ZARÁNDY, CSABA REKECZKY, PÉTER FÖLDESY, and ISTVÁN SZATMÁRI

Analogical and Neural Computing Systems Laboratory,
Computer and Automation Research Institute of the Hungarian Academy of Sciences,
Kende u. 13-17, Budapest, 1111 Hungary

The first CNN technology-based, high performance industrial visual computer called Aladdin is reported. The revolutionary device is the world premier of the ACE4k Cellular Visual Microprocessor (CVM) chip powering an industrial visual computer. One of the most important features of the Aladdin system is the image processing library. The library reduces algorithm development time, provides efficient codes, error free operation in binary, and accurate operation in grayscale nodes. Moreover the library provides an easy way to use the Aladdin system for those who are not familiar with the CNN technology.

Keywords: CNN technology; vision system; Cellular Visual Microprocessor.

1. Introduction

Cellular Neural Network (CNN)[1–3] research, among many other results, concluded the development of a number of analogic array processor chips.[4–9] The most advanced completed family member by these chips is the ACE4k,[8] which is constructed by an array of 64×64 analog processor cells. These cells can handle and store either gray level or binary images. The specialty of the analog processor array is that it is programmable. Its operation set is wide. It can handle grayscale or binary images, and it can apply dynamic analog array operations and local logic operations.[3,8] Its input–output speed reaches the minimal requirements of video processing. The accuracy of the chip is 7–8 bits, which is high compared to other analog processor chips[10] operating with this speed. These very good properties encouraged us to build the first high performance, CNN technology-based, industrial quality *visual computer*, called Aladdin.

The other fact which also motivated this development was the foreseen ACE16k chip, which has four times larger processor array, and roughly 10 times higher computational speed, thanks to a number of even more advanced features. The developed Visual Computer with the ACE4k chip will require only some minor software–hardware modification to be upgraded for the ACE16k chip.

The Aladdin system is a computational infrastructure around the ACE4k Cellular Visual Microprocessor. However, it is more than a straightforward

hardware–software environment. Beyond proving the possibility to reach all the functionality of the ACE4k chip on a basic level, with the Aladdin system we developed an image processing library. These high level functions can be called from C or the native language of the Aladdin system. The easy use of the image processing library makes the Aladdin system usable for those, who are not familiar with the CNN technology. This opens the gate wide for a large number of potential users, who have never dealt with these processors before.

2. System Description

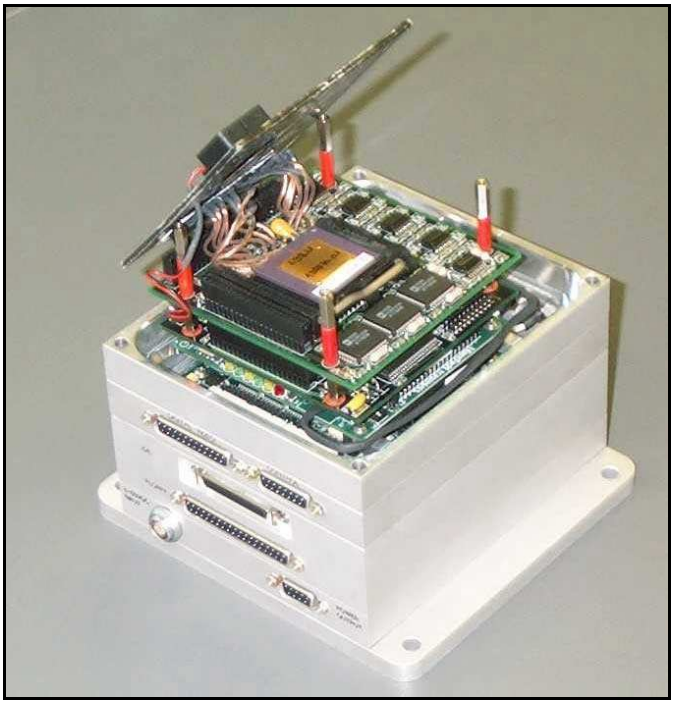
The Aladdin system is the high performance, professional computational environment of the ACE4k chip. This sophisticated software–hardware-optical environment is a high-speed successor of the earlier developed CNN Chip Prototyping System (CCPS),[11] which was already used for testing a number of analogic chip prototypes.[4–9] The main differences between the CCPS and the Aladdin are that while the former was designed to perform medium speed functional test in laboratory environment only, this latter can reach the peak performance of the ACE4k chip and can also be applied in industrial environment.

