

# Efficient Motion Vector Refinement Architecture for Sub-Pixel Motion Estimation Systems

Tiago Dias  
DEETC-ISEL / INESC-ID  
Lisbon, Portugal  
E-mail: tiago.dias@inesc-id.pt

Nuno Roma  
DEI-IST / INESC-ID  
Lisbon, Portugal  
E-mail: nuno.roma@inesc-id.pt

Leonel Sousa  
DEEC-IST / INESC-ID  
Lisbon, Portugal  
E-mail: las@inesc-id.pt

**Abstract**—This paper proposes a new, scalable and efficient VLSI architecture for real-time sub-pixel motion estimation. The proposed structure is optimized for search strategies using small search ranges, such as hierarchical or sub-pel refinement algorithms. Based on the proposed architecture, a highly modular and configurable motion estimation co-processor capable of estimating optimal motion vectors with any given accuracy and using any known interpolation algorithm is presented. The performance of this processing structure was evaluated by embedding it in a two-level motion estimation system with minimum memory bandwidth requirements, that estimates half-pixel accurate motion vectors using a two-step search procedure. Experimental results for implementations on ASIC and FPGA devices show that by using the proposed architecture it is possible to estimate motion vectors up to the 4CIF image format, in real-time with any given sub-pixel accuracy.

## I. INTRODUCTION

Motion compensation (MC) is a fundamental video coding technique, that exploits temporal redundancy between consecutive frames in video sequences in order to maximize the subjective quality of the coded video for a given bit rate [1]. Although the set of motion vectors used by the MC module can be estimated using any known motion estimation (ME) algorithm, block-matching schemes [2] are the most used in video coding, mostly due to its low complexity and regularity. In this approach, the current frame is divided into blocks of pixels (macroblocks) and a motion vector (MV) is estimated for each macroblock. The MV coordinates define the displacement between the macroblock under processing and the best matching block of pixels defined within a search region in a search frame using a given cost function, such as the Sum of Absolute Differences (SAD) [3].

Most video coding applications estimate MVs with integer-pixel accuracy (IPA), where the candidate blocks are spaced by an integer number of pixels. However, in both natural and synthetic video sequences, the true frame-to-frame displacement of moving objects is rarely an integer number of pixels [4]. Consequently, in order to efficiently exploit temporal redundancy, ME should be performed over a search area with greater pixel resolution, where MVs may point to candidate blocks placed at half, quarter or even eighth-pixel locations of the search area.

Any of the existing optimal or sub-optimal ME algorithms can be used to estimate MVs with sub-pixel accuracy (SPA) at the whole sub-pixel level. Nevertheless, the efficiency of these

algorithms is seriously degraded due to the excessive storage requirements and the need for a large number of computations [2]. Furthermore, the implementation of such algorithms in hardware structures leads to circuits with high latency, high power consumption and low operating frequencies, which are all key factors towards a real-time implementation. On the other hand, ME with SPA requires the use of more complex, powerful and thus more expensive CoDecs. That is why most modern video coding standards (H.263, MPEG-2, H.264 and MPEG-4/AVC) have adopted ME with SPA only as an optional and extended part of the standard, in order to comply with the low bit rate requirements of modern mobile and portable applications. As a consequence, a two-step search approach is usually preferred for estimating MVs with SPA [1], [2]. In this approach, the best matching block at integer locations of the search area is found in a preliminary step of the search procedure. Then, in a second step, the search area surrounding the selected integer candidate block is interpolated into a higher resolution and the integer-pixel MV is refined into sub-pixel accuracy.

This paper proposes a new, efficient and highly scalable VLSI architecture for real-time ME, optimized for sub-pixel resolution. This innovative structure not only provides minimum latency, maximum throughput and a full utilization of its computational units, but also requires few hardware resources for its implementation. Moreover, the proposed architecture only requires three sets of data for its operation: *i*) the initial coarser MV coordinates, that can be pre-computed in any other hardware or software application; *ii*) the search area pixels surrounding that location; and *iii*) the current macroblock pixels. Consequently, this novel architecture can be used both in hardware or in hybrid software-hardware video coding systems, either to improve the accuracy of the ME process or to estimate local motion based on MVs predicted from previous frames.

