

A ROBUST MOTION DETECTION AND ESTIMATION FILTER FOR VIDEO SIGNALS

Markus Latzel, John K. Tsotsos

Centre for Vision Research
York University, Toronto
{markus, tsotsos}@cs.yorku.ca

ABSTRACT

The problem of detecting areas of motion in video sequences and estimating parameters such as speed, direction and dynamics is addressed in many applications of image processing such as video surveillance, object tracking, image stream compression or autonomous navigation systems. Real world computer vision highly depends on reliable, robust systems for recognition of motion cues to make accurate high-level decisions about its surroundings. In this paper, we present a simple, yet high performance low-level filter for motion detection and estimation in digitized video signals. The algorithm is based on constant characteristics of a common, 2-frame interlaced video signal, yet its applicability to generically acquired image sequences will be shown as well. In general, our approach presents a computationally low-cost solution to motion estimation application and compares very well to existing approaches due to its robustness towards environmental changes. A simple application of motion parameter estimation based on a pedestrian surveillance application is illustrated.

1. INTRODUCTION

Motion detection and estimation in image sequences has multiple applications ranging from image stream data compression to artificial intelligence or automatic surveillance problems and is considered a vital component of any of these systems. Accordingly, a large amount of work has been done to identify moving objects within a video sequence and estimate their motion parameters. In the following, yet a different approach for this task is presented, for the reason of its astonishing simplicity and robust performance, as shown in experimental applications.

Motion in a sequence of images is defined as a situation where part of the scene is moving in front of a non-uniform background. This moving part may not be connected, as in the case of multiple moving objects, each drifting in different directions. Furthermore, additional to a simple transla-

tion of static image data, the objects may be rotating, deforming (i.e. experience local motion within themselves) or temporarily occlude each other. On top of this, the capturing device may be submitted to ego-motion, which results in a moving background altogether.

A motion estimation algorithm identifies connected objects within the image that experience motion relative towards the background, and estimate motion parameters, i.e. direction, speed, and occlusion events. In partial solutions, a simple detection of "something in motion" suffices. In many motion alarm systems for surveillance applications, this is the case.

1.1. Background Subtraction

Motion in front of a background should generally be detectable by subtracting a previously stored "background" image, or *reference frame* from the current input, and identify areas of high response as disturbances in the image. Obviously, background subtraction relies on zero ego-motion of the system, and no undesired changes to the observed background. Lighting changes for example, have to be accommodated for by an adaptive background extraction algorithm, which iteratively updates the stored reference frame. In [8], A. Makarov compares background extraction algorithms. Common to his paper and [2] is that the choice of threshold values for discrimination between noise and real changes to the scene is crucial to eliminate false positives during motion detection. In [1], an automatic reference frame update is given to be able to detect motion even after sudden lighting changes occur in an image sequence.

1.2. Interlaced Video Signals

A notion generally left out of consideration is that motion (as temporal changes) is inherently encoded in the image sequence signal acquired from commercial video cameras. The NTSC signal scans 525 lines of image data per frame, with a frame rate of 30fps. To eliminate flicker, two interlaced frames are sent alternatively with a rate of 60fps. Accordingly, alternating lines of the captured image are scanned

Funding of the work was gratefully received from Communication Information Technology Ontario (CITO).

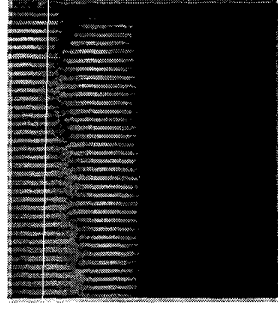


Fig. 1. Motion artifacts in interlaced video signals.

with a delay of $16.7ms$. If we consider an object moving horizontally before a background, then *interlacing artefacts* can be observed along edges of the object, since each even scan line has been captured exactly one frame later than its preceding and following odd line. Figure 1 shows a magnified snapshot of such an artefact. The horizontal extent of these artefacts are directly proportional to the horizontal component of the edge velocity. In terms of the image window, the objects velocity v_o is thus

$$v_o = \frac{l}{16.7ms}$$

in pixel per second, with l being the length of an interlacing artefact.

2. A MOTION BANDPASS FILTER

Detecting the interlace artefacts can be achieved by applying a vertically oriented bandpass filter with the characteristic frequency

$$f_{BP} = \frac{1}{pixel}$$

Since none of the original image should be preserved, the filter thus has a low frequency component of 0, i.e. weight of the filter kernel $k_{BP} = 0$. In order to discriminate weight of scan lines further from the center of the filter kernel, a standard band pass filter was multiplied with an approximated Hamming window function:

$$\begin{aligned} k_{BP} &= k_{stdrd} * k_w \\ &= \begin{bmatrix} -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \\ &\quad * \begin{bmatrix} 1 & 2 & 3 & 4 & 3 & 2 & 1 \end{bmatrix}^T \\ &= \begin{bmatrix} -1 & 2 & -3 & 4 & -3 & 2 & -1 \end{bmatrix} \end{aligned}$$

With the windowing function k_w approximated by a triangle function.

Applied to a raw video image $I(x, y)$, the given filter responds to columns of pixels with alternating intensities. In particular, if the centre of the kernel is located on a white

pixel within a motion artefact, the response will be positive, while on a black pixel, the response is negative. In areas with a frequency other than f_{BP} , the output $I_{BP}(x, y)$ is close to 0.

Theoretically, the filter responds to any high frequency along a vertical axis such as sharp, horizontal edges, for example. Increasing the filter length dy will increase its discrimination towards such 'false responses', and ensure only truly alternating lines produce a response. However, if slanted lines moving with slow velocity produce an artefact length of less than $dy/2$, sensitivity of the filter is reduced. Experimentally, we found that a filter length $dy = 7$ was optimal for the motion detection applications presented in the following.

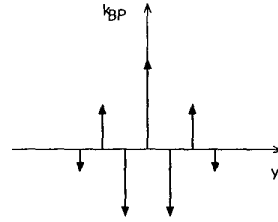


Fig. 2. A possible bandpass filter for k_{BP} .

To decide whether moving edges are present in an image or not, the complete filter takes the absolute value of the output and generates a threshold image thereof:

$$M(x, y) = \begin{cases} 1 & \text{if } |I_{BP}(x, y)| > \text{threshold} \\ 0 & \text{else} \end{cases} \quad (1)$$

In an experimental application, we were able to use the filter for a simple intrusion detection system with this method. An intrusion into the field of view was detected if

$$\sum_{x,y} M(x, y) > k,$$

with k being a sensitivity parameter, depending on dimensions of the acquired image and expected size of a moving object. A larger object will generally produce longer edges and thus increase positives in M .

Since this algorithm does not store any reference data, it does not react with false positives towards gradual changes in the image. In other words, very slow and global changes do not trigger the bandpass filter at all. As a result, lighting changes do not result in any false intrusion alarms.

2.1. Thresholding

Selecting the threshold to attain $M(x, y)$ proved to be quite robust. In fact for both the indoor and outdoor images shown,

the same threshold t was used. However, a simple threshold adjustment scheme was used to automatically increase the threshold on a static image input until the $M(x, y) = 0$ for all x, y . The thus computed threshold was afterwards treated as a system dependent variable and needed no further updating. With k_{BP} as given in figure 2, and an input value range of $[0..255]$ for I , I_{BP} can assume values within $[-8 \cdot 255..8 \cdot 255]$. The histograms in figure 3 show a recording of $|I_{BP}|$ over this period, with the computed threshold marked. Note that for the still reference image, no values above t exist (i.e. $M(x, y) = 0 \forall x, y$). Again, the histogram does not change significantly for extreme changes in lighting conditions (indoor/outdoor).

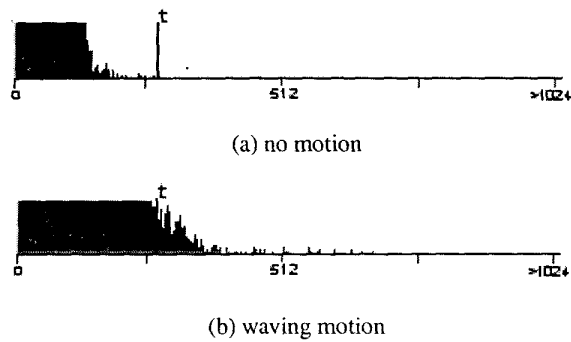


Fig. 3. Value histogram of $|I_{BP}|$. (b) was captured with a waving motion as in figure 4.

3. MOTION PARAMETER ESTIMATION

In order to provide information for higher level computer vision systems, it is necessary to estimate motion parameters such as location, velocity and direction of moving objects from the filter output. Below, an outline of this is achieved and some consideration on issues are given:

3.1. Direction of motion

In order to utilise the polarisation property of the filter, the output signal I_{BP} was further processed to display this information: a three colour image $N(x, y)$ was defined as:

$$N(x, y) = M(x, y) \cdot \text{sign}(I_{BP}(x, y))(-1)^y,$$

with y being the scan line number. Figure 4 shows $N(x, y)$ encoded in three different colours, black for $M(x, y) = 0$.

$N(x, y)$ denotes a gradient function of image values: a transition from lower intensity to a higher one results in a positive value. Thus, the direction of motion can be computed from information derived from a single frame.

3.2. Motion velocity

Motion velocity $v_{o,h}$ along the horizontal trajectory can be estimated from a single image snapshot by measuring the artefact length l , as mentioned earlier. However, a vertical motion will also yield filter response at slanted and horizontal edges of the object. Thus, any object motion yields an edge response of the filter, but the exact component of vertical motion can not be estimated with this algorithm unless a more sophisticated approach is taken, such as operating on a vertically interlaced input image. However, real time performance of the simple horizontal motion estimation suggests that the given approach will assist a higher level decision system to in image processing, identified frames and areas of interest, with a preliminary motion velocity estimation algorithm.

Also, given the high temporal resolution of the motion filter approach (an inter-frame approach may - with appropriate hardware - at best analyse motion in intervals of 33.3ms), *fast motion* is easy to detect and estimate due to the scan line interval of 16.7ms. Thus, for example in a traffic surveillance application, speeding drivers are captured even if they appear only in a single frame of the image sequence. In contrary, very *slow motion* produces a dimmed output of the filter. This drawback can be accommodated by periodically *interlacing* previously stored key frames with the current image. In other words, to virtually triple the speed of passing objects, a composite frame $J(x, y, n)$ consists of interlaced frames $I(n)$, and $I(n - 1)$, with n being the number of the current frame:

$$J(x, y, n) = \begin{cases} I(x, y, n) & \text{if } y \text{ even} \\ I(x, y, n - 1) & \text{else} \end{cases}$$

Applying the bandpass to J amplifies the sensitivity towards slow motion.

4. EXPERIMENTAL RESULTS

As an experimental application, bypassing pedestrians were recorded with the motion filter. Our primary concern was to validate the robustness of our approach by comparing processing of indoor and outdoor motion scenes. As shown in figure 5, the walking person could be identified as an outline response by the filter. By clustering filter output responses, the average response location for a moving object can be computed (shown as hashmarks in figure 5). An even larger response length was recorded for a running motion. Note that high frequency components of the rather noisy background do not lead to false positives in the threshold image $M(x, y)$.

For this experiment, a Pentium III processor was used in combination with a Canon video conference camera and a commercial frame grabber.

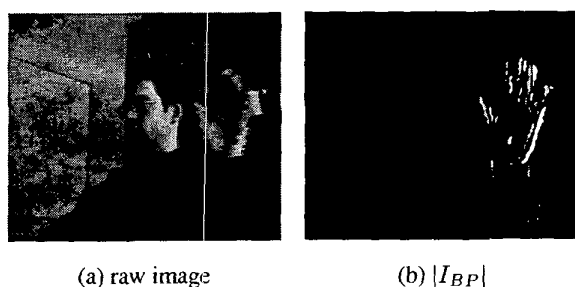


Fig. 4. Raw image data and filtered output of a hand in waving motion. Note the low contrast of the acquired image

5. CONCLUSIONS

In this paper, we showed that a simple approach to motion detection can be derived from the interlaced property of NTSC video signals. The motion filter proposed was tested in several experimental applications and proved to be robust towards environmental changes such as noise and lighting conditions. From the acquired data, it is possible to estimate motion parameters in a high temporal resolution not possible to inter-frame motion detection approaches. Further work will involve a more sophisticated motion tracking system, combining the motion filter with a model-based tracking algorithm to track known objects in real time.

6. REFERENCES

- [1] E. Durucan, F. Ziliani, O.N. Gerek, *Change Detection with Automatic Reference Frame Update and Key Frame Detector*, in Proceedings of the IEEE-Eurasip Workshop on Nonlinear Signal and Image Processing (NSIP'99), vol. 1, pp. 57-60, Antalya, Turkey, June 20-23, 1999
- [2] F. Ziliani, A. Cavallaro, *Image Analysis for Video Surveillance Based on Spatial Regularization of a Statistical Model-Based Change Detection*, ICIAP'99, pp. 1108-1110, Venezia, 1999
- [3] A. Francois, G. G. Medioni, *Adaptive Color Background Modeling for Real-Time Segmentation of Video Streams*, In Proc. Int. Conf. on Imaging Science, Systems, and Technology, pp. 227-232, Las Vegas, NA, June 1999
- [4] D. Toth, T. Aach, V. Metzler, *Illumination-Invariant Change Detection*, Southwest Symposium on Image Analysis and Interpretation April 2-4, 2000, Austin
- [5] B. G. Schunck, *The Image Flow Constraint Equation*, Computer Vision, Graphics, and Image Processing, 35:20-46, 1986



(a) Raw outside image



(b) Filtered image

(c) Running motion

Fig. 5. Raw image data and filtered output of a pedestrian scene. Image (b) and (c) show hash marks at the centre of detected motion, while the hull of clustered response points is shown as an overlay in image (a). Image (c) shows larger response lengths due to 'running' motion.

- [6] B. Jähne, *Digital Image Processing*, Springer, Berlin, 1997
- [7] K.P. Karmann, A. Brandt, R. Gerl, *Moving Object Segmentation Based on Adaptive Reference Images*, Proc. 5th European Signal Processing Conference, pp. 951-954, Barcelona, 1992
- [8] B. Makarov, *Comparison of Background Extraction Based Intrusion Detection Algorithms*, IEEE International Conference on Image Processing (ICIP'96) pp. 521-524, 1996
- [9] A.K. Jain, *Fundamentals of Image Processing*, Prentice-Hall, 1989