MOTION COMPENSATED DE-INTERLACER

Daniele Bagni *PierLuigi lo Muzio*
Paola Carrai *Vittorio Riva* *Stefano Vedani*

Philips S.p.A, Philips Research Monza
Via Philips, 12, 20252 Monza (MI), Italy

e-mail: bagni@monza.research.philips.com Tel. +39.39.203.7804
e-mail: lomuzio@monza.research.philips.com Tel. +39.39.203.7809

Abstract

This paper describes a real time hardware prototype for the de-interlacing of Standard Definition video signals. The line rate is doubled in order to remove specific artefacts of interlaced signals, like interline flicker and line crawl. The hardware applies Digital Signal Processing (DSP) in order to achieve a high performance sequential scan conversion of interlaced signals. The programmability of the prototype gives the possibility to use it as a basis for evaluations of real time line-rate conversion algorithms for both 50 Hz or 60 Hz video sequences.

1 INTRODUCTION

Because of the limited bandwidth, the current European broadcasting standards have adopted an interlaced scan. Although interlacing is a useful method of bandwidth reduction, it can degrade the picture quality due to intrinsic aliasing and, in addition to the loss of vertical resolution, it can generate the so called flicker artefacts: the "line flicker", principally noticeable in those areas of the image containing high vertical frequencies and the "line crawl", mainly visible when some nearby object moves with a particular vertical velocity[1] .

A proper format conversion from interlaced to progressive scan, as the de-interlacing, could reduce flicker artefacts and could improve the vertical resolution of the displayed images. Furthermore, because of the wide variety of more or less incompatible images sequences formats which have to be displayed on any kind of monitor, de-interlacing becomes the starting and also the most critical step for every format conversion. The tape to film transfer and the home theatre are examples of professional and consumer applications of the de-interlacing.

The transmission of interlaced pictures will probably continue for a long time, therefore a high quality receiver needs to remove the flicker artefacts.

Philips Research Monza (PRM) laboratory designed a real time de-interlacer, using a 19 inches rack, containing eleven double-extended Euro-sized boards. In order to perform an excellent conversion even in the case of moving pictures, motion compensated interpolation is applied. In fact the prototype implements a recursive motion compensated interpolation algorithm including two main DSP sections:

- the motion estimator, which carries out the computation of the motion vectors with sub-pixel accuracy through a sophisticated algorithm named "3-Dimensional Recursive-Search Block-Matching";
- the motion compensated interpolator, which calculates the intermediate lines by time-recursive interpolation along the displacement direction of the moving objects, with sub-pixel accuracy too.

Both these techniques have been developed in [1, 2, 3].

The organisation of this paper is the following. Section 2 presents the architecture of the hardware prototype. Section 3 and 4 describe respectively the motion estimation and the motion compensated interpolation algorithms as well as their hardware implementation. Finally, testing procedures are reported in Section 5 and some conclusions are drawn in Section 6.

[1] The velocity is around one frame line per one field period. Even with homogoneous brightness areas the scanning lines can be interpreted as moving upwards or downwards with the indicated velocity.

2 THE HARDWARE PROTOTYPE

The prototype (see Fig.1) has been designed and realised using digital video processing programmable boards. The video processing element core is the Philips VSP, a general purpose programmable "Video Signal Processor", particularly suited for flexible and design-cost effective implementation of real time video systems. The processor architecture is modular, each module comprises ten parallel operating and fully pipelined processing units, individually controlled by cyclostatic programs. The VSP can handle several signals with different sampling rates, like luminance, chrominance and synchronisation, simultaneously [4].

"Philips VSP8-flexboards" have been exploited for some DSP parts: on each board there are eight VSPs working in parallel. The VSP8-flexboards are accountable for the following functions: vertico-temporal median filtering (MEDIAN), vector block structure post-processing (BLOCK-POSTP), luminance horizontal (HOR-FILT) and vertical (VERT-FILT) filtering.

For other parts, requiring more memory, a new programmable board, called VSPRAM, has been designed. The VSPRAM board can implement several functions: field stores, vector shifters, sub-pixel interpolation, line compression. Several technologies are combined in a proper way: Xilinx FPGA for memory address generation, Altera EPLD for control signals generation, Philips VSP for video processing. In the de-interlacer prototype it is used for the following purposes: motion estimation (Y1ME and Y2ME), chrominance interpolation (CHROMA), luminance motion compensated interpolation (MOT-COMP), luminance line compression (LN-COMPRES).