The rest of this paper is organized as follows. Section II presents the proposed VLSI architecture for real-time sub-pixel accurate ME. Section III introduces a parameterizable sub-pixel ME co-processor based on the proposed ME architecture, that refines coarser MVs into sub-pixel accuracy. This co-processor can be implemented with any generic interpolation algorithm. Section IV describes an implementation of a complete ME system that estimates MVs with half-pixel accuracy using a two-step search procedure. This system uses a generic

ME structure with integer-pixel accuracy in its first stage and the sub-pixel ME processor mentioned above in the latter one. Experimental results concerning implementations on ASIC and FPGA devices of the SPA ME co-processor are presented in Sections III and IV, respectively. Section V concludes the paper.

## II. SUB-PIXEL ACCURATE MOTION ESTIMATION ARCHITECTURE

Although several different architectures have been proposed to estimate sub-pixel accurate MVs, only a few of them satisfy the previously mentioned requisites [5], [6]. Nonetheless, most of such structures are based on sub-optimal search algorithms, which makes its hardware implementation difficult and rather inefficient, mainly due to the non-regular processing and complex control schemes required by sub-optimal algorithms. To avoid these problems, it is often preferable to use regular processing structures in hardware implementations, such as systolic array architectures, that provide maximum throughput and require simpler control units.

From a comparative analysis of the main array structures that have been proposed for ME, it is possible to conclude that the type-II architecture proposed by Vos [7] is one of the structures that best minimizes the hardware requirements and provides a constant and relatively low processing time for small search ranges ( $p \leq \frac{N}{2}$ , where  $N$  represents the macroblock width and  $p$  the maximum displacement of the candidate blocks in each direction of the search area). Consequently, this architecture proves to be well suited for refining the precision of pre-computed IPA MVs into half ( $p = 1$ ), quarter ( $p = 3$ ) or even eighth-pixel accuracy ( $p = 7$ ), using macroblocks with  $8 \times 8$  ( $N = 8$ ) or  $16 \times 16$  pixels ( $N = 16$ ), the more frequently adopted parameters by the ITU-T H.26x and ISO MPEG-x video coding standards. However, considering that this architecture was firstly designed to estimate IPA MVs, several changes have to be performed in its internal structure so that it can be used in sub-pixel refinement algorithms. As a result, the proposed SPA ME architecture is based on the type-II architecture proposed by Vos [7], but presents significant improvements in what concerns its structure. Moreover, it also presents several optimizations that improve its efficiency in terms of hardware resources, power consumption and throughput.

Just like in any other type-II structure [8], in the proposed architecture the cost functions for all candidate blocks are computed concurrently within the processing array. Denoting  $k$  as the SPA factor ( $k = 2$  for half-pixel,  $k = 4$  for quarter-pixel, and so on), on each processing cycle each of the  $(2k - 1)^2$  processing elements (PEs) that compute the cost function, designated by *active PEs*, is fed with the same pixel of the current macroblock and a different search area pixel corresponding to the same relative location for each of the candidate blocks being processed. Since in type-II architectures the pixel rate is equal to the clock rate, after  $N^2$  clock cycles there will be  $(2k - 1)^2$  cost functions simultaneously available at the active PEs outputs. To obtain the optimum MV, i.e., the one holding the lowest cost function, these values are subsequently compared against each other in a binary tree comparator.

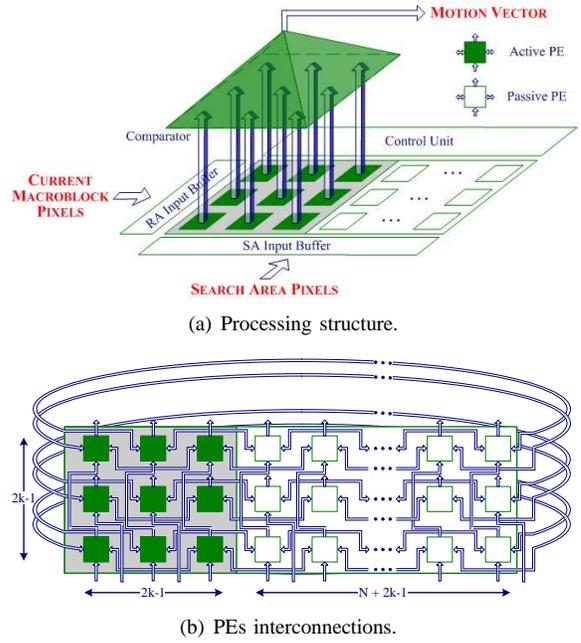


Fig. 1. Proposed type-II architecture for SPA ME, considering half-pixel accuracy ( $k = 2$ ).

