MVA '98 IAPR Workshop on Machine Vision Applications, Nov. 17-19, 1998, Makuhari, Chiba, Japan

11—4

Design and realisation of a Parallel Systolic Architecture dedicated to Aerial Image Matching.

Edwige E. Pissaloux
François Le Coat
Université de Paris, IEF
91 405 Orsay Cedex, France
{ep, lecoat}@ief.u-psud.fr

Patrick Bonnin
Université Paris 13,
93 430 Villetaneuse, France,
bonnin@iutv.univ-paris13.fr

André Tissot, François Durbin,
Thierry Garié
CEA (Comissariat à l'Energie Atomique)
Direction d'Applications Militaires
(DAM)
DRIF/DCRE/SEIM
91 680 Bruyères-le-Châtel, France

Abstract.

This paper addresses a hardware design of low cost, real time (faster than video rate) and suitable for on board applications systolic VLSI circuit, named μPD, for aerial image matching. Matching operation temporal results from Pentium PC200 (software simulation of the μPD) and estimated with Xilinx XC 6264 (encompassing μPD) working at 50 MHz are provided : the speed up factor is 2000 with frequency equivalent systems.

1. Introduction.

Image matching is a very important operation in vision : almost all autonomous guided systems (for nuclear, industrial and medical robotics, environment protection, trajectory matching, ...) use it.

The Bellman's dynamic programming algorithm ([Bel57]) can be used not only to calculate the distance between 1D signals (voice, [Qué92]) but also images. Indeed, it is possible to consider image as a 2D signal which temporal values can be obtained using two separable variables, one per dimension. Therefore, the search of the matching path between two lines/columns (one from two different images) can be done using the dynamic programming algorithm.

However, the distance between images has to encompass the potential changes in images due not only to a possible geometric transform between images, but also to the image content and lighting modifications. Consequently, a convenient image distance function in any pixel of both images has to be defined ([Pis96]).The calculation of 64K distances (for 256 × 256 images) is a very time consuming operation, even evaluated on a parallel computer ([LeC97]). Consequently, in real-time system, the matching has to be performed by a dedicated hardware.

Unfortunately, there is not commercially available VLSI circuit which supports dynamic programming algorithm. This paper presents a systolic architecture dedicated to match lines/columns, designed using the Xilinx FPGAs (XC 6264), named μPD, and briefly shows its application to image matching.

The following sections address : the dynamic programming parallel algorithm (Section 2) ; μPD circuit architecture and its cost optimisation (Section 3) ; μPD circuit usage for image matching (Section 4) ; some concluding remarks (Section 5).

2. Dynamic programming parallel algorithm.

The proposed algorithm is Bellman's dynamic programming principle adaptation for image processing purposes. It uses two concepts :

- the **local cost function C(s)** which can be
  - — a simple L1 distance between image pixels,
  - — any more elaborate form of distance, in function of information on images (lighting condition, content, plane transform linking two images,...) (cf. [Pis96], [LeC97]) ;

  this distance is used in order to define the cost of the local path s development between two consecutive compared pixels (in orthogonal or diagonal directions). The diagonal cost is less than the orthogonal (because the diagonal path is developed when the pixels are equal) ;

- the **global cost function** (score) :

$$score = Min \sum_{paths} d_{ij}.C(s)$$

this cost corresponds to the shortest path developed between 2 lines/columns (shortest in terms of time).

Figure 1 summarises different processing steps which aims to calculate the *score*. All paths in three directions are developed asynchronously in parallel starting with pixel (0,0). A path is constructed iteratively, from one to another couple of compared pixels (of scanned lines/columns) (cf. Fig. 2).