To match the industrial requirements the system was implemented on PC-104plus form factor cards (Fig. 1(a)). In this way, a standard industrial quality PC-104plus PC hosts a specially-designed DSP module, which drives the platform board carrying the ACE4k chip. To reach high-speed communication, PCI bus is used as a communication channel between the host PC and the DSP module. The DSP module, the platform and the hosting industrial PC constitute the rugged, high performance visual computer. However, if the system is used in less demanding environment, one can also install it in a desktop PC (Fig. 1(b)). This leads to a less expensive, but from the PC side, an even more powerful and versatile system.

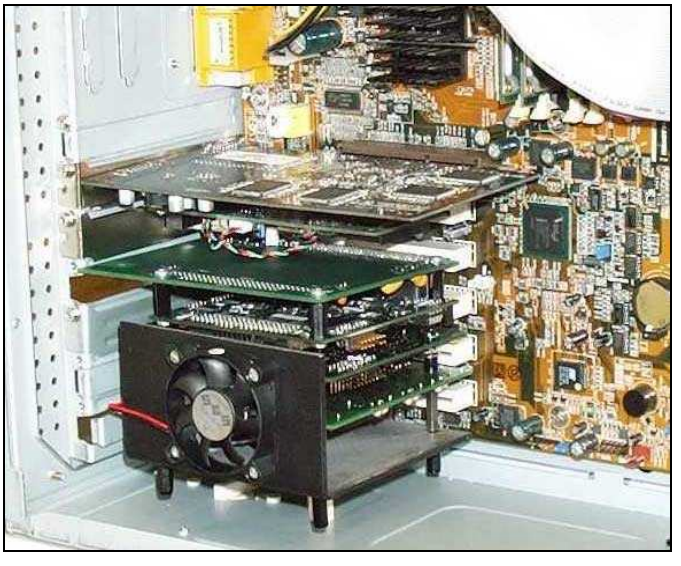
The physical arrangement of the industrial system can be seen in Fig. 2. This is composed of eight electrical cards, and a metal plane to carry the lens system. The size of a PC-104 form factor card is roughly the same as the size of a 3.5 inch floppy disk.

The lowest five cards of the rack (the power supply, the Pentium class motherboard, the hard-drive module, the network module, and the frame grabber module) are commercially available components. The power supply operates from any battery or DC power source (voltage range between 8 V and 25 V). The display card is integrated in the motherboard. In case of standalone operation the display and keyboard are not even connected to the Visual Computer. The network card provides the possibility of communication with remote computers. The frame grabber card is needed to support integration of different cameras to the system.

The block diagram of the upper three electronic cards (the DSP module, the digital and the analog platform) can be seen in Fig. 3. The main role of the DSP module is to execute the image processing program on the ACE4k chip. Since the ACE4k is a passive device, the DSP is used to provide all the fine control for it.



(a)



(b)

Fig. 1. The Aladdin system hosted by (a) an industrial and (b) a desktop PC.

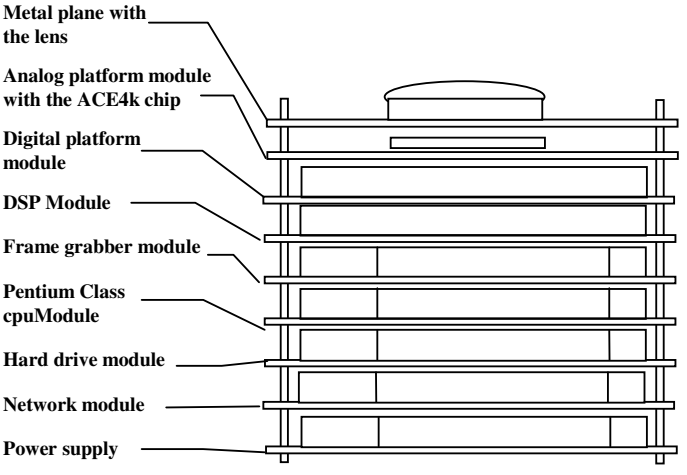


Fig. 2. The physical arrangement of the Aladdin industrial system.

