SIGNAL PROCESSING IV: Theories and Applications
J.L. Lacoume, A. Chehikian, N. Martin, and J. Malbos (Editors)
Elsevier Science Publishers B.V. (North-Holland)

Stereo by Dynamic Programming boosted using Binary Images

Jean-Pierre AMANN & Heinz HUGLI

Institut de Microtechnique, Université de Neuchâtel
Rue A.-L. Breguet 2, CH-2000 Neuchâtel, SWITZERLAND

This paper addresses the problem of how to reduce computation time in a stereo image matching system used for depth determination in a 3D scene. The following steps are considered. First comes the preprocessing of the image data, edge detection and edge orientation extraction. Second is the correspondance search algorithm, which we propose to solve using a simplified Two-Dimensional Dynamic Programming (2D-DP) approach. In essence, we consider applying DP to binary images in order to simplify and speed up computation.

1. Problem definition

The key problem in stereo is a search problem which finds the correspondance between points of the left and the right images such that, given the camera model, the depth can be computed by triangulation. Using the epipolar camera model, the general problem of finding correspondances within the whole image can be simplified to a scanline-to-scanline problem [1],[2], called "intra-scanline search".

Among the various methods used, the one based on edge detection and matching using Dynamic Programming (DP) is known to show good performance and will be considered here. While these general techniques are well known [1],[2], few proposals have been made to reduce significantly the time of computing.The main bottlenecks in edge-based stereo are i) the edge point extraction and ii) the correspondance search algorithm.
Our basic approach is to apply edge detection to images with the full resolution and to use binary images only while applying DP. The first contributes to the precise location of edge transitions while the second helps to cut down computation costs.

2. Edge point extraction

Considering edge extraction first, we have to solve the problems i) detection of edges and their orientation and ii) edge thinning.

2.1 Gradient and orientation image

Algorithm
Following the Nevatia/Babu approach [3], we use a set of six 3x3 local operators which are sensitive each to one orientation. The choice of a 3x3 operator size instead of the original 5x5 size proposed in [3] is due to its better localization properties but carries the disadvantage that it is more sensitive to noise.
For each pixel p(x,y), the orientation **maxori** for which the absolute value of the gradient **grad** reaches its maximum value as well as the maximum gradient value **maxgrad** have to be determined. For further use, they are memorized in a an "orientation image" and "gradient image" respectively.

$$(1)\ \mathbf{grad(x,y,ori)} = \sum_{l}\sum_{m} \mathbf{p(x+l,y+m)*op(l,m,ori)}$$

$$(2)\ \mathbf{maxgrad(x,y)} = \max_{ori}(|\mathbf{grad\ (x,y,op(ori))}|)$$

$$\Rightarrow \mathbf{maxori(x,y)}$$

where $l, m = -1..1 \quad ori = 1..6$

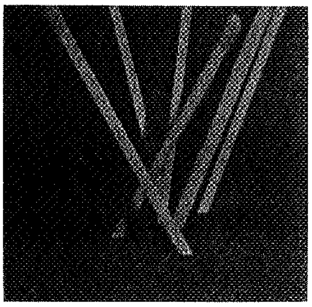
While the "gradient image" is defined over all the image, the "direction image" is defined only where variations in the intensity had been detected. Hence, the complete information is available only for pixels located on or near egdes.

Computation cost
The implementation of this edge point and edge orientation extraction algorithm in software is extremely time consuming. This problem can be resolved using specialized hardware [4] which allows real time computing.

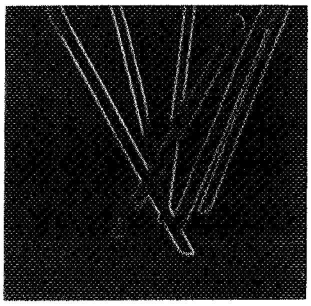
2.2 Edge thinning

Algorithm
As we mentioned above, the gradient image shows non-zero values not only on pixels representing edges but also on pixels nearby. This inhibits a precise localization, a very important point in stereo vision since the precision of depth computing depends on it directely.
In our experiments, we used the edge thinning algorithm described in [3]. A pixel is retained as edge point only if its gradient value is i) bigger than a predefined threshold, ii) its gradient is bigger than the one of its neighbouring pixels on a line perpendicular to the edge orientation detected and iii) if the orientations of these points are consistent.

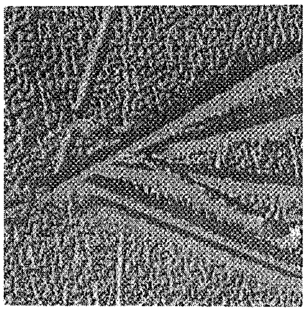
Computation cost
It can be shown that line thinning algorithm can be done in real time too, using the same type of specialized hardware as above.



