N. O. Stöffler and G. Färber. An Image Processing Board with an MPEG Processor and Additional Confidence Calculation for Fast and Robust Optic Flow Generation in Real Environments. In Proc. 1997 Int. Conf. on Advanced Robotics (ICAR'97), 845-850, 1997

An Image Processing Board with an MPEG Processor and Additional Confidence Calculation for Fast and Robust Optic Flow Generation in Real Environments

Norbert O. Stöffler and Georg Färber
Laboratory for Process Control and Real-Time Systems
Technische Universität München
D-80333 München, Germany

www.lpr.e-technik.tu-muenchen.de/∼stoffler

Abstract

This paper describes a vision system based on a PC-board which can calculate a sparse but robust optic flow at frame rate. A correlation chip, which was originally designed for MPEG video compression, is used to calculate displacement vectors between blocks of pixels in consecutive frames. The main disadvantage of similar approaches, the noisiness of the displacement field in areas with weak structure, is compensated by a computational inexpensive confidence criterion, which is calculated for each vector in hardware. The performance of the criterion and the improvements in the flow field are demonstrated by experiments.

Keywords

robot vision, optic flow, correlation, block-matching

Motivation

One of the basic problems in robot vision is the detection and measurement of motion. Three-dimensional motion in the real scene induced by moving objects and/or a moving camera results in a two-dimensional velocity field (respectively displacement field in the case of isochronous sampling) on the image plane, according to the equations of optic flow [7]:

$$(1)\quad \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \frac{1}{Z}\begin{pmatrix} T_z x - T_x \\ T_z y - T_y \end{pmatrix} - \omega_x \begin{pmatrix} xy \\ y^2+1 \end{pmatrix} + \omega_y \begin{pmatrix} x^2+1 \\ xy \end{pmatrix} + \omega_z \begin{pmatrix} -y \\ x \end{pmatrix}$$

Before reconstructing the 3D motion, which is a topic by its own, the 2D velocity field has to be estimated from the variations of the illumination on the projection plane. Basically, two approaches for the determination of velocity in the image plane are discussed in literature [1]:

a) The gradient-based approach [3, 11] which depends on the assumption of constant illumination and the evaluation of spatio-temporal derivatives, and

b) the matching approach, which is based on the determination of correspondences between consecutive frames. These correspondences can be found by tracking particular elements belonging to moving objects like edges and corners or by correlating small patches of the images.

The image processing system described in this paper is motivated by the similarity of the matching problem and the so called *motion compensation* defined by the MPEG video compression standards. MPEG uses the spatio-temporal redundancy in an image sequence for bandwidth reduction. If possible, only correspondence vectors between consecutive frames are transmitted instead of the complete pixel information. The MPEG standard does not regulate how these correspondences have to be computed, but the state-of-the-art technique is correlation.

As correlation is computationally expensive, specialized processors, so called *MEP*s (Motion Estimation Processors) have evolved from this area [6, 8, 9]. Since MPEG-1 and MPEG-2 work block-oriented the basic operation of those MEPs is also referred to as block-matching. A 16x16 pixel reference block (RB) is correlated with a search window (SW) which is typically of the size 32x32. For all possible displacements Δx and Δy (for the mentioned SW size ranging from -8 to $+7$ each) a correlation like value called SAD (Sum of Absolute Differences) is calculated:

$$SAD(\Delta x, \Delta y) = \sum_{y=0}^{15} \sum_{x=0}^{15} |SW(x+\Delta x, y+\Delta y) - RB(x,y)|$$

The result of this operation is a correlation matrix indexed by Δx and Δy. The vector $(\Delta x\ \Delta y)$ referring to the minimum SAD discriminates the block in SW which is most similar to RB and thus the found correspondence.

The idea to use one of those extremely optimized MEPs for the generation of optic flow is not new. Especially Inoue et al describe the integration of the MEP from SGS Thomson [8] into their image processing transputer network [5]. Resulting from their work, a commercial version is available from Fujitsu which is based on the same chip and is meanwhile used in various research projects [2, 4].

A problem which several researchers report is that the optic flow generated by such a correlation processor can become very noisy. This is the case when the image structure in some of the reference blocks is ambiguous or completely missing. Then the detection of a significant minimum in the correlation matrix fails. Unfortunately, this is the case in most indoor environments like e.g. office buildings.

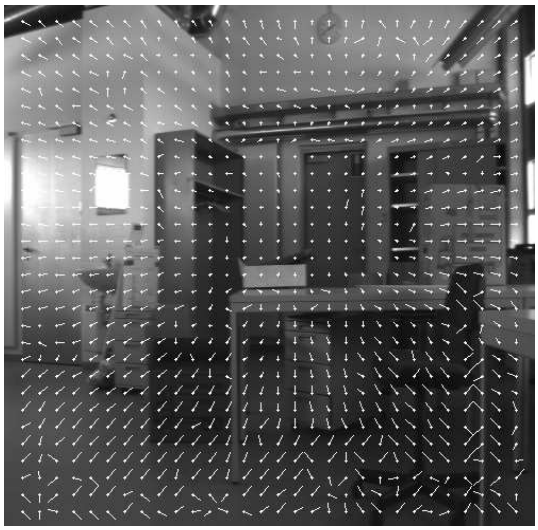

Figure 1: Noisy optic flow, generated by a MEP

Fig. 1 illustrates the problem. The flow was generated by a linear forward movement of the camera, so all vectors should meet in a single point, the FOE (Focus Of Expansion). Due to the local lacks of structure, this can only be observed in a few areas.

To evaluate this flow field, e.g. for the reconstruction of the three dimensional motion, extra knowledge would be necessary.

Approach

We propose a confidence criterion which can be used to sift out the noisy flow vectors. The criterion is simple enough to be calculated in parallel to the MEP operation by few additional circuitry.

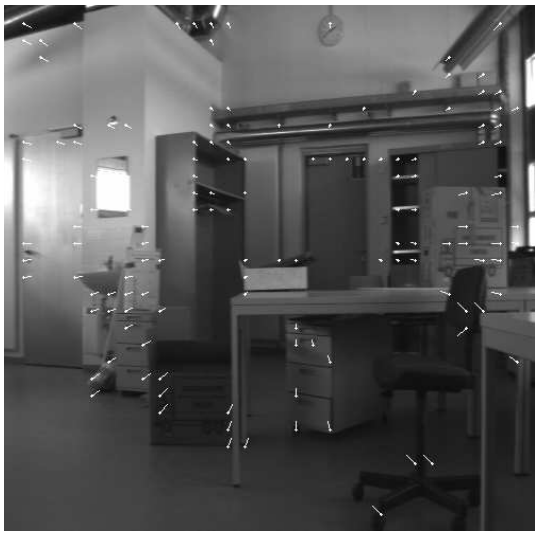

Figure 2: Sifted flow

Fig. 2 shows the resulting flow. The remaining vector field is sparse, but the noise is significantly reduced.

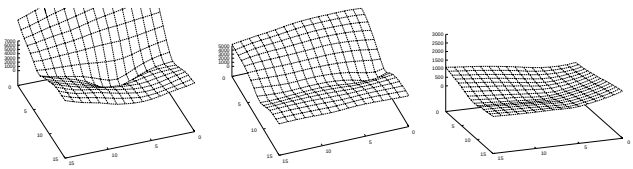

Figure 3: Cases of correlation matrices: a) significant b) ambiguous c) indifferent

When searching for an adequate confidence value for each vector, several ideas come to mind. Obviously the quality of the minimum detection is correlated with the structure in the reference block. A measure for this structure could be calculated by summarizing the differences between adjacent pixels. This computation can be implemented very efficiently by ADSPs or special convolver chips, e.g. [12]. But in the experiments this kind of criteria did not work too well due to their local nature and sensibility to the camera noise.

The confidence value we have finally chosen derives from evaluating the correlation matrix itself. In principle, the three cases depicted in fig. 3 have to be taken into consideration.

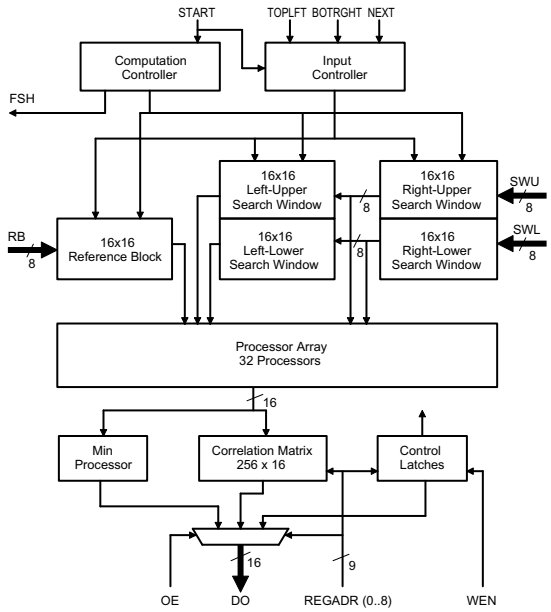

Figure 4: The MEP LSI74720

We calculate the difference d between the SAD of the best match and the SAD of the second best match. In case a) the value of d describes the steepness of the minimum. High values stand for significant matches. In case b) the position of the best match is ambiguous. If both minima have similar SADs, d will be small. In case c) the reference block is missing any structure at all. The SAD for all possible correspondence vectors is similar. Also in this case d is small.

More sophisticated evaluation of the matrix is possible (see e.g. [10] for an overview), but this simple approach showed very good results in our first simulations and is easy to compute if the complete correlation matrix is accessible. These arguments led to the hardware design presented in the next chapter.

System structure

Fig. 4 shows the MEP LSI74720. For each block-match the RB and the SW have to be transferred into the internal buffers of the MEP by external circuitry.

As for the nominal application matching has to be performed for adjacent RBs and symmetric SWs, the search windows overlap by 16 pixels. For this reason, the SW buffer is pipelined and only the right half has to be loaded before the next operation. At the beginning of a new row, two load operations are necessary which roughly doubles the time needed for the first match. The 256 SADs are stored in an additional correlation buffer; the minimum is calculated automatically and written out along with the corresponding $(\Delta x\ \Delta y)$ at the end of the operation. All buffers are double-buffered which allows simultaneous calculation and loading of the next data.

The external calculation of the proposed confidence value without any performance loss is permitted by the LSI74720 because also the complete correlation matrix can be read in parallel to the nominal operation. The second smallest value is then determined by some latches and comparators.

A prototypic system has been realized on an ISA-bus PC-card (see fig. 5). Three frame memories allow comparison of two images and simultaneous acquisition of the next image into the third memory. To manage the data-flows a flexible control logic has been implemented by a set of FPGAs. It contains the address generation for the memories, the calculation of the second smallest SAD and a central crossbar which permits random connection of memories, MEP, and a digital camera.

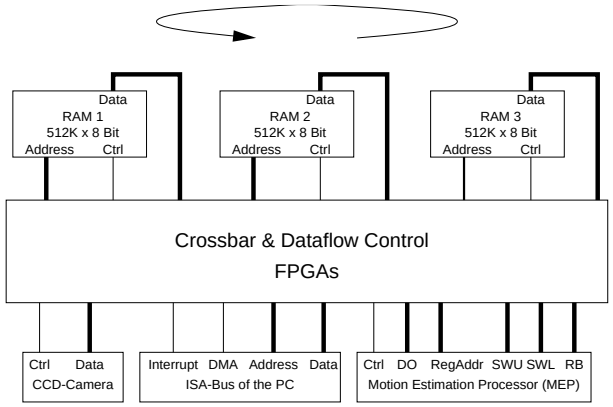

Figure 5: Structure of the developed system

The random configuration of the crossbar allows the usage of the three memories as a ring-buffer, as well as other strategies. For example a reference image can be kept in one of the memories and permanently compared to the next frame. This technique could be used for background-foreground separation or for dynamic control of the comparison rate, allowing also the detection of very slow movements.

For special applications like the tracking of multiple objects or big displacements, the MPEG-like block-raster and fixed search window positions are not adequate. To achieve the highest possible flexibility, the coordinates of RBs and SWs can be randomly set by the software running on the PC for each single matching operation. A list of so called *MEP commands*, containing the desired coordinates for each match, is stored in the PC memory. To take optimal advantage of the mentioned MEP internal pipeline, each com-

mand can also contain a row counter, designating the number of adjacent matches. This command list is read via DMA by the control logic, which then feeds the addressed pixel-blocks into the MEP. The results of each calculation, namely displacement and the SADs of the best and the second best match are written back into the PC memory via DMA again. This saves most of the CPUs processing power for application software which evaluates the generated flow fields. The sifting according to the confidence value d is done by the application software, thereby allowing more sophisticated strategies like e.g. sorting by confidence.

Depending on how often the MEP pipeline has to be interrupted, up to 525 vectors can be calculated per frame (PAL, 25 Hz).

A PC-interrupt is generated for every new frame; the corresponding interrupt handler has to write the next crossbar-configuration and pointers to command and result list into the registers of the control logic. This permits double buffering of the two lists.

Each frame memory can also be accessed directly by the PC. Thus the card could be used as a frame grabber, but the main intention of access by the PC is to load reference images into the frame memories. This allows comparison of current frames with databases, to e.g. recognize stored patterns or objects.

Experiments

Figure 6: Experimental setup with manipulator and poster

Several experiments have been carried out to test the performance of the proposed system. At first, various confidence values were evaluated. An adequate confidence criterion should be able to discriminate the vectors of the optic flow according to their coincidence with the theoretically expected flow. This ability has been statistically tested for the considered criteria. To calculate a reference flow according to equation 1 the six motion parameters and the z-coordinate have to be known for each vector.

In a first series of experiments this was achieved by the setup shown in fig. 6. The camera was mounted on the wrist flange of a standard 6 DOF manipulator, thus allowing exactly known camera movements. The "scene" consisted of a strictly planar poster.

Each optic flow vector was tested against the calculated reference vector. If both were equal (the pixel quantization allowed to test for exact coincidence) the vector was called a *hit*, in the other case a *miss*. A suitable confidence value must be high for hits and low for misses.

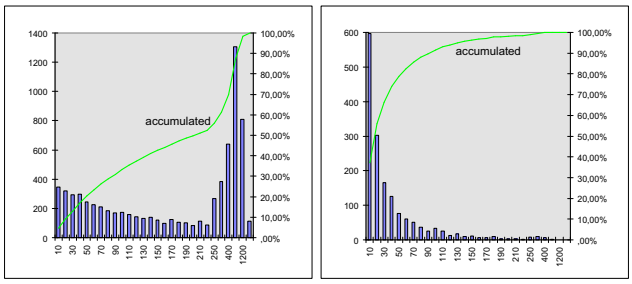

Figure 7: Proposed confidence value: a) *hits*, b) *misses*

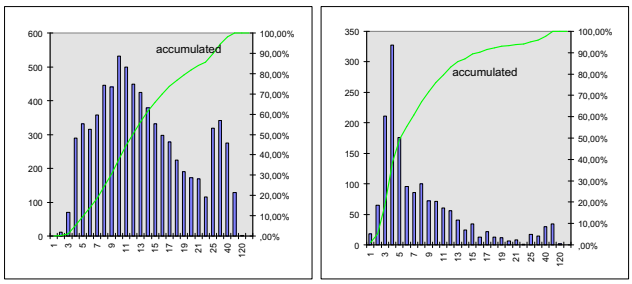

Figure 8: Another considered confidence value: a) *hits*, b) *misses*

The histograms in fig. 7 a) and b) show the results for the proposed confidence value, the abscissa corresponding to the difference d between best and second best match. As demanded, a lot of hits but only very few misses have high values. A criterion can thus be defined by thresholding this value. A threshold of e. g. 100 would sift out 95% of the misses, i.e. the noisy part of the flow, while still 65% of the hits would remain.

A second example from the considered criteria is shown in fig. 8 a) and b). In principle, the behavior is similar. The peak of the hits is shifted toward higher values relative to the peak of the misses. But

Figure 9: Examples for sifted flows in real environments

its not possible to find an appropriate threshold which significantly separates the hits from the misses.

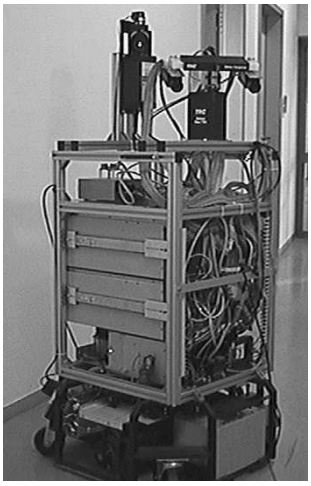

Figure 10: *MARVIN*

As this merely statistic evaluation depends on the kind of the scene, a second series of experiments was carried out in a real world environment. Flows were generated for about 50 random views throughout the office building containing our lab (see fig. 9 for some examples). The setup consisted of a TRC Labmate chassis, a TRC Zebra pan-tilt-head equipped with digital cameras, and several PCs, forming together our mobile "robot" *MARVIN* (Mobile Autonomous Robot with VIsion-based Navigation, see fig. 10).

In this setup unfortunately the z-coordinates are not known. In order to still be able to calculate a reference vector field, only rotations of he camera along an axis through the focal center were performed. This was achieved by changing the vergence angle of the Zebra pan-tilt-head. The results are documented by the histograms in fig. 11.

In this histograms an increased amount of hits has also a low confidence value and thus is sifted out. The reason for this are the areas with little structure which sometimes lead to a hit, but at general have no significant minimum in the correlation matrix. But from the over 500 calculated vectors per frame enough hits remain for most applications. The histogram for the

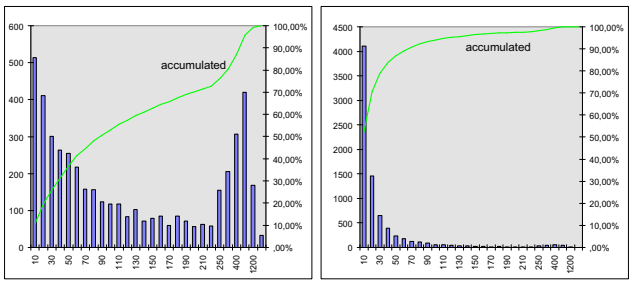

Figure 11: Office environment: a) *hits*, b) *misses*

misses still guarantees the elimination of noisy vectors.

To show the improvements in applications of the optic flow a simple flow evaluation has also been performed. In this experiment, MARVIN starts a turn with a small radius, inducing virtually pure lateral optic flow on the projection plane. The diagrams show the mean lateral flow for the sifted and the complete field.

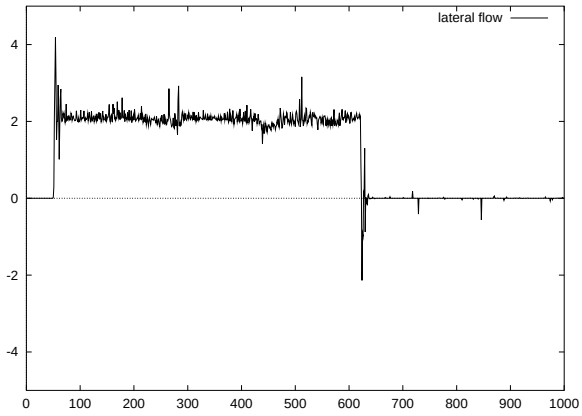

Figure 12: Sifted flow

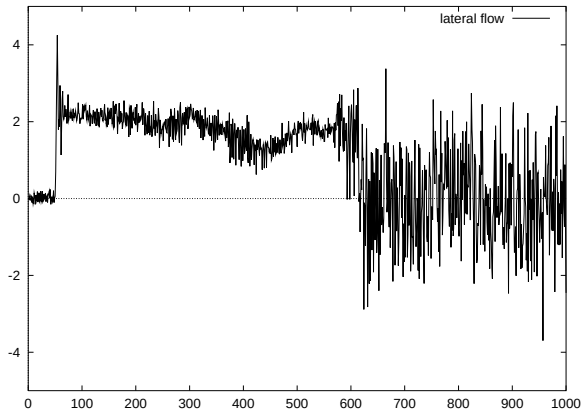

Figure 13: Complete flow

The mean value of the sifted flow according to fig. 12 corresponds well to the actual robot movement. After standing still, the robot accelerates to a constant turning speed, leading to a short vibration of the camera at the beginning. During the turn, a nearly constant flow of two pixel length is measured. After about 22 seconds (550 frames) the robot stops, again resulting in a short vibration, and remains standing still.

The mean value of the complete flow resembles this behavior for the first 10 seconds. During this part of the movement a very structured wallpaper dominates the images. After frame 250 a standard indoor scene like fig. 1 comes into view. From this point on, the noise increases until it is impossible to reconstruct the motion anymore.

Conclusion and further work

We have presented an image processing system for correlation-based real-time optic flow calculation, which overcomes the noise problem by introducing a confidence criterion for each vector and sifting the vector field. As the experiments show, this does lead to a dramatic improvement for the resulting field which on the other hand gets sparse.

Further work will be concerned with adapting standard optic flow applications and evaluation algorithms to these sparse vector fields.

References

[1] P. Anandan. A Unified Perspective on Computational Techniques for the Measurement of Visual Motion. In *International Conference on Computer Vision*, pages 219–230. IEEE, June 1987.

[2] G. Cheng and A. Zelinsky. Real-Time Visual Behaviours for Navigating a Mobile Robot. In *International Conference on Intelligent Robots and Systems*, pages 973–980. IEEE, November 1996.

[3] B. K. P. Horn and B. G. Schunk. Determining Optical Flow. *Artificial Intelligence*, 17:185–203, 1981.

[4] M. Inaba, K. Nagasaka, F. Kanehiro, S. Kagami, and H. Inoue. Real-Time Vision-Based Control of Swing Motion by Human-form Robot Using the Remote-Brained Approach. In *International Conference on Intelligent Robots and Systems*, pages 15–22. IEEE, November 1996.

[5] H. Inoue, T. Tachikawa, and M. Inaba. Robot Vision System with a Correlation Chip for Real-Time Tracking, Optical Flow and Depth Map Generation. In *International Conference on Robotics and Automation*, pages 1621–1626. IEEE, 1992.

[6] LSI Logic. *Image Compression Databook*. LSI Logic Corporation, 1993.

[7] L. Matthies, R. Szelinski, and T. Kanade. Kalman Filter-based Algorithms for Estimating Depth from Image Sequences. *Int. J. Computer Vision*, pages 2989–2994, 1989.

[8] SGS-THOMSON Microelectronics. *Image Processing Data Book*, chapter STI3320 Motion Estimation Processor. SGS-THOMSON Microelectronics, 1990.

[9] Array Microsystems. a77300: Motion Estimation Coprocessor, April 93.

[10] T. Mori, M. Inaba, and H. Inoue. Visual Tracking Based on Cooperation of Multiple Attention Regions. In *International Conference on Robotics and Automation*, pages 2921–2928. IEEE, 1996.

[11] M. Otte and H.-H. Nagel. Optical Flow Estimation: Advances and Comparisions. In Jan-Olof Eklundh, editor, *Computer Vision - ECCV 94*, volume 800 of *Lecture Note in Computer Science*, pages 51–60, Stockholm, Sweden, May 1994. Springer Verlag.

[12] Harris Semiconductor. *Digital Signal Processing Databook*. Harris Corporation, 1993.