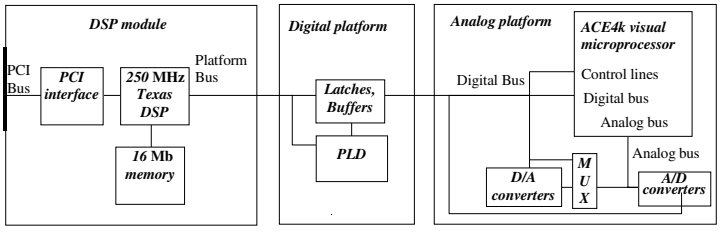


Fig. 3. The block diagram of the Aladdin system.

Moreover, the DSP feeds the ACE4k with data and reads out the results from it. Some nonlocal image processing tasks (like FFT) are also executed on this DSP board. We selected a high performance (250 MHz) Texas DSP to fulfill all these requirements. The other specialty of this DSP card is that it contains a specially designed high-speed asynchronous bus, called platform bus. This bus bridges the digital platform and the DSP module.

The main role of the two platform cards is to interface the analog and digital buses of the ACE4k chip to the DSP module. This mixed signal unit contains a large number of AD and DA converters, analog multiplexers, operational amplifiers, latches, and a PLD.

Above the platform, there is a metal layer, which has only mechanical and no electronic functionality. This layer carries the lens system, which focuses an image onto the silicon surface of the ACE4k to support its direct optical input.

The block diagram which shows the internal logic arrangement and the interconnection system of the Aladdin system can be seen in Fig. 4. The DSP module is connected to the motherboard of the host PC via the PCI bus. The PCI bus

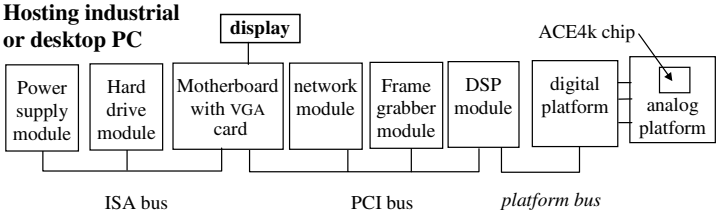


Fig. 4. The block diagram of the Aladdin visual computer.

Table 1. The measured operation speeds of the system.

Function	Speed
64×64 grayscale image transfer	2300 fps
64×64 binary image transfer	22 000 fps
Pixel-by-pixel logic operation on a 64×64 binary image	3.8 μ
Template operation on a 64×64 binary or grayscale image	3–15 μs
Processing of 64×64 sized images (grayscale input, binary output)	1500 fps
Processing of 64×64 sized images (binary input, binary output)	5000 fps

bridges the Aladdin system to the standard frame grabbers. The output images (if there is any) leave the system through the PCI bus and can be displayed on the display of the PC. In many cases, only decisions or a couple of identified events are read out from the system, or stored in the hard drive, and there is no need to send or store original or processed images.

From this architecture the motherboard, the power supply, the frame grabber, the hard drive module, and the network card are commercially available system components. The DSP module was designed and developed by RTD USA BME Laboratory. The platform was specified, designed, and developed by us.

The DSP module may be hosted by either a desktop or an industrial PC. The operating system on the PC can be either Windows NT or 2000. Since the DSP module uses a number of resources of the PC, a hosting program is needed (called CNNRUN) to provide the appropriate services. On the DSP module another program, called CNN Operating System (COS), runs. This program drives the platforms and the ACE4k chip.

The data transfer speeds and some general processing speed figures can be found in Table 1. More specific processing speed comparison tables can be found in the next section, in which the image processing library is introduced.

3. Image Processing Library

In this section, the image processing library functions are introduced. User can call them from the native AMC language[12] of the Aladdin Visual Computer and also from C language. The template values, transient time settings, reference values, etc. are all hidden from the users. Users have to parameterize the functions in the top

level only, like setting threshold levels, describing the structuring element set of a morphological operation, etc. In the list of the library functions the italic typeset indicates that a function is implemented on the ACE4K chip. Image processing functions with normal typeset are executed on the DSP of the system.

