PLANAR MOTION ESTIMATION WITH A FAST CONVERGING ALGORITHM*

Norbert Diehl, Hans Burkhardt

Arbeitsbereich Technische Informatik I, Technische Universität Hamburg-Harburg
Postfach 90 14 03, D-2100 Hamburg 90, West-Germany

ABSTRACT

An iterative algorithm to estimate simultaneously rotation and translation parameters of moving planar rigid bodies in grey-scale image sequences is developed and discussed. The advantage of the algorithm is the combination of very good performance features as large stability region and high image-bandwidth-adaptive convergence rate of at least second order near the optimum with a minimum of numeric expense within each iteration step. Furthermore the algorithm does not require knowledge of any point correspondence but uses the normal grey-scale images within an area of interest.

Two modifications of the algorithm using clipped signals are given and results testing the algorithm and its modifications using real image data are presented.

1. INTRODUCTION

This paper is concerned with the joint estimation of rotation and translation parameters of moving rigid planar objects in image sequences. The problem of motion estimation has become of great interest in several areas of image processing as motion compensated image coding, remote sensing by satellites, robotics and biology. Details are given in the surveys of Nagel [1,2] and the book of Huang [3]. It may be observed that a lot of different algorithms to estimate pure translatoric displacement have been published and discussed in detail. These algorithms may not straightforwardly be extended to the problem of a joint estimation of rotation and translation because rotation and translation are not independet and therefore do not commute. Only a few papers [4,5,6,7,8] discuss this problem in detail or give an efficient parameter estimation algorithm based on ordinary grey-scale images without using special features as corresponding points.

This paper describes a new fast converging algorithm to estimate rotation and translation simultaneously in a few iterative steps. The advantages of the algorithm are manifold. First it has a high, image-bandwidth-adaptive convergence rate near the optimum which is suitable for tracking problems as the motion leads only into a near neighbourhood. The convergence rate of the estimate is at least of second order [9]. Second the method has a large region of stability which is very useful when only little a priori knowledge of the motion vector is available. Furthermore the algorithm is specially designed to keep the numeric complexity within each iteration step as low as possible. Therefore the algorithm seems to be suitable also for near real-time applications.

* This work was supported by the Deutsche Forschungsgemeinschaft (DFG)

2. THE ALGORITHM

The model adaptive parameter estimation algorithm is based on a simplified signal model. The two grey-scale images $I_1(\mathbf{x})$ and $I_2(\mathbf{x})$ describe the motion of a rigid planar object $S(\mathbf{x})$ in front of a uniform background where no occlusion effects occur:

$$\begin{aligned} I_1(\mathbf{x}) &= S(\mathbf{x}) = S(x,y) \\ I_2(\mathbf{x}) &= S(x\cos\phi - y\sin\phi - d_1, x\sin\phi + y\cos\phi - d_2). \end{aligned} \tag{1}$$

$I_2(\mathbf{x})$ results from $I_1(\mathbf{x})$ by rotation ϕ and translation d_1, d_2. To get an appropriate estimate $\hat{\mathbf{T}} = (\hat{\phi}, \hat{d}_1, \hat{d}_2)^T$ of the parameters $\mathbf{T} = (\phi, d_1, d_2)^T$ the motion is modelled by

$$I_m(\mathbf{x}, \hat{\mathbf{T}}) = S(x\cos\hat{\phi} - y\sin\hat{\phi} - \hat{d}_1, x\sin\hat{\phi} + y\cos\hat{\phi} - \hat{d}_2), \tag{2}$