Hence, provided that data is always available to feed this systolic processing structure, a MV can be estimated in just  $N^2 + \log_2 \left[ (2k - 1)^2 \right]$  clock cycles. In Fig. 1(a) it is presented the block diagram for the case of half-pixel accuracy ( $k = 2$ ).

To reduce the power consumption of the proposed type-II structure, the active PEs include an extra module that dynamically controls the computation of the cost function. This circuit consists on a standing-data register to store a threshold value, such as the cost function of the best integer-pixel candidate block, and comparison circuitry to disable the computation of the current cost function whenever the partial accumulated value exceeds the threshold one. In such situation, the threshold value is outputted by the active PE and fed into the binary tree comparator for comparison with the remaining cost function values.

To further optimize the estimation, the proposed type-II architecture uses a set of  $(2k - 1) \times (N + 2k - 1)$  PEs, designated by *passive PEs*, to store and displace the search area pixels of the candidate blocks under processing. Unlike other type-II architectures that only provide integer-pixel accuracy, all PEs of the proposed structure are connected to each other in an interleaved manner in groups of  $N$  in the horizontal direction and in groups of  $k$  in the vertical direction, being spaced apart by  $k - 1$  PEs in both directions. Such regular interconnections are required to guarantee that the search area pixels are displaced over a sampling grid with  $\frac{1}{k}$ -pixel resolution within the processing array. Furthermore, to minimize the hardware requirements of the proposed architecture, i.e., to avoid redundant passive PEs, the processing array was also displaced over a cylindrical structure by connecting the passive PEs located in the right margin of the array to the active PEs in its left margin. It should also be noted that unlike the Vos type-I architecture [7], in the proposed structure there are few PEs in each row of the array ( $N + 3$  PEs for half-pixel accuracy). As

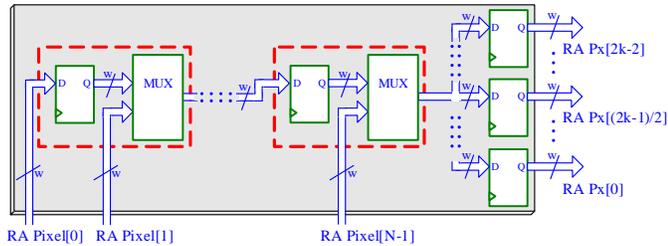


Fig. 2. RA input buffer of the proposed sub-pixel ME architecture.

a result, the delays associated to the longest interconnections at the vertical borders of the array do not influence the critical path of the circuit [9].

By using the interconnection scheme described above, and illustrated in Fig. 1(b), search area pixels are displaced in three different directions within the processing array: upwards, to the right and to the left. Whenever a set of  $k$  lines of search area pixels is fed into the array through its lower inputs, all search area pixels within the structure are simultaneously shifted one position upwards. In the subsequent  $N - 1$  clock cycles search area pixels are shifted to the right, one position per clock cycle. As soon as all these  $N - 1$  shift-right operations have been carried out, the search area data is shifted upwards once again and a new set of  $k$  lines of search area pixels is fed at the bottom of the array through its lower inputs. During the next  $N - 1$  clock cycles, search area data is shifted to the left in a similar manner as described above, being shifted upwards after the  $(N - 1)^{th}$  clock cycle. Hence, by using such zig-zag processing scheme and the adopted cylindrical structure it is possible to maximize the efficiency of the proposed architecture: all PEs are kept busy at any time instant and it avoids both redundant accesses to the frame memory and the need for dummy clock cycles between any two adjacent rows of search area pixels.