- **Basic data movement routines:**
  - Image transfer and type conversion from/to/inside the chip (e.g., *threshold*)
  - Full PC frame grabber card support
- **Pixel-wise image handling:**
  - *Logic operations*
  - *Addition, subtraction, multiplication, division*
  - *Binary and gray-scale translation*
- **General statistics:**
  - Histogram calculation
  - Mean, variance, median, *global maximum*, *minimum*
- **Image features, extraction:**
  - *Mathematical morphology* (*dilation*, *erosion*, *skeleton*, *prune*, *thicken*, *fill*, *shrink*, *pattern matching*, *single pixel removal*, *reconstruction*, ...)
  - *Classic functions* (*edge*, *shadow*, *hole filler*, *patch maker*, *concave place detection*, *perimeter structural operator*, *center-of-mass*, *majority*, ...)
  - *Connected component labeling*, object information: area, solidity, bounding box, orientation, convex area, ordered contour, ...
- **Image filtering:**
  - *Contrast stretching*, histogram equalization
  - *Linear and adaptive edge enhancement*, *Laplacian*, *Gaussian*, *Sobel*, *correlation*, *adaptive diffusion*, *gradient*, *thresholded gradient*
- **Transformations:**
  - Forward DCT, inverse DCT, wavelet processing, FFT-1, FFT-2, Haar, Hadamark, and Radon, JPEG compression, matrix quantization with rounding
- **Image synthesis:**
  - *Patterns* (*e.g.*, *checker*, *stripes*), noise, gradients
- **Signal processing:**
  - FIR, IIR filters, autocorrelation, weighted vector sum, matrix operations

3.1. *Binary image processing routines*

The library contains over 40 optimized binary image processing functions. All of these are error-free implementation of the binary operators. The operator implementations are based on dramatically reprogrammed switch configurations, signal

and weight ranges compared to the original recommendations of the user's guide of the ACE4K chip.[13] Moreover, the self and the spatial feedback loops are fully neglected resulting in higher robustness and significantly reduced settling time in binary output generation.[14] Therefore, the propagation type operations are implemented iteratively. With regards to the propagation type operations, an optional possibility is also incorporated to the library functions, namely the detection of the end of processing. By the usage of the global "AND" gate and logic comparison between consecutive results, at the price of a slight additional overhead, the automatic termination of the iterative operations is achieved.

3.1.1. *Pattern detection type morphological operators*

Pattern detection type binary morphological operations are implemented in a special way. It is based on the execution of a number of templates derived from the decomposition of the original nonrobust template. The subresults are combined with logic operators.

In case of the hit-and/or-miss type operations (e.g., skeleton, convex hull), one more step is needed, namely the accumulation of the pattern detection to the input. This is also done by the logic unit. The pattern can be detected within 6 microseconds.

3.1.2. *General threshold logic type operators*

The implementation of the general threshold logic class requires the detection of the black pixel ratio of the neighboring locations. We have implemented the cases that are possible to implement without errors, such as the four-neighborhood majority-vote-taker.

3.1.3. *Statistical filters and feature extraction*

Several image analyzing operations result in vectors or even scalars. Some of those functions can be executed at an efficient parallel way on the ACE4K chip. These are implemented with other functions shared between the DSP-ACE4K twins, which leads to a significant speed up. Typically, such operations are the convex area or the ordered contour functions.

3.1.4. *Examples*

In this section, we show some examples and analyze the execution times. Table 2 summarizes the execution times of the classic (nondecomposed) CNN solution, the library functions and the DSP solution as a comparison. As it can be seen, the library functions are not faster all the time, however they provide error-free results all the time. If we compare the execution times of the library function to those of a state-of-the-art DSP, it turns out that the ACE4K chip has almost an order of

Table 2. Runing time comparison for some typical operations between the nondecomposed CNN-type single template executions and the library solution in case of chip-sized images. As a comparison, we also show the execution times with a state-of-the-art DSP with its optimized assembly code. All images are 64×64 sized. The displayed times include on-chip data transfer and processing, which means that all the images are stored in the internal memories of the ACE4K and the DSP.

