Proceedings of the 1999 IEEE
International Conference on Robotics & Automation
Detroit, Michigan • May 1999

Frame-rate stereopsis using non-parametric transforms and programmable logic

Peter I. Corke [1] Paul A. Dunn [1] Jasmine E. Banks[23]

CSIRO Division of Manufacturing Technology [1]
CRC for Mining Technology and Equipment [2]
Space Centre for Satellite Navigation [3]
Pinjarra Hills, AUSTRALIA 4069.
http://www.cat.csiro.au/cmst/automation

Abstract

A frame-rate stereo vision system, based on non-parametric matching metrics, is described. Traditional metrics, such as normalized cross-correlation, are expensive in terms of logic. Non-parametric measures require only simple, parallelizable, functions such as comparators, counters and exclusive-or, and are thus very well suited to implementation in reprogrammable logic.

1 Introduction

Reliable and real-time 3D perception is critical to the operation of robots in complex real-world environments. This paper discusses recent developments in matching algorithms and in computing technology which together can meet this need.

In our work we have concentrated on area-based stereo matching since it provides dense range maps and works well in the natural outdoor scenes we are interested in. Stereo vision requires no special scene illumination or moving parts, and the sensors (CCD cameras) are small, low-cost, rugged and can acquire data very rapidly. The primary disadvantage has always been that considerable computation is required to establish *correspondence* or *matching* of points between the two images.

A number of software only solutions are now available commercially, for instance the Triclops[1] and SVM[2] modules. Using modern Pentium processors, and exploiting the MMX SIMD architecture extension very high matching rates can be achieved. However in order to achieve multiple frames per second even these systems have to work with very low resolution images. The capability and performance of some reported systems are compared in Table 2.

Section 2 discusses area-based matching metrics: classical as well as non-parametric transforms. Section 3 discusses a reprogrammable logic architecture, and Section 4 describes how that it is applied to frame-rate stereo matching.

2 Area-based matching

At the heart of all area-based matching systems is a matching metric, and a number of classical metrics are listed in Table 1. These compute similarity between square regions of pixels I_1 and I_2.

Non-parametric techniques[3] are based on the relative ordering of pixel intensities within a window, rather than the intensity values themselves. Consequently, these techniques are robust with respect to radiometric distortion, since differences in gain and bias between two images will not affect the ordering of pixels within a window. In addition they are tolerant to a small number of outliers within a window, and are therefore robust with respect to small amounts of random noise[4]. Two non-parametric transforms which are suited to fast implementation are[3]:

Rank Transform This is defined as the number of pixels in the window whose value is less than the centre pixel. The images will therefore be transformed into an array of integers, whose value ranges from 0 to $N-1$, where N is the number of pixels in the window.

Census Transform This transform maps the window surrounding the centre pixel to a bit string. If a particular pixel's value is less than the centre

0-7803-5180-0-5/99 $10.00 © Crown Copyright

Sum of Absolute Differences	SAD	$\sum_{(u,v)\in W} \lvert I_1(u,v) - I_2(x+u,y+v)\rvert$
Zero mean Sum of Absolute Differences	ZSAD	$\sum_{(u,v)\in W} \left\lvert (I_1(u,v) - \overline{I_1}) - (I_2(x+u,y+v) - \overline{I_2})\right\rvert$
Sum of Squared Differences	SSD	$\sum_{(u,v)\in W} (I_1(u,v) - I_2(x+u,y+v))^2$
Zero mean Sum of Squared Differences	ZSSD	$\sum_{(u,v)\in W} ((I_1(u,v) - \overline{I_1}) - (I_2(x+u,y+v) - \overline{I_2}))^2$
Normalised Cross Correlation	NCC	$\dfrac{\sum_{(u,v)\in W} I_1(u,v)\cdot I_2(x+u,y+v)}{\sqrt{\sum_{(u,v)\in W} I_1^2(u,v)\cdot \sum_{(u,v)\in W} I_2^2(x+u,y+v)}}$
Zero mean Normalised Cross Correlation	ZNCC	$\dfrac{\sum_{(u,v)\in W} (I_1(u,v) - \overline{I_1})\cdot (I_2(x+u,y+v) - \overline{I_2})}{\sqrt{\sum_{(u,v)\in W} (I_1(u,v) - \overline{I_1})^2\cdot \sum_{(u,v)\in W} (I_2(x+u,y+v) - \overline{I_2})^2}}$

Table 1: Classical area-based matching measures.