Fig. 1 shows also two additional boards for analogue (AD-DA) and digital input-output (DIG-IO).

3 THE MOTION ESTIMATOR

The DSP section of motion estimation is composed by Y1ME, Y2ME and BLOCK-POSTP boards, as shown in Fig. 1. Both Y1ME and Y2ME utilise recursive block matching with only four candidate vectors per block and calculate in parallel their own displacement best vector, but each one uses a different set of predictors and updates along an appropriate convergence direction.

The final best vector is selected by BLOCK-POSTP and stored in a proper prediction memory. The main task of the board is however the elimination of blocking effects in the motion compensated interpolation process by reducing the block dimensions of the motion vectors. Moreover it calculates a parameter to check the case of motion vectors inconsistencies, called "smoothness figure"; this indicator will be used in the motion compensated interpolation DSP section.

The features explained in [1] and [3] - such as bi-directional convergence, temporal convergence accelerators, asynchronous cyclic search strategy, a quarter pixel accuracy in both horizontal and vertical direction, block erosion post processing - produce an extremely coherent and smooth vector field particularly suitable for interlaced to progressive scan-conversion.

4 THE MOTION COMPENSATED INTERPOLATOR

Referring to a couple of following interlaced fields, labelled A and B for simplicity, in the motion estimation process a displacement vector $V_{AB}=(V_X, V_Y)$ is assigned to the block of co-ordinates (i, k) of the current field B=B(x, y, t) if there is the best match with the block of co-ordinates $(i+V_X, k+V_Y)$ of the previous field A(x, y, t-T), where T is the field period and t the current time.

Assuming that A and B respectively take "odd" and "even" position inside the same frame, a new odd field A_{MC} is obtained by motion compensating the former field A, according to:

$$A_{MC}(x, y, t) = A(x-V_X, y-V_Y, t-T) \quad (1)$$

This field is the basis to reconstruct the missing lines of the incoming even field B, by making use of vertico-temporal median filtering:

$$B_{REC}(x, y, t) = \text{median}[\, B(x, y, t), A_{MC}(x, y-1, t), B(x, y-2, t) \,] \quad (2)$$

The progressive output frame B_{OUT} would thus be composed by the original even field B and the reconstructed odd field B_{REC}. On the other hand, in an attempt to modify the frequency spectrum from both type of lines in the same way, not only the new lines, such as B_{REC}, result from an interpolation process, but also the original B lines have to be temporally filtered. As explained in [2], this filtering of original lines helps to suppress alias and it is not necessarily the same as that of the interpolated lines, as shown by the following equation:

$$\begin{matrix}(16-k)\, B_{REC}(x, y, t) + k\, A_{MC}(x, y, t), & \text{reconst. lines} \\ (16-p)\, B(x, y-1, t) + p\, A_{REC}(x, y-1, t), & \text{orig. lines}\end{matrix} \quad (3)$$

where k and p are adaptive parameters controlled by the smoothness indicator.

For high quality results, a quarter pixel accurate interpolation is used, and therefore bi-dimensional high-pass adaptive filtering is performed to eliminate the remaining blurring due to the low-pass filtering side effect of the sub-pixel interpolation itself.

The Eq. (3) reveals also the application a time-recursive scheme of motion compensated interpolation: if

a de-interlaced field is written in a memory it can be used to de-interlace the next incoming field and so on.

The hardware implementation of this refined process is made up by five boards: the MEDIAN for vertico-temporal median filtering (Eq. (3)), the MOT-COMP to motion compensate the luminance signal (Eq. (1)), the HOR-FILT and VER-FILT which act as a bi-dimensional adaptive high-pass filters, and finally the LN-COMPRES. This board performs a line compression: the reconstructed and original lines (13.5 MHz sample rate) are put in a sequential order in the output progressive frame (27 MHz sample rate) before the digital-to-analogue conversion.

The CHROMA board processes the chrominance U and V signals by a vertico-temporal median filter, as the luminance, but without any motion compensation, because the human observer is not very critical regarding chrominances.