Operator	Nondecomposed CNN-solution (not error-free)	Library functions (error-free)	DSP solution (Texas C6202 @ 250 MHz) (error-free)
Binary edge detection of 4-nbr connectivity	7 μs	4.1 μs	31 μs
Binary edge detection of 8-nbr connectivity	7 μs	6.7 μs	58 μs
Reconstruction using 4-nbr connectivity	Initialization: 3 μs Propagation time per pixel: 1.1 μs	Initialization: 3 μs Propagation time per pixel: 0.65 μs	Propagation time per pixel: 31 μs
Reconstruction using 8-nbr connectivity	Initialization: 3 μs Propagation time per pixel: 1.2 μs	Initialization: 3 μs Propagation time per pixel: 1.5 μs	Propagation time per pixel: 58 μs
Skeletonization of 8-nbr connectivity	171 μs	61.7 μs	464 μs

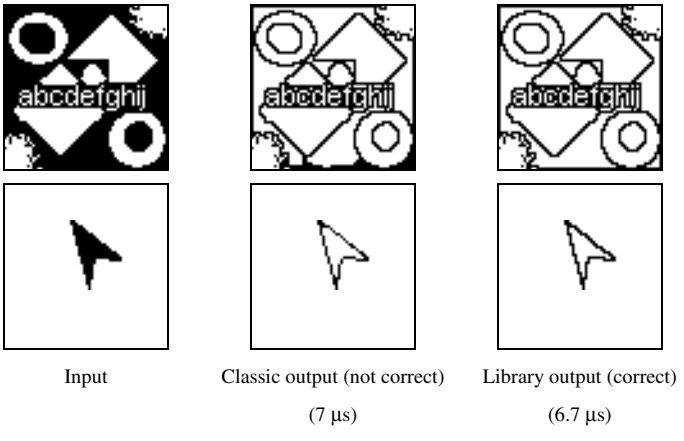


Fig. 5. The edge detection of eight-neighborhood connectivity. The classic CNN solution is not error-free.

magnitude speed advantage. We have to mention here that the DSP functions are optimized assembler functions provided by the manufacturer.

In Fig. 5 there is an eight-neighbor edge detection. A typical problem of the classic solution is that the template operation is image content sensitive. For example, the same template does not work correctly for two images having large difference in the number of the black pixels.

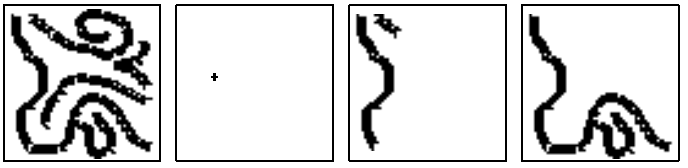


Fig. 6. The reconstruction using eight-neighborhood connectivity. The propagation front in the classic solution stopped earlier then it was supposed to, on the other hand, the unexpected propagation front was opened.

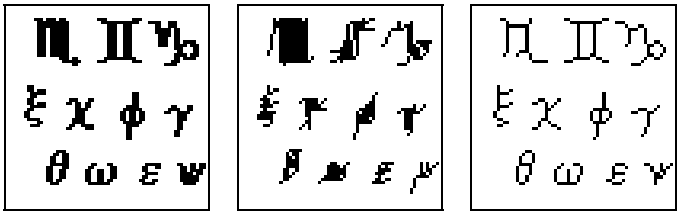


Fig. 7. Skeletonization.

Figure 6 shows a propagation type example. The propagation front in the classic solution stopped earlier then it was supposed to, on the other hand, the unexpected propagation front was opened. The iteration in the library implementation led to correct result.

Figure 7 shows the skeleton operator. Again, the classic method cannot precisely calculate the operator. The other interesting feature of the operation is that due to the uniqe calibration process, the library function is not just error-free, but also more than an order of magnitude faster.

Figure 8 shows the center point detection. It can be seen from its flow-chart (Fig. 9) that it is a very complex algorithm executed in a very short time.

3.2. *Library function examples: Grayscale*

The gray scale functions cannot be generalized in such an efficient way as the binary ones. Each operator is optimized individually including switch configuration sequences, timing, reference levels, and template values.[15] Table 3 shows the execution times of the optimized image processing library functions.