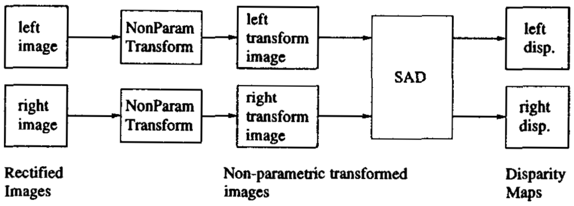


Figure 1: Overall matching process using non-parametric transform.

pixel then the corresponding position in the bit string will be set to 1, otherwise it is set to 0. The images will therefore be transformed into an array of N-bit integers ranging from 0 to $2^N - 1$.

The steps involved in matching using the rank and census transforms are shown in Figure 1. The stereo images are first transformed then matched using a traditional measure. We choose the SAD metric where the difference is arithmetic for the rank transform, and the Hamming distance

$$\sum_{(u,v)\in W} \text{Hamming}(I_1'(u,v), I_2'(x+u,y+v)) \qquad (1)$$

for the census transform.

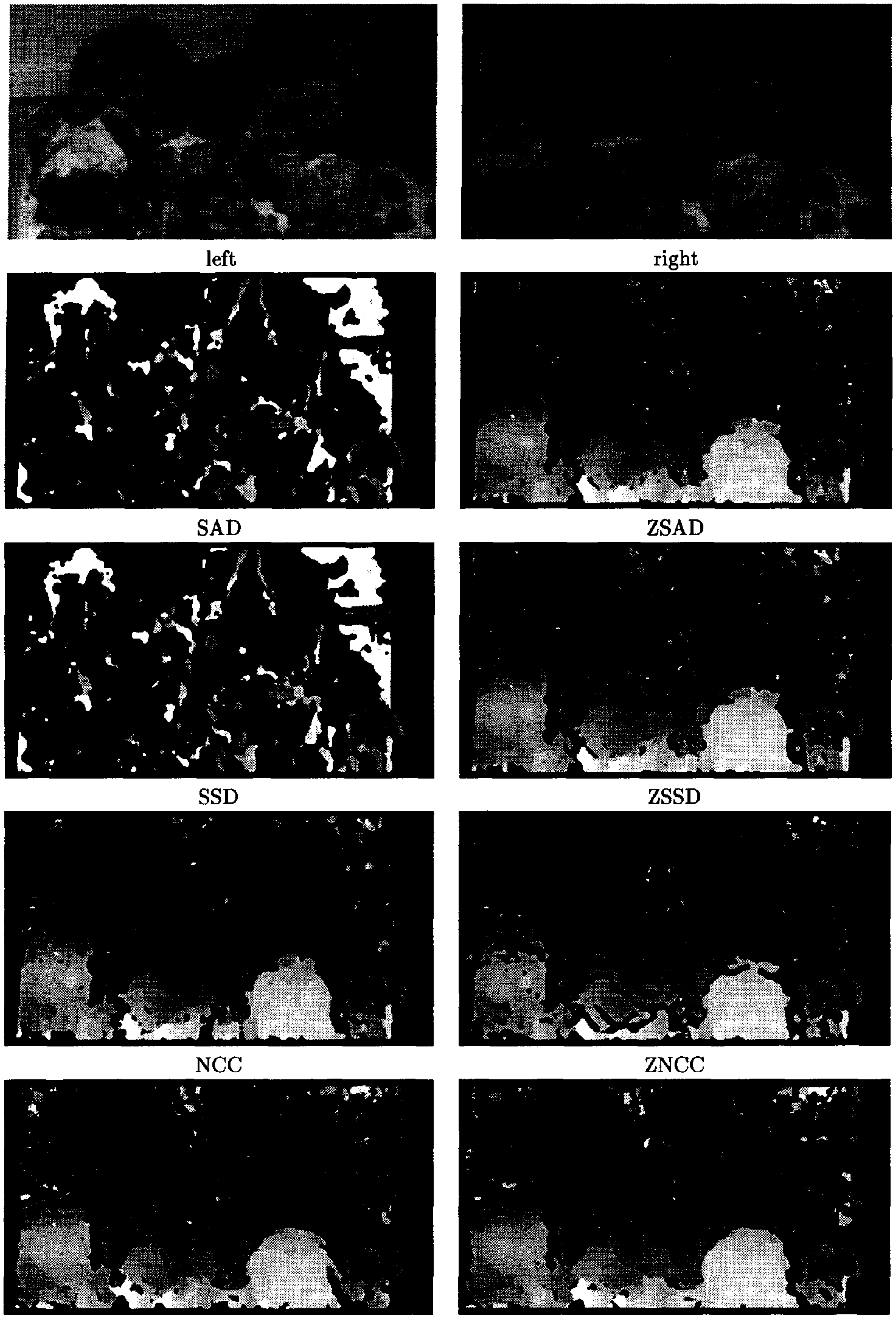
2.1 Results

The stereo pair, Figure 2[9], suffers significant radiometric distortion, that is, one image is brighter than the other. The disparity results obtained using the matching metrics above are shown in Figure 2. In each case, the disparity map is with respect to the right image and lighter regions correspond to larger disparities. A matching window size of 11 x 11 was used for each metric. The census and rank transforms were performed using windows of size 5 x 5.

It can be seen that the SAD and the SSD are not robust with respect to radiometric distortion. Use of the ZSAD, ZSSD, NCC and ZNCC resulted in improved performance but at the cost of increased computational complexity. The NCC and ZNCC are particularly expensive due to the presence of floating point multiplication, division and square root operations. More quantitative results are given in Banks etal[10].

3 The Configurable Logic Processor

A configurable logic device is a digital integrated circuit which contains combinatorial logic (logic gates) and state devices (flip-flops and RAM) which can be interconnected to form complex circuits. The interconnections between these components is by means of switches controlled by the contents of static memory



left

right

SAD

ZSAD

SSD

ZSSD

NCC

ZNCC

RANK

CENSUS

Figure 2: Disparity of IROCKS1 stereo pair, produced using various metrics.

Group	$n \times m$	d	T	FOM	L/R	comments
			(s)	(ns)		
CMU iWarp[5]	240×256	16	64 ms	65		64 processor iWarp computer
CMU STORM	256×256	32	2.82	1370		Sparc 10
INRIA software[6]	256×256	32	59	28000	yes	Sparc 2, forward-back match
INRIA DSP[6]	256×256	32	9.6	4600	yes	4 M96002 DSP