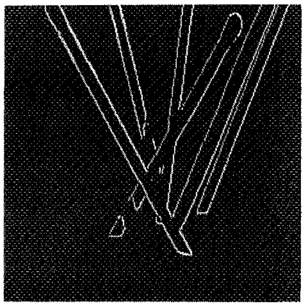
a) original image



b) gradient image



c) direction image



d) thinned-edge image

figure 1: edge extraction and line thinning

Line linking to fill gaps in edges has not been implemented since it can not be realized in a pipelined form.

2.3 Example

Figure 1 shows an example with fig. d) as the final result which is used later for stereo matching purposes. The localization of the edges as well as the continuity of the edges is very good despite of the edge linking which had not been implemented.

2.4 Conclusion on edge point extraction

We have shown that edge detection and edge thinning can be done in real time using known specialized hardware. Therefore, these preprocessing steps will not be the limiting factor in the correspondance problem.

3. Correspondance search by DP

As in the heart of the DP algorithm for the correspondance search there is repeated computation of the partial path cost, i.e. the cost of bringing two segments in correspondance in order to form a partial path, it is of utpost importance to keep simple this computation, if simple processor architecture and fast computation is aimed.

3.1 Estimation of the number of partial pathes

Using typical values [2], we can estimate on the number of partial path computations N_{pc} to perform, considering "intra-scanline" matching (table 1). Taking into account "inter-scanline" dependencies 3D-DP, N_{pc} grows further by a factor $\varepsilon \approx 5$ [2]. This huge number of partial pathes motivated our research for simplier partial-path computation.

(3)		number of partial path computations 2D-DP	$\mathbf{N_{pc} \cong n\, w^2 e^2 d} \approx 2\cdot10^4 .. 2\cdot10^7$
	where	maximum disparity allowed / scanline length	$d = 0.05 .. 0.2$
		mean value of edges found	$e = 5 .. 90$
		DP search window size	$w = 5$
		height and width of an image	$n = 512$

table 1: number of partial path computations

(4)	$C_p = C_l * C_i$	where	C_i = f (difference of intensities)
			C_l = f (length of the segments)
(5)	$C_l = \sqrt{(l^2 + r^2)}$	where	l = length of the left segment
			r = length of the right segment
(6)	$C_i = \sigma^2 = 0.5\,({}^1/_l\, \Sigma\, (a_i - m)^2 + {}^1/_r\, \Sigma\, (b_j - m)^2)$		
(7)	$m = 0.5\,({}^1/_l\, \Sigma\, a_i + {}^1/_r\, \Sigma\, b_j)$		
		where	a_i = pixels of the left scanline segment (i = 1..l)
			b_j = pixels of the right scanline segment (j =1..r)

table 2: traditional partial path computation

3.2 Reducing computation cost

A traditional method for partial cost computation can be given by the formulas of table 2 [2].
Our proposal is to use binary images for the DP correspondance search. So the traditional formula (6) [2] can be replaced by a new metric presented in formula (8) which allows us to reduce the computation time to a minimum.

$$(8) \quad C_i = \sigma^2 \cong ({}^1/_l\, \Sigma\, a_i - {}^1/_r\, \Sigma\, b_j)^2$$

We will show in the result section that formula (8) behaves similarly to the original formula (6) while reducing drastically both the complexity and the time of computation of the PD partial-path cost.

3.3 Maintaining good performance

While the traditional approach [2] extends the 2D-DP to a more complicated and time consuming 3D-DP, taking into account inter-scanline dependencies, we propose to keep the simplier 2D-DP approach. Increased performance can be obtained from dissimilarity measure on edge directions, adding a C_a factor to formula (4) as given in table 3.
As we mentioned above, edge directions are can be extracted easily using specialized hardware. While we used 3x3 operators for edge localization, edge orientation and edge thinning since we were interested in good localization, we use here 5x5 operators in order to limit noise influence.
Second, we take care of the sign of the gradient in order to obtain a signed orientation of the edges, called direction. Since there is only a limited number of directions, the comparison can be done very rapidly using look-up tables. Hence we obtain a significantly better performance at a very small additional computation cost.

3.4 Comparison of computation time

Using the assumptions of table 1 again, we obtain the number of significant operations (additions and multiplications) for the match of a couple of stereo images as it is shown in table 4.

While N_{pc} is the number of partial pathes as discussed before, N_{prep} represents the preparational work which leads to a more efficient PD algorithm. In fact, some informations on segments to be compared (mean intensity of the space between two edges, length) can be extracted in advance and can be reused in several partial path computations.

3.5 Discussion

While in the traditional formula (6), data preparation can be used for the m terms only, the sums of squares have to be computed for each new partial path. Our formula (8) in turn uses mean values which can perfectly be computed in advance.
Second, following our proposal, binary images can be used. This simplification has not been considered in table 4. So, the computational cost will be cut down further.
Third, the extension of the matching process, using edge direction attributes, is very cheap in terms of computation cost. While taking advantage of this supplementary feature, its cost can be neglected when using look-up-tables.