5 TESTING PROCEDURES

Generally speaking the test of an hardware prototype requires a great effort and takes a lot of development time, hence it is very important to design the architecture in a modular way, by making use of as less different boards as possible. Except for the digital and analogue input-output service, only two kind of completely re-programmable digital boards were used, the VSPRAM and the VSP8-flexboard, to avoid building a dedicated unit for every function.

Saving time in the hardware debug process (unconnected signal lines, layout mistakes, burnt ICs, etc.) means reserving the main labour to functional testing. As already said, all the video functions were implemented at a firmware level in the programmable VSPs which also have a powerful and comfortable simulation tool, so every functional block was properly "VSP simulated" and compared with the correspondent C code subroutine, exploring all the possible statistical combinations that the input data could have. This comparison was not easy, because the C programs process large video data arrays (the active area of the picture as in the Recommendation CCIR 601) without blanking signals, while the VSP tool simulates the behaviour of the algorithm as it was in real-time by taking sequential data included blanking; so there was the necessity of proper C routines for conversion between this two ways.

After having completely simulated all the VSP boards, an Image Sequencer Processor (ISP) was interfaced to the prototype through the DIG-IO board. This aspect is very important: the ISP can receive video signals and store them as video files; this means that is possible to compare, in an objective way, the simulation results of some C code routines, running the algorithm not in real time, with the results of the hardware stage which is emulating the same algorithm. The ISP could send to the prototype - as well as receive from it - up to four video buses just by changing the placement of the DIG-IO, thanks also to the fact that a particular attention was paid to the layout of the backplane mother-board and to the position of the connections to facilitate the test of the different stages.

The following list explains the testing procedure adopted to guarantee that the hardware prototype emulates perfectly the video processing algorithm mentioned in Section 3-4 (see also Fig. 1):

A) Every board was tested alone by comparing its real time results with the C simulation results, again by exploring all the possible statistical combinations that the input data could have.
B) The time-recursive interpolation section was tested at first without using motion vectors, because although the interpolation is a straightforward thing, the interlace of video signal yield some nontrivial problems and proper delay time memory has to be used to carefully re-synchronise all the signals; it is simpler to perform this action with zero forced vectors. After this step, the input motion vectors - as calculated by the C code simulator- were sent to the interpolator from the ISP, through the DIG-IO board. The motion compensated output pictures were then received by the ISP and compared with the C code simulation results.
C) The recursive motion estimation section was tested by sending it the input signals through the already developed interpolation part, to have the worst condition in terms of hardware stressing. Again the hardware results of this part were compared with the C code software ones.
D) The motion estimation and motion compensated interpolation stages were integrated together taking care of the initial conditions that could affect the algorithm convergence: during the first input field A, when no motion vectors are yet available, the output frame A_{OUT} is produced by intra-field line average and all the memories are properly blanked; then the second output frame B_{OUT} is obtained by median filtering, as in Eq. (2) but without motion compensation; only from the third output frame C_{OUT} the motion compensation is used, to avoid propagation of errors due to interpolation defects.

6 CONCLUSIONS

A time-recursive motion compensated de-interlacer re-programmable hardware prototype working in real time has been realised: it makes use of innovative motion estimation and compensation techniques. The DSP activities of the PRM laboratory related to this theme has also product an HD-TV motion compensated 50 to 100 Hz field rate up-converter hardware prototype [5].

Actually several activities are carried out by the laboratory concerning: Optical Channel Equalisation, Storage Architectures (SMASH European project), Digital

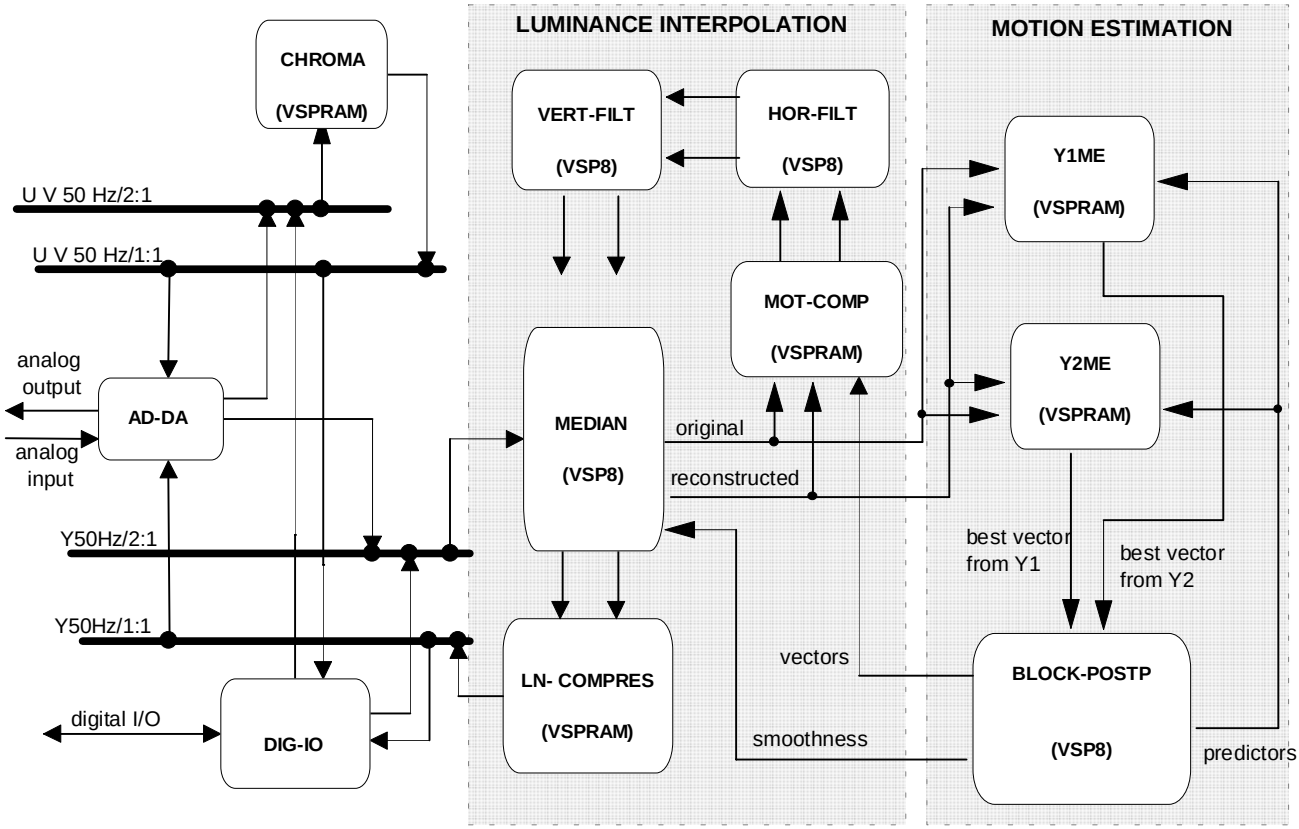

Fig. 1. De-interlacer block diagram: Y, U and V are respectively the luminance and chrominance signals. The data paths sample rate of both the Luminance Interpolation and Motion Estimation parts is 13.5 MHz, while it is 27 MHz for the LN-COMPRES output.

Video Laser Disk Player, Scan Format Conversions, Output Display Modules (VADIS European project), Digital Image Enhancement.

Acknowledgements

The authors are grateful to Mr. G. de Haan at Philips Research Eindhoven for his proofreading of this paper and for his helpful suggestions during the prototype development.

References

[1] G. de Haan and P.W.A.C. Biezen, "Sub-pixel motion estimation with 3-D recursive search block-matching", *Signal Processing: Image Communication* 6 (1994), pp. 229-239.

[2] G. de Haan and P.W.A.C. Biezen, "Time-recursive de-interlacing for high-quality television receivers", *Proc. of International Workshop on HD-TV '95*, Taipeh, Taiwan, November 1995.

[3] G. de Haan. Motion Estimation and Compensation. PhD dissertation, Delft Univ. Tech, September 1992.

[4] A.H.M. van Roermund, P.J. Snijder, H. Dijkstra, C.G. Hemeryck, C.M. Huizer, J.M.P. Schmitz, R.J. Sluijter, "A general-purpose programmable video signal processor", *IEEE Tr. on Consumer Electronics*, Vol. 35, No. 3, August 1989.

[5] D. Bagni and R. Lancini, S. Landi, S. Tubaro, "HD-TV Spatio- temporal up-conversion", *Proc. of International Workshop on HD-TV '94*, Torino, Italy, October 1994.