INRIA hardware[6]	256×256	32	0.28	134	yes	PeRLe-1 board
JPL[7]	64×60	32	0.6	1630	yes	Datacube + 68040
CMST CLP	256×256	32	34 ms	16	no	custom FPGA system†
PARTS engine[8]	320×240	24	23.8 ms	13	no	custom FPGA system
SRI SVM	160×120	32	45 Hz	36	yes	233 MHzPentium+MMX
Triclops	160×120	16	10 Hz	325	yes	Trinocular 266 MHzPentium+MMX

Table 2: Summary of reported stereo vision system performance. L/R indicates left right consistancy checking, figure of merit FOM = T/ndm and is an indication of the time to perform a single similarity measure (small is good). $n \times m$ is the image size and d is the number of disparties computed.

on the chip. By the simple means of loading the configuration memory with an appropriate data pattern, arbitrary logical circuits are created. These devices are known as *Field Programmable Gate Arrays* (FPGAs), the largest of which exceed 100,000 gates and 10,000 flip-flops.

An FPGA can be used as a computing device by constructing digital circuits which directly transform the data. The use of digital circuits specifically designed for the task at hand has of course been widespread, however until the advent of the FPGA this was an exercise whose cost had to be amortised over a large number of identical units to be worth considering. Application-specific logic has the principal advantages that

- the logic is specifically tailored to the task so that registers, adders, data paths etc. need be no larger that the data width required
- only functional units specifically required need be included, since the hardware is not general
- there is no sequential fetching and decoding of instructions, so hardware for this is not needed
- arbitrary degrees of parallelism are possible within the constraints of available resources

and the disadvantages are that the:

- interconnect tracks and switching, and configuration memory, occupy a substantial chip area and therefore the logic density is much lower than comparable logic devices
- switched logic paths result in slower propagation of signals

Processing clock rates up to 50MHz can be employed with current FPGAs, but this is rather slower than the clock rates of 300MHz and more that we now see in conventional CPUs. Effective use of FPGAs for computation is warranted where the algorithm meets the following conditions:

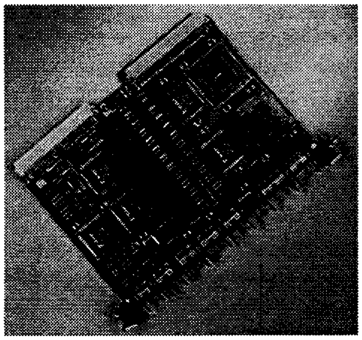


Figure 3: CLP board.