To overcome the problem of broadcasting the reference area (RA) pixels to all the active PEs in the array, these pixel values are loaded into the active PEs through a specially designed input buffer. To achieve such goal, the proposed circuit is composed by  $N$  cascaded registers that are loaded in parallel with an entire line of reference area pixels. Such pixels are then transferred serially to the active PEs through a set of  $2k - 1$  redundant registers, as depicted in Fig. 2. Each of these output registers contains a copy of the pixel value to be fed to the active PEs, but is only responsible for feeding the active PEs of a specific line of the processing array, thus minimizing the required fan-out for the RA input buffer.

### III. SUB-PIXEL MOTION ESTIMATION CO-PROCESSOR

The majority of ME architectures that have been recently proposed in the literature are fairly efficient and present reasonable performance results [8]. Nevertheless, most structures only provide integer-pixel accuracy, which is not enough to comprise with the most recent video coding requirements. Consequently, several modifications have to be performed on these structures to accomplish such requisites. However, by changing the functionality of existing modules or by embedding new ones in running ME systems, it frequently implies

the redesign of the entire system architecture, thus significantly increasing the development costs. A more feasible approach to develop ME systems with sub-pixel accuracy considers the application of pixel-refinement search strategies, by combining the use of these processing structures with sub-pixel ME co-processors. However, in most cases this integration is not so trivial, being quite complex and difficult to implement. To avoid such problems and to ease the development of customizable sub-pixel accurate ME systems, a new modular and fully configurable sub-pixel ME co-processor is proposed in this section.

The proposed circuit is based on the highly efficient type-II processing structure presented in Section II and is characterized by a very simple and efficient interface, requiring only three sets of data for its operation: *i*) the initial coarser integer accuracy MV coordinates; *ii*) the search area pixels surrounding the corresponding best candidate block location; and *iii*) the current macroblock pixels. By using such a straightforward interface, the proposed sub-pixel ME co-processor can be used not only in hardware implementations but also in hybrid software-hardware video coding systems.

#### A. Architecture of the co-processor

The proposed sub-pixel ME co-processor consists of four functional blocks and a control unit, as shown in Fig. 3. It is capable of refining MVs to any accuracy, using whichever interpolation algorithm and embedded in any hardware or hybrid software-hardware ME system. To do so, the proposed ME co-processor, which is based on the efficient type-II processing array presented in Section II, was designed using a highly flexible, scalable and modular architecture and makes use of simple and efficient interface protocols to transfer the data between its several units. Furthermore, a pipeline processing scheme was also adopted to interconnect the several modules of the system. As a result, not only does the proposed architecture maximize the data throughput but it also provides an easy integration and development of new and different functional blocks.

The input buffers of the proposed ME co-processor were designed to retrieve all the required pixels from the main (or an intermediate) frame memory, either in line-scan or block-scan mode [7]. To increase the flexibility of the architecture, the search area input buffer can be fed either with all the pixels of the lower resolution search area, or only with the subset

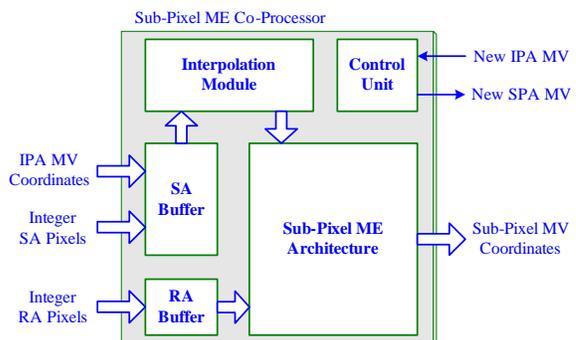


Fig. 3. Simplified block diagram of the proposed sub-pixel ME co-processor.

of pixels required by the interpolation module. Nonetheless, when the first alternative is adopted, the buffer must also be fed with the coarser MV coordinates, in order to select the subset of pixels to be outputted to the interpolation module.

Due to the numerous interpolation algorithms that have been proposed in the literature in the latter years [4], [10], and since video coding standards do not specify the algorithm to be used in the interpolation process, the interpolation module was designed using a scalable architecture in order to allow the implementation of any generic algorithm and the reusability of different interpolation modules. Hence, depending on the required accuracy extension and hardware constraints, the proposed interpolation module may compute higher resolution blocks of the original search area surrounding the best candidate block found at a lower resolution with any given precision.