Here we present two groups of examples. The common property of the operators belonging to the first group (Fig. 10) is that the operators introduced on images are larger than the chip size. The images are cut into overlapping pieces, and the

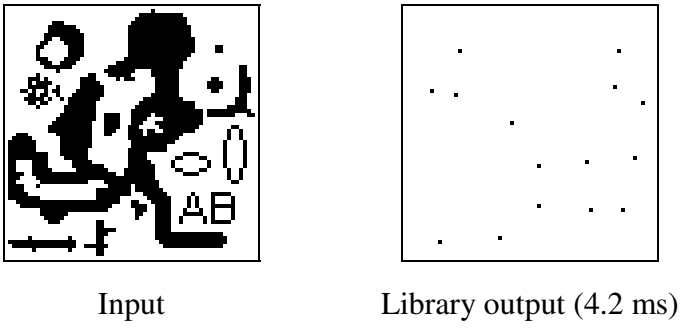


Fig. 8. Input and output examples for the complex center point estimation algorithm.

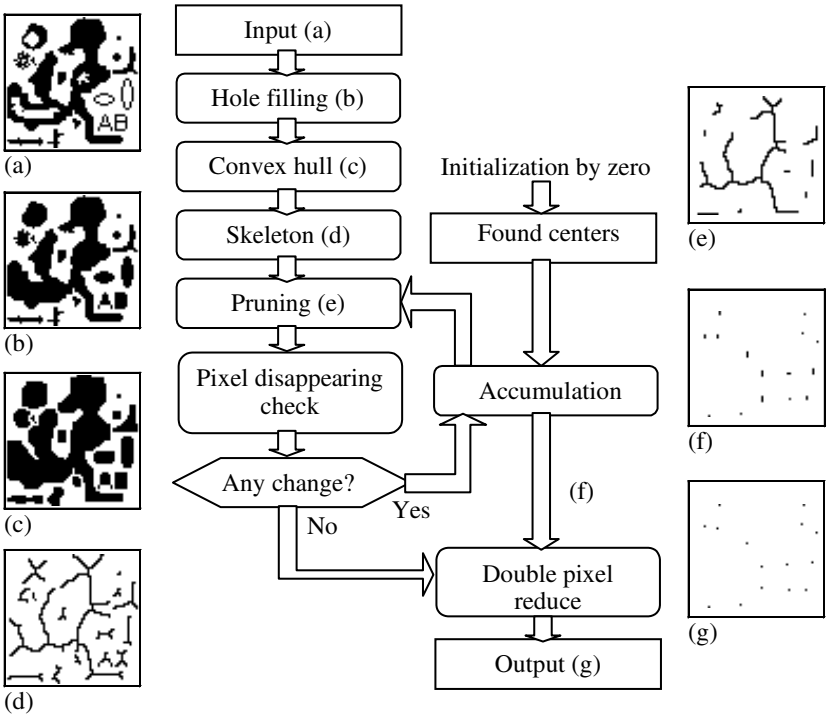


Fig. 9. The flow-chart of the algorithm of the complex center point estimation. For the sample image the whole process lasts for 4.2 ms.

segments are processed one after the other. Note that there are no checkerboard patterns in the large images, which indicates that the operation of the chip was robustly held in hand, and there were no offset differences between the runs on individual segments. Figure 11 introduces grayscale pattern detection, which is also a library function.

Table 3. The execution time for some typical grayscale function in case of chip-sized images.

Operation	Library solution	DSP solution (Texas C6202 @ 250 MHz)
Blurring	10 μs/1-nbr	110 μs (single 3×3 convolution)
Translation	Initialization: 3 μs Prop. time per pixel: 6 μs	n/a
Subtraction/addition	8.5 μs	12 μs
Contrast stretching	8.5 μs	21 μs
Window comparison	1.6 μs	25 μs
Grayscale pattern matching	20.7 μs/3×3 nbr 68.0 μs/5×5 nbr 181.3 μs/7×7 nbr	110 μs/3×3