- the parallelism that can be employed is at least 10
- the arithmetic requirements are simple

The parallelism that can be achieved is often surprising as there are usually many housekeeping tasks that can be done in parallel with data operations. Parallelism can also be increased substantially by pipeline operations. In image processing the data is usually 8 bits and fixed point operations can be used, so arithmetic complexity is moderate.

The CLP[11], see Figure 3, was developed in 1993 and is a VMEbus circuit board containing several FPGAs[1] which can be configured via the VMEbus. The board includes digital input and output ports and associated timing compatible with the Datacube 10 MHz MAXBUS digital video format.

[1] Xilinx 4013 each of 13,000 'gates'.

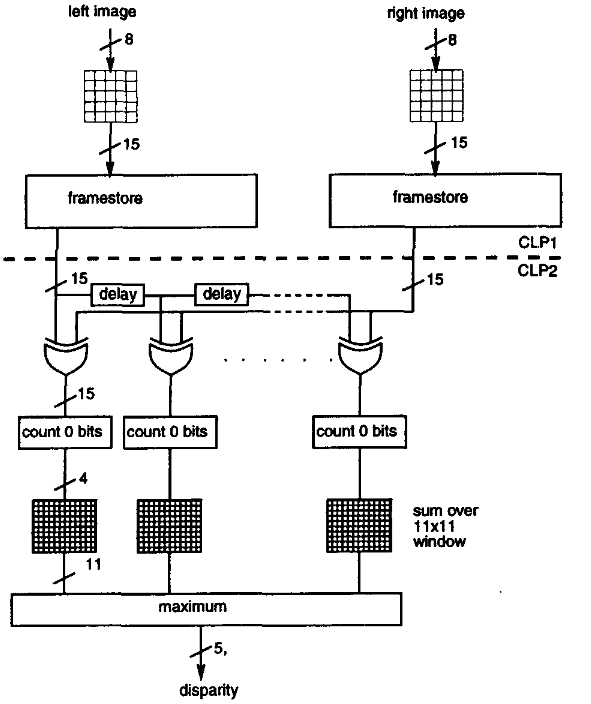


Figure 4: Data flow through the FPGA system.

The board contains four 512×512 8-bit frame stores which can be switched between various FPGAs. These use 20 ns static RAMs and are connected via a shuffle network to four of the FPGAs. All address, data and control signals are separate so as to allow independent access to each frame store. There are two 8,000 gate FPGAs for managing image input and output and four 13,000 gate FPGAs for data processing.

4 Configurable Logic Implementation

Our implementation has three major stages:

1. Compute census values over all 5 x 5 windows in each image.
2. Compare two census values using the exclusive OR operation and count the number of zero bits resulting. The sum of the number of matching bits over a larger, 11 x 11, window provides the similarity measure.
3. Find the maximum of the array of computed similarity measures, the index being the disparity.

Our stereo-matcher is implemented for an image size of 256 x 256 and two CLP boards are required.

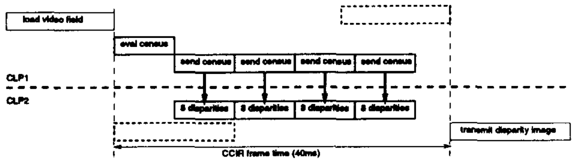


Figure 5: Timing diagram for the CLP implementation.

Figure 4 shows a general block diagram and the data flow. The first board writes the incoming left and right image pixel streams into frame stores, then reads the frame stores to form 15-bit census values for a 5 x 5 window around each pixel. These values are stored in two more frame stores. The census values are then read in raster order 11 lines apart for both left and right images and the four resulting data streams are passed to the second board for disparity computation.

The census algorithm requires 15 8-bit comparisons and this is very easily done in parallel with little logic. Census data generation from left and right images is done in parallel. The counting of match bits after the exclusive OR is also done in parallel. The row sums and column sums are done in parallel. Data streams are compared at 8 different disparities simultaneously. While census generation and image matching are done in separate passes, within these passes operations are pipelined so that data processing proceeds at clock rate. The many pointers used to access images and intermediate results are updated synchronously in parallel with data operations thus incurring no extra cycles. Thus while the main line processing is carried out at 10MHz, the parallelism achieved probably exceeds 100.

The second processing board compares the data streams from the left and right images at each of eight horizontal offsets (disparities) and stores the maximum match value and the offset at which it occurred. Four such passes are used to build up a total disparity range of 32 and the resulting disparity image is then written to an output port.