### B. Implementation and experimental results

To assess and evaluate the performance of the proposed MV refinement structure, the sub-pixel ME co-processor was completely described using both behavioural and fully structural parameterizable IEEE-VHDL. Several setups of these descriptions were synthesized and implemented using two different technologies, ASIC and FPGA, considering the typical parameters currently adopted in videoconferencing applications [11]: half-pixel accuracy and the bilinear interpolation algorithm [2]. Such setups consider a straightforward implementation of a high throughput 4-tap FIR filter with low hardware requirements in the interpolation module, 8-bit to represent the pixel values and macroblocks with  $8 \times 8$  ( $N = 8$ ) and  $16 \times 16$  pixels ( $N = 16$ ), which corresponds to the set of parameters more frequently adopted by the ITU-T H.26x and ISO MPEG-x video coding standards.

Table I(a) presents the results obtained for two implementations using a general purpose FPGA, the *Xilinx* Virtex-E XCV3200E-7 device [12], and the *Xilinx* Synthesis Tool from ISE 6.1.3i. The hardware resources required by the proposed sub-pixel ME co-processor were assessed in terms of the percentage of CLB slices and LUTs that were required for each implementation and evidence the very low hardware requirements of the proposed ME circuit. Moreover, considering that in the proposed structure the pixel rate equals the clock rate, the maximum operating frequency values depicted in Table I evidence that the proposed structure is able to refine MVs into half-pixel accuracy in real-time for the 4CIF image format.

The implementation in an ASIC was performed using

TABLE I

IMPLEMENTATION RESULTS FOR THE SUB-PIXEL ME CO-PROCESSOR.

N	F [MHz]	# Slices	# LUTs
8	141.4	949	1736
16	138.8	1712	3197

(a) Using the *Xilinx* XCV3200E-7 FPGA device.

N	F [MHz]	Area [ $\mu m^2$ ]	Power [mW]
8	408.16	265768.46	399.65
16	393.70	621455.86	801.57

(b) Using the StdCell library from *UMC*.

*Synopsys* synthesis tools and a high density StdCell library from *UMC*, which is based on a  $0.13 \mu m$  CMOS process from *Virtual Silicon Technology Inc* [13]. Table I(b) summarizes the main characteristics of the implemented circuits for the two setups mentioned above. The obtained results evidence that by using the ASIC technology it is possible to operate at a frequency that is about 2.8 times faster than the one obtained with the FPGA, thus allowing the estimation of MVs with half-pixel accuracy in real-time for the 16CIF image format.

From the experimental results presented for the two considered technologies, it is also possible to conclude that more accurate MVs can be obtained using the proposed sub-pixel ME co-processor by just increasing the number of PEs in its processing array. Due to the pipeline systolic structure of the proposed single array architecture, the increase in the number of PEs will only change the latency of the circuit, owed to an augment in the number of levels of the binary tree comparison unit. Consequently, from the experimental results presented in Table I, it is possible to conclude that the proposed sub-pixel ME co-processor is also able to estimate MVs with quarter or eighth-pixel accuracy in real-time for any x-CIF image format, with only a minor increase of its latency.

Experimental results concerning the power consumption of the proposed sub-pixel ME co-processor were obtained using the *Synopsys* synthesis tools for implementations using the same setups mentioned above and operating at a normalized operating frequency of 100MHz. The obtained values,  $394.79 mW$  and  $801.41 mW$  for the configurations using  $N = 8$  and  $N = 16$ , respectively, correspond to the worst case situation, where all active PEs are kept computing the cost function for all  $N^2$  iterations. However, it should be noted that such scenario hardly ever happens in practice, due to the control circuitry embedded in each active PE, that disables the computation of the cost function whenever the partial value exceeds a given threshold (usually, the value of the cost function for the integer-pixel MV). In such situation, the power consumption of the active PEs becomes equal to the one of the passive PEs for the remaining iterations of the algorithm.