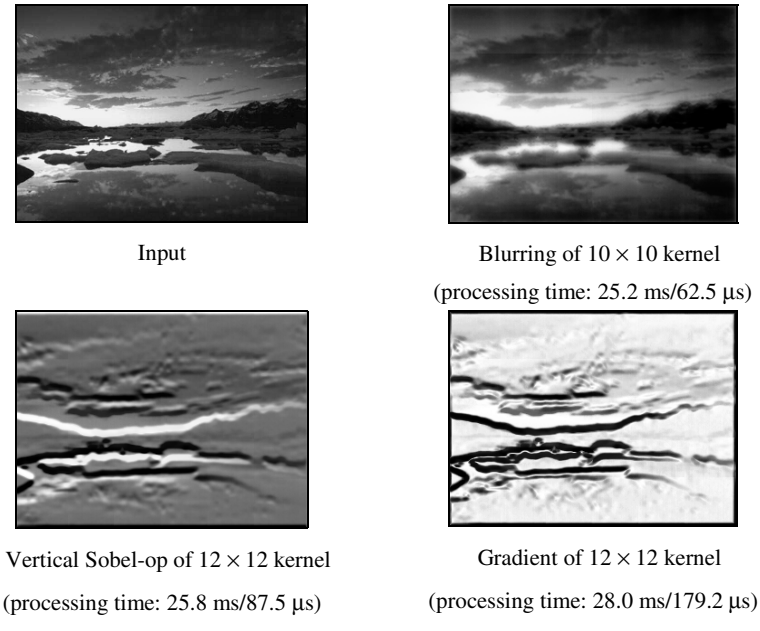


Fig. 10. Grayscale processing examples. The image resolution is 320×240. The images are cut into 24 overlapping 64×64 pixel-sized segments for processing. The first number indicates the total processing time including I/O between the chip and its environment, while the second number indicates the processing time including calibration and on-chip I/O only for a single 64×64 segment.

4. Application Areas

The Aladdin Visual Computer can be applied in three modes. The first is when a high-speed external camera is connected to the system via a PCI frame grabber. In this mode, ultra high frame rate (up to 5000) can be reached with low resolution

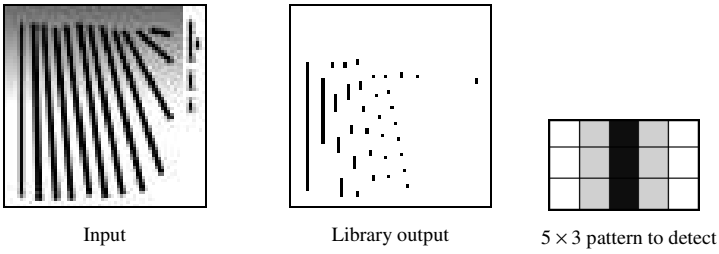


Fig. 11. Grayscale pattern detection. The processing has been executed in 40 μs. The lines are drawn using anti-alising filter and have 2 pixel width.

(64×64–256×256) images. The system in this mode can be applied as a visual trigger or a high-speed visual event detector. Moreover, the system can testify extremely high speed events, like flashes of a spark-plug, or can make shape analysis of a rapidly moving object, like a pill, or a grain.

The second mode uses the optical input of the ACE4k chip. Here the system can deal with 64×64-sized images only with a moderate frame rate. The good feature of this setup is the simplicity. It only requires power supply, the DSP module, and the platforms. This mode will be much more useful, when the ACE16k chip is integrated to the system, due to its enhanced optical input makes it possible to capture and process up to 10 000 128×128-sized images in a second. This leads to 10 000 visual decisions in a rapidly changing environment, which will be absolutely unique.

In the third application mode, the system processes video image flows coming from a single or a number of different cameras in real-time. Due to the high computational power of the system, it can process the whole frames, not just a small region of interest, like the digital image processing systems. This makes possible, e.g., the surface quality control even in those cases, when the surface is not periodic. Another typical application in this mode is the multi-modal image fusion, which can be used in a number of security or traffic safety applications.

5. Conclusion

A high performance visual computer based on CNN technology is introduced. Its computational power is roughly 10 times larger than up-to-date DSPs. An optimized image processing library was also developed, which provides efficient codes, error-free operation in binary, and accurate operation in grayscale nodes.

Acknowledgments

This work was supported by the DICTAM European 5th framework project (grant no.: IST-1999-19007) and the Hungarian OMKFHT grant (grant no.: IKTA-00019/2000).

References

1. L. O. Chua and L. Yang, "Cellular neural networks: Theory and application", *IEEE Trans. Circuits Syst.* **35** (1988) 1257–1290.
2. L. O. Chua and T. Roska, "The CNN paradigm", *IEEE Trans. Circuits Syst. I* **40**, 3 (1993) 147–156.
3. T. Roska and L. O. Chua, "The CNN universal machine: An analogic array computer", *IEEE Trans. Circuits Syst. II* **40** (1993) 163–173.
4. H. Harrer, J. A. Nossek, T. Roska, and L. O. Chua, "A current-mode DTCNN universal chip", *Proc. IEEE Int. Symp. Circuits Systems*, 1994, pp. 135–138.
5. J. M. Cruz, L. O. Chua, and T. Roska, "A fast, complex and efficient test implementation of the CNN universal machine", *Proc. Third IEEE Int. Workshop on Cellular Neural Networks and Their Application* (*CNNA-94*), Rome, December 1994, pp. 61–66.
6. R. Domínguez-Castro, S. Espejo, A. Rodríguez-Vázques, R. Carmona, P. Földesy, Á. Zarándy, P. Szolgay, T. Szirányi, and T. Roska, "A 0.8 μm CMOS 2-D programmable mixed-signal focal-plane array-processor with on-chip binary imaging and instructions storage", *IEEE J. Solid State Circuits* (1997).
7. A. Paasio, A. Kananen, and V. Porra, "A 176 $\times$ 144 processor binary I/O CNN-UM chip design", *European Conf. Circuit Theory and Design — ECCTD'99, Design Automation Day Proceedings* (*ECCTD'99-DAD*), Stresa, Italy, 1999.
8. S. Espejo, R. Domínguez-Castro, G. Liñán, and A. Rodríguez-Vázquez, "A 64 $\times$ 64 CNN universal chip with analog and digital I/O", *Proc. 5th Int. Conf. Electronics, Circuits and Systems* (*ICECS-98*), Lisbon, Portugal, 1998, pp. 203–206.
9. G. Liñán, R. Domínguez-Castro, S. Espejo, and A. Rodríguez-Vázquez, "ACE16k: A programmable focal plane vision processor with 128 $\times$ 128 resolution", *European Conf. Circuit Theory and Design* (*ECCTD '01*), Espoo, Finland, 28–31 August 2001, pp. 345–348.
10. P. Dudek and P. J. Hicks, "A CMOS general-purpose sampled-data analogue microprocessor", *ISCAS 2000*, Geneva, Switzerland, 28–31 May 2000.
11. Á. Zarándy, T. Roska, P. Szolgay, S. Zöld, P. Földesy, and I. Petrás, "CNN chip prototyping and development systems", *European Conf. Circuit Theory and Design — ECCTD'99, Design Automation Day Proceedings* (*ECCTD'99-DAD*), Stresa, Italy, 1999, pp. 69–81.
12. T. Roska, Á. Zarándy, S. Zöld, P. Földesy, and P. Szolgay, "The computational infrastructure of analogic CNN computing — Part I: The CNN-UM chip prototyping system", *IEEE Trans. Circuits Syst. I* **46**, 2 (1999) 261–268.
13. G. Liñán, R. Domínguez-Castro, S. Espejo, and A. Rodríguez-Vázquez, *CNNUC3B User Guide*, Instituto de Microelectrónica de Sevilla Centro Nacional de Microelectrónica, Universidad de Sevilla, 1999.
14. P. Földesy, G. Liñán, A. Rodríguez-Vázquez, S. Espejo, and R. Domínguez-Castro, "Sructure reconfigurability of the CNNUC3 for robust template operation", *Proc. IEEE Int. Workshop on Cellular Neural Networks and Their Applications* (*CNNA'2000*), Catania, 2000, pp. 289–294.
15. G. Liñán, P. Földesy, A. Rodríguez-Vázquez, and R. Domínguez-Castro, "Realization of nonlinear templates using the CNNUC3 prototype", *Proc. Sixth IEEE Int. Workshop on Cellular Neural Networks and Their Applications* (*CNNA'2000*), Catania, 2000, pp. 219–234.