Image input and output can be overlapped with processing so the entire operation takes 5 passes where each pass occupies 6.8 ms. Total time is 34 ms which is less than the CCIR video frame time, see Figure 5.

4.1 Programming the FPGAs

Two compilers[2] have been developed for FPGA programming. A circuit generation language,

[2] http://www.mlb.dmt.csiro.au/IVT/clp/clp/clp_use /clp_use.html

CCGL[11], was developed first and is based on C language syntax. Its output is a logic netlist which is passed to the proprietry Xilinx tools for logic placement and routing. CCGL was developed before VHDL was available and continues to be used. It has proved to be an enormously flexible and concise way of expressing designs while still allowing optional control of placement and timing. Unfortunately such hardware description languages still require the user to be accomplished in circuit design.

4.2 Next generation CLP

The CLP board is no longer state-of-the-art and development of a more ambitious configurable logic processor is nearly complete. The new system, A Hybrid Modular Processor, provides three building blocks -

- an FPGA module for primary parallel logic
- a DSP module for intensive arithmetic or complicated sequential code
- an I/O module to interface to cameras and other external hardware devices

A selection of these modules can be interconnected into a processing network of a size and topology appropriate to the application.

Each module has two input and two output 20-bit data ports with a data rate of 60Mword/s. Data port interconnections are simple coaxial cables carrying serial data at 1.2GHz. As the processing network grows the data communications grows with it and hence there are no bottlenecks. Module control and communications is accomplished via an IEEE1394 (Firewire) backbone to a host processor (PC or workstation). Each module also carries a DEC/Intel StrongARM processor to manage backbone communications and provide any extra local processing.

The FPGA module contains 9 completely independent frame stores, 2 large lookup tables, a Xilinx 28,000 gate FPGA for input data buffering and control and uses a Xilinx 0.9 ns 150,000 gate array with 40 MHz, 60 MHz and 80 MHz clocks for processing. The DSP module contains a TI TMS320C6701. The I/O module couples the StrongARM processor with an FPGA for handling a diverse range of cameras, analog and signal conversion being managed via mezzanine board.

5 Conclusions

This paper has briefly discussed non-parametric transforms for stereo matching. These transforms exhibit matching quality at least as good as more traditional matching measures and have the significant advantage of being ammenable to hardware implementation. A frame-rate hardware implementation based on reconfigurable logic was briefly described. Increasing density and speed of FPGA technology would allow the complete system to now be implemented in a single chip.

References

[1] Point Grey Research, *Triclops stereo vision system*, 1998. http://www.ptgrey.com.

[2] K. Konolige, "Small vision module." http://www.ai.sri.com/ konolige/svm/index.html.

[3] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proc. 3rd European Conf. Computer Vision*, (Stockholm), May 1994.

[4] D. Bhat and S. Nayar, "Ordinal measures for visual correspondence," in *Proceedings Computer Vision and Pattern Recognition*, (San Fransisco), pp. 351–357, IEEE, 1996.

[5] J. A. Webb, "Implementation and performance of fast parallel multi-baseline stereo vision," in *Computer Architectures for Machine Perception*, (New Orleans), Dec. 1993.

[6] O. Faugeras, B. Hotz, H. Mathieu, *et al.*, "Real time correlation-based stereo: algorithm, implementations and applications," Tech. Rep. 2013, INRIA, Aug. 1993.

[7] L. Matthies, A. Kelly, T. Litwin, and G. Tharp, "Obtacle detection for unmanned ground vehicles: a progress report," in *Robotics Research: the Seventh International Symposium* (G. Giralt and G. Hirzinger, eds.), Springer, 1996.

[8] J. Woodfill and B. V. Herzen, "Real-time stereo vision on the parts reconfigurable computer," in *IEEE Workshop on FPGAs for Custom Computing Machines*, pp. 242–250, Apr. 1997.

[9] R. Bolles, H. Baker, and M. Hannah, "The JISCT stereo evaluation," in *Image Understanding Workshop*, pp. 263–274, DARPA, 1993.

[10] J. Banks, M. Bennamoun, and P. Corke, "Fast and robust stereo matching algorithms for mining automatio n," in *Proceedings of the International Workshop on Image Analysis and Information Fusion*, (Adelaide, Australia), pp. 139–149, November 1997.

[11] P. Dunn, "A configurable logic processor for machine vision," in *Proc. 5th International Workshop on Field-Programmable Logic and Ap plications* (W. Moore and W. Luk, eds.), pp. 68–77, Springer, 1995.