To estimate an average value of the percentage of iterations that are likely to be avoided in the computation of the cost function, several QCIF benchmark video sequences, characterized by different spatial detail and amount of movement, were coded in interframe mode with integer-pixel accuracy using the H.263 video encoder provided by Telenor R&D [14]. Fig. 4 depicts the estimates of the power saving rates obtained for the five considered benchmark video sequences taking into account the power consumption values of all functional units of the proposed sub-pixel ME architecture, i.e., the search and reference area input buffers, the hierarchical comparator, the control module and the PE array.

The obtained power consumption values and power saving rates demonstrate that the proposed sub-pixel ME co-processor imposes low power constraints when compared with other ME structures [15]. Furthermore, the experimental results presented in Fig. 4 evidence that significant power saving rates can be obtained, except for video sequences with high spatial detail and amount of movement.

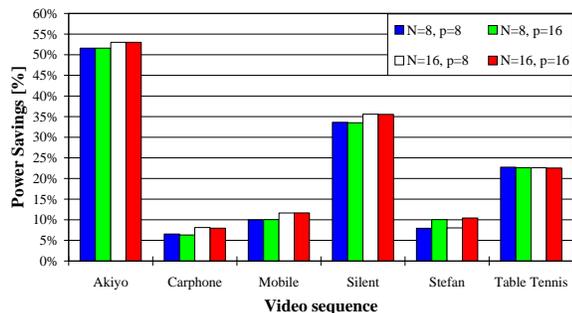


Fig. 4. Power savings obtained by using the enhanced active PEs in the processing array of the proposed sub-pixel ME co-processor.

#### IV. INTEGRATION OF THE SUB-PIXEL ME CO-PROCESSOR IN A COMPLETE ME SYSTEM

To validate the functionality and assess the performance of the proposed sub-pixel ME co-processor in a practical realization, a complete ME system was developed and implemented in hardware. The proposed system estimates MVs with sub-pixel accuracy using a two-step search procedure, where the MV coordinates are firstly estimated at a coarser integer resolution and subsequently refined into half-pixel accuracy by using the sub-pixel ME co-processor described in Section III.

##### A. Architecture of the ME system

The proposed ME system consists of a fully parameterizable modular architecture, that implements the two-steps of the adopted pixel refinement search strategy in two distinct and independent processing units: one that estimates the MVs with integer-pixel accuracy and a second one that refines these MV coordinates into sub-pixel resolution. The IPA ME module is based on an efficient class of bi-dimensional processing structures for FSBM ME that was recently proposed in the literature [9], [16]. The SPA ME module consists of an implementation of the sub-pixel ME co-processor presented in Section III, considering half-pixel accuracy and using the bilinear interpolation algorithm.

To optimize the data processing, i.e., to maximize the throughput and minimize the latency, all processing units in the proposed ME system work concurrently. Furthermore, simple and fast communication protocols are used to interconnect all the modules within the system architecture and to synchronize the several processing units, therefore minimizing the number of processing stalls that may arise due to data dependencies.

The memory bandwidth requirements of the main frame memory are also minimized in the proposed ME system, by reusing and transferring the data between the several modules of the architecture. To achieve such goal, a dedicated and highly efficient data reuse module was also developed and embedded in the proposed ME system [16]. Such processing unit is mainly composed by data register banks and plays two key roles within the proposed ME system. On the one hand, it stores in local memory the set of reference and search area pixels already processed by the IPA ME module and that may be required by the SPA ME module, so as to avoid redundant accesses to the main picture memory. Furthermore, based on the obtained IPA MV coordinates, it also selects the subset

of integer-resolution search area pixels that should be sent to the SPA ME module. On the other hand, the proposed data reuse module was also used to minimize the huge number of memory accesses of the IPA ME module [17], thus further reducing the memory bandwidth requirements of the entire ME system. To achieve such objective, the proposed data reuse module increases the data reuse level of the IPA ME module to data reuse level C as described by Tuan [18], which provides the best trade-off in terms of memory bandwidth and local memory size, by avoiding redundant accesses to the main frame memory for two adjacent integer-resolution search areas. Such redundant accesses are avoided by composing the lines of search area pixels entering the IPA processing array with data stored in the local memory bank of the data reuse module and/or retrieved from the main frame memory (new pixels).