The corresponding parallel algorithm has been implemented in C* on the CM—5 of the ETCA, a Defence Research Laboratory, France. A 2D array of PEs is associated with, one processing elements (PE) per pixel. Each PE is 3 — connected to its 3 neighbours in a grid : south-east, east, south, what corresponds to the diagonal and two orthogonal path directions development. Each PE up-dates conveniently the global variable *score* with its local path development cost. Once a PE has contributed to a path development and score calculation, it disables its activity.

```
ALGORITHM
        INIT = true ;
        /* PE(0, 0) = active ; score = 0 */
        FIN = false;
        /* a global Boolean variable which
allows to stop the distance calculations */
        WHILE (NOT FIN) DO IN
PARALLEL ON ALL ACTIVE PEs
                • RECEIVE, from the valid
neighbour, the score ;
                • UPDATE the score with
your cost local value ;
                • ACTIVATE (at convenient
instant) your direct neighbours in three
directions (south, south-east, east) ;
                • SEND the updated score to
your neighbours (in three directions);
                • FIN := IF [(i = N AND (j =
N)] THEN true ;
                • DISABLE your work.
```

Figure 1. Parallel algorithm for minimal cost path search for two lines/columns.

A temporal complexity of the parallel algorithm is O(N) (i.e. one order less than this of the sequential algorithm).

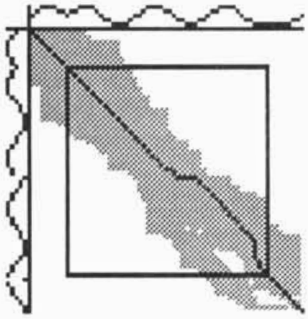

Figure 2. Parallel paths development when comparing 2 lines/columns (shadowed area ; bold path gives the minimal cost path for compared signals).

The matching of the whole images consists of the matching of N lines and N columns, thus it can be performed in $O(N^2)$ steps.

Section 3. Architecture of a parallel systolic circuit.

Figure 3 overviews the global architecture of a parallel systolic circuit supporting algorithm of Figure 1 (for 1D matching). The corresponding circuit, according to the theoretical considerations, should have N^2 processing elements (PEs) (N = 48 for the first chip version). However, taking into account the final application constrains it is possible to reduce this number. Indeed, in function of (affine or projective) transform linking two images parameter variations, only few PEs, spatially close to matrix diagonal, have to be physically implemented (cf. Figure 4). In the case of our application to aerial image matching, up to 20% disparity between pixels luminosity distance is admitted. Therefore, the « wide » of the suitable diagonal VLSI circuit, named µPD, will be 8 (8/32 = 25%) on diagonal's both sides. Consequently, the final VLSI circuit will have only 17N PEs (instead of N^2 of the full array).

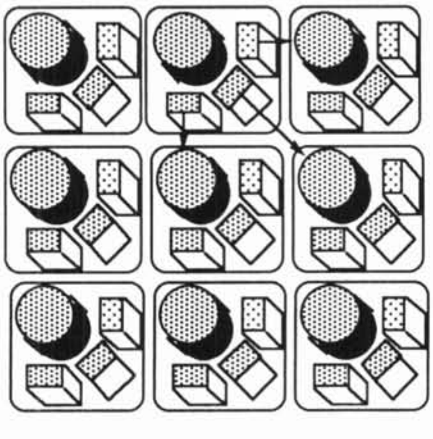

Figure 3. Global architecture of the µPD circuit.

The internal architecture of a PE, given on the Figure 4, is very simple. The CE signal activates (enables) circuit work (described by the algorithm Fig. 1). Two internal registers are used for south, east and south-east neighbours activation at convenient instant (in function of pixel distance d_{ij}). The memory 2-bit cell saves the PE activation direction ; its content will be used during the optimal path backtracking step.

A PE encompasses the operative part only ; the circuit control will be assumed (for this prototype) with a micro controller. The 256 × 256 µPD circuit will be implemented with 6 XC 6264 Xilinx FPGA, 70% of each used. About 60 input/output paths per circuits seem necessary.

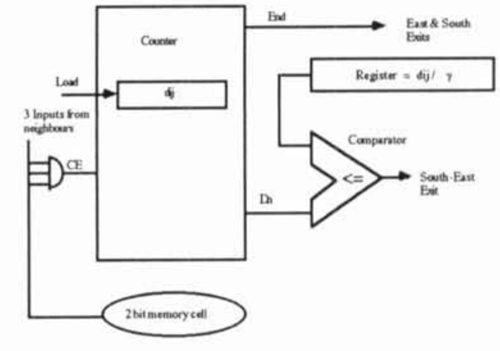

Figure 4. Detailed architecture of a PE.