(9)	$C_a = \Delta_{dir}l + \Delta_{dir}r$	where	$\Delta_{dir}l$ = difference of edge directions (left end)
			$\Delta_{dir}r$ = difference of edge directions (right end)
(10)	$C_p = C_l * [(1-\alpha) * C_i + \alpha * C_a]$		α = weighting factor

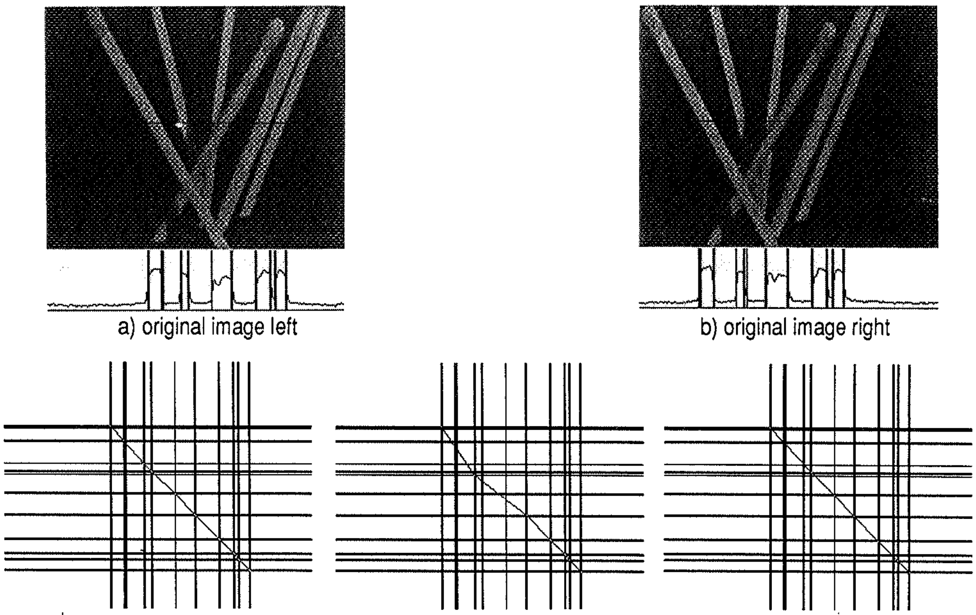
table 3: new metrics taking into account inter-scanline dependencies

Ohta/Kanade 2D DP	$O \cong N_{pc} * (10+4\omega+6\omega\mu) + N_{prep}$	$\approx 4.0*10^7 .. 3.4*10^9$
Ohta/Kanade 3D DP	$O \cong N_{pc}*((10+4\omega+6\omega\mu) + N_{prep}) *\varepsilon$	$\approx 1.9*10^8 .. 1.7*10^{10}$
proposal 1: DP binary	$O \cong N_{pc} *(11+4\omega) + N_{prep}$	$\approx 9.6*10^5 .. 5.6*10^8$
proposal 2: DP binary+directions	$O \cong N_{pc} *(12+4\omega) + N_{prep}$	$\approx 9.8*10^5 .. 5.8*10^8$

where $N_{pc} = nw^2e^2d$, $N_{prep} = 2n(n+e)$, $\mu = {}^n/_e$, $\omega = w-1$, $\varepsilon = 5$ (factor for 3D DP [2])

table 4: comparison of computational cost

4. Results



a) original image left

b) original image right

c) PD grid Ohta 2D

d) PD grid proposal 1

e) PD grid proposal 2

figure 2: edge matching

The figures above show some results obtained by the different DP algorithms described. In the upper half, the couple of stereo images of the original scene is shown, together with an intensity profil for the scanline used in this comparison. Edges are marked with vertical lines.

Figure 2c shows the DP grid, composed of the edges considered, as well as the optimal path found for the Ohta/Kanade algorithm 2D-DP. The "intra-scanline" match is correct.

Figure 2d shows the results for proposal 1, a modified 2D-DP using binary images for the C_i computation. This simplified algorithm works well, except for one couple of edges (upper left of PD grid. This non-ideality disappears when the additional edge direction attribute is considered too (figure 2e).

5. Conclusions

We have proposed solutions to reduce the time of computation in a edge-based stereo ranging system by several means. Using for the preprocessing of the image data, edge detection and edge orientation high-speed hardware, we propose the following two measures: i) for the correspondance problem the use of binary images which allows us to speed up the computation of the DP-partial-path cost and ii), in order to improve the matching results, to include a component measuring edge directions. The experiments provide qualitative informations about the performance of our approach.

References

[1] H.H.Baker, Th.O.Binford
"Depth from Edge and Intensity Based Stereo"
Proc. 7th Int. Joint Conf. Artif. Intell., Aug 1981

[2] Y.Ohta, T.Kanade
"Stereo by Intra-and Inter-Scanline Search Using Dynamic Programming"
PAMI, Vol 7, No 2, March 1985

[3] R.Nevatia, K.Babu
"Linear Feature Extraction and Description"
Computer Graphics and Image Processing, Vol 13, 1980

[4] T.Fukushima and al
"ISP: A dedicated LSI for gray image local operations"
Proc 7th Int Conf on Pattern Recognition, 1984