Consequently, within the proposed ME system, data is retrieved from the main frame memory solely by the data reuse module (search area pixels) and by the IPA ME module (reference area pixels). The subset of search area pixels surrounding the best candidate block at the coarser resolution, required for the interpolation algorithm, is then selected by the data reuse module, as a function of the IPA MV coordinates, and bypassed to the sub-pixel ME co-processor. Within the sub-pixel ME co-processor, such pixels are firstly used to compute the higher resolution search area in the interpolation unit and are subsequently transferred to the processing array. This array is also fed with the current macroblock pixels, formerly used to compute the coarser MV and stored in the data reuse module, in order to obtain the final estimate of the MV with half-pixel accuracy.

##### B. Implementation and experimental results

To validate the functionality of the described ME system with sub-pixel accuracy, a methodology entirely similar to the one that was adopted for the implementation in FPGA devices of the sub-pixel ME co-processor, proposed in Section III, was taken. As before, the set of parameters commonly adopted in videoconferencing applications, and described in Subsection III-B, were also adopted to synthesize and implement in FPGA devices several setups of the IEEE-VHDL descriptions of the proposed ME system.

Table II presents the experimental results obtained for two

TABLE II  
IMPLEMENTATION RESULTS FOR THE PROPOSED ME SYSTEM USING THE Xilinx VIRTEX-E XCV3200E-7 FPGA DEVICE.

Module	F [MHz]	# Slices	# LUTs
IPA ME	129.4	5131	5572
Data reuse	174.8	2101	1213
SPA ME	141.4	949	1736
<b>ME system</b>	129.4	9054	9490

(a)  $N = p = 8$ .

Module	F [MHz]	# Slices	# LUTs
IPA ME	124.1	20372	22493
Data reuse	174.8	7335	1507
SPA ME	126.9	1712	3197
<b>ME system</b>	124.1	30868	29573

(b)  $N = p = 16$ .

implementations of the proposed ME system using a *Xilinx* Virtex-E XCV3200E-7 device. Such results demonstrate that by using the proposed ME system it is possible to estimate MVs with half-pixel accuracy in real-time (30 fps) for the 4CIF image format. Moreover, the obtained results also evidence that the performance of the proposed ME system is solely constrained by the adopted IPA ME module. This fact was already expected and is owed both to the modular interconnection and to the pipeline alike cascade processing schemes implemented in the proposed ME system.

In what concerns the memory bandwidth requirements, Fig. 5 depicts the reduction of the number of accesses to the main frame memory obtained by using the proposed data reuse module, as well as the corresponding increase in the amount of hardware resources required to implement each of the synthesized setups. From the presented chart one can conclude that for search area ranges greater than twice the macroblock width ( $p \geq 2 \times N$ ), the increase of the involved hardware requirements does not justify the use of the proposed data reuse module in systems with strict hardware constraints.

## V. CONCLUSION

A new VLSI type-II architecture for ME in video sequences, capable of estimating motion vectors in real-time with any accuracy, was proposed in this paper. This structure proves to be highly suitable either for recursive or small search range estimation strategies, such as hierarchical or sub-pixel refinement algorithms. Moreover, not only does it provide maximum throughput and a full utilization of the hardware resources, but it also presents minimum hardware requirements.

An efficient modular and configurable sub-pixel ME co-processor, that can be used in both hardware and hybrid software-hardware ME systems to refine the precision of coarser resolution MVs, was also presented in this paper. This structure is based on the proposed type-II architecture and is able to retrieve reference and search area pixels from either the main picture memory or from the previous stages local memories, thus avoiding redundant accesses to the main frame memory. Furthermore, the highly modular and flexible architecture of this circuit provides an easy integration of each of its modules, thus allowing the implementation of different interpolation algorithms.

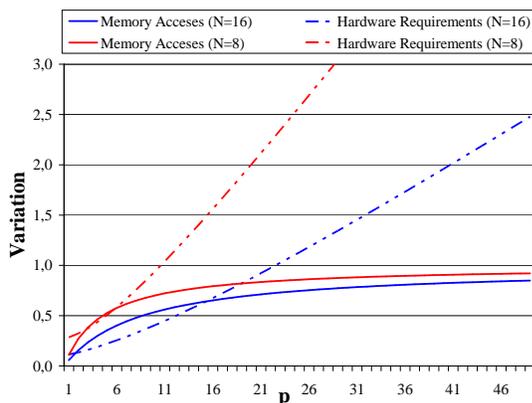


Fig. 5. Memory accesses and required registers for the proposed ME system.

Experimental results obtained from implementations on ASIC and FPGA devices of a two-level half-pixel motion estimation system, using the proposed sub-pixel ME circuit in its last stage and the more frequently adopted parameters of the ITU-T H.26x and ISO MPEG-x video coding standards ( $N = p = 8$  and  $N = p = 16$ ), have shown that by using such circuits it is possible to estimate MVs with half-pixel accuracy up to a rate of 30 fps for high-quality video (16CIF format).

## ACKNOWLEDGMENT

This work has been supported by the POSI program and the *Portuguese Foundation for Science and for Technology* (FCT) under the research project *Configurable and Optimized Processing Structures for Motion Estimation* (COSME) POSI/CHS/40877/2001.

## REFERENCES

- [1] P. M. Kuhn and K. P. M., *Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation*. Norwell, MA, USA: Kluwer Academic Publishers, 1999.
- [2] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards: Algorithms and Architectures*, 2nd ed. Kluwer Academic Publishers, June 1997.
- [3] S. Vassiliadis, E. A. Hakkennes, J. S. S. M. Wong, and G. G. Pechanek, "The sum-absolute-difference motion estimation accelerator," in *Proceedings of the 24th Euromicro Conference*, vol. 2, Aug. 1998, pp. 559–566.
- [4] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 604–612, Apr. 1993.
- [5] F.-H. Cheng and S.-N. Sun, "New fast and efficient two-step search algorithm for block motion estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 7, pp. 977–983, Oct. 1999.
- [6] J. C. B. Zhou, "A fast two-step search algorithm for half-pixel motion estimation," in *Proceedings of the IEEE International Conference on Electronics, Circuits and Systems (ICECS03)*, vol. 2, Sharjah - U.A.E, Dec. 2003, pp. 611–614.
- [7] L. Vos and M. Stegherr, "Parameterizable VLSI architectures for the full-search block-matching algorithm," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 10, pp. 1309–1316, Oct. 1989.
- [8] P. Pirsch, N. Demassieux, and W. Gehrke, "VLSI architectures for video compression - a survey," *Proceedings of the IEEE*, vol. 83, no. 2, pp. 220–246, Feb. 1995.
- [9] N. Roma and L. Sousa, "Parameterizable hardware architectures for automatic synthesis of motion estimation processors," in *Proceedings of the IEEE Workshop on Signal Processing Systems - Design and Implementation (SiPS'01)*, Antwerpen - Belgium, Sept. 2001, pp. 428–439.
- [10] D. Wang, "Comparison of motion-compensated algorithms for frame interpolation," *Optical Engineering*, vol. 42, pp. 586–590, Feb. 2003.
- [11] *ITU-T Recommendation H.263 - "Video Coding for Low Bitrate Communication"*, ITU-T, Feb. 1998.
- [12] *Virtex-E 1.8V Field Programmable Gate Arrays Datasheet*, v2.3 ed., Xilinx Inc., July 2002.
- [13] *UMC High Density Standard Cells Library - 0.13μm CMOS process*, v2.3 ed., Virtual Silicon Technology Inc., Dec. 1999.
- [14] Telenor, TMN (Test Model Near Term) - (H.263) encoder/decoder (source code), v2.0 ed., Telenor Research and Development, June 1996.
- [15] S. Yang and W. Wolf, "Power modeling of motion estimation VLSI architecture," in *Proceedings of the The 36th International Symposium on Microarchitecture (MICRO-36)*, San Diego, CA - USA, Dec. 2003.
- [16] T. Dias, "Dedicated processors for motion estimation in video sequences," Master's thesis, Instituto Superior Técnico, Lisbon, Sept. 2004.
- [17] S. Saponara and Fanucci, "Data-adaptive motion estimation algorithm and VLSI architecture design for low-power video systems," *IEE Proceedings-Computers And Digital Techniques*, vol. 151, pp. 51–59, Nov. 2004.
- [18] T.-S. C. Jen-Chieh Tuan and C.-W. Jen, "On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 1, pp. 61–72, Jan. 2002.