4. Application for 2D image matching.

Figure 5 shows that the 1D dynamic programming primitive can be used for image matching by its successive orthogonal composition : one for two lines and one for convenient two columns (orthogonal dynamic programming). The whole image matching processing applies iteratively orthogonal dynamic programming through multi-resolution (processing pyramid) from 32 × 32 up to the final resolution (256 × 256 in our case) ; the iteration tunes projective transform parameters.

The temporal results on a system using µPD circuit proves its efficiency ([LeC97], [Pis96]). Indeed, the average time for 2 256 × 256 image matching is about 2000 shorter using µPD when running it on the Pentium PC 200 MHz (8 minutes/2,5 seconds = 500 or, with the frequency equivalent systems, 2000).

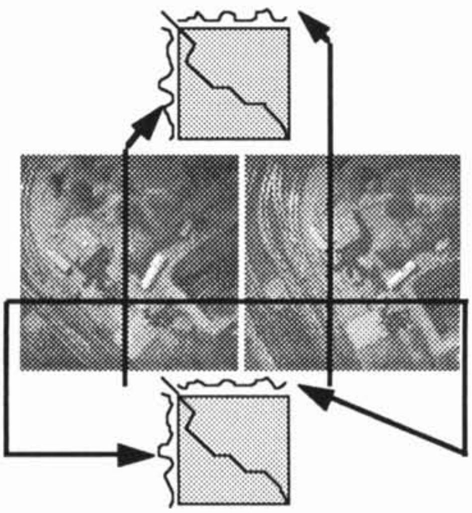

Figure 5. Principle of 2D signals matching using 1D modified dynamic programming.

5. Concluding remarks.

This paper has proposed a new fast and low cost systolic architecture for hardware implementation of a modified dynamic programming algorithm, named µPD circuit. The proposed parallel architecture (algorithm) reduces by one order, the sequential complexity of the basic dynamic programming algorithm.

The temporal results are very encouraging : the average speed-up factor for grey-level 256 × 256 images matching performed with 50 MHz XC 6462 , when compared with the Pentium 200 MHz PC, is 2000.

Further speed-up improvements can be obtained with faster clock, VLSI or analog circuit implementation (in the silicon, AsGe, ...), or by adding new constrains on images (such as stereo, for example).

Some circuit cost minimisation considerations were outlined as well. Further architectural optimisations are under investigation.

An extension of the proposed approach for matching of any n dimensional separable signals is straightforward.

Acknowledgement.

We thank the ETCA, a Defence Research Laboratory, France for the access to the CM−5.
We thank the French Ministry of Foreign Affairs for its supporting of the presented project.

6. Bibliography.

[Bel57] Bellman, R., *Dynamic programming*, Princeton University Press, 1957

[Ben92] A. Bensrhair, *Contribution à la réalisation d'un capteur de vision 3D par stéréovision passive*, Thèse de doctorat, Université de Rouen, 1992

[Cor95] Cormen, *at al.*, Introduction to Algorithms, MIT 1995

[LeC97] F. Le Coat, E.Pissaloux, P.Bonnin, T. Garié, F. Durbin, A. Tissot, *A Parallel Algorithm for a Very Fast Velocity Field Estimation*, IEEE ICIP'97 International Conference on Image Processing, Santa Barbara, USA, October 26-29 1997, vol. II pp. 179-183

[Pis96] Pissaloux E., Le Coat, F., Bonnin, P., Bezencenet G., Durbin, F., *A Parallel Method for Matching of Aerial Images*, SPIE's Int. Symp. on Intelligent Systems & Advanced Manufacturing, Boston, USA, November 18−22, 1996, vol. 2904, pp. 75−81

[Qué92] G. M. Quénot *The "Orthogonal Algorithm" for Optical Flow Detection using Dynamic Programming*, IEEE International Conference on Accoustics, Speech and Signal Processing, San Francisco, CA, USA, March 1992.

[Sor94] Sorel, Y., *Massively Parallel Computing Systems with Real Time Constrains − the Algorithm Architecture Adequation*, Proc. of the IEEE Conference on Massively Parallel Computing Systems, pp. 282-294, Ischia, (Italy), 2-6 May , 1994, pp. 44−53

[Wu87] Maître, H., Wu, Y., Improving dynamic programming to solve image registration, Pattern Recognition, 20(4), pp. 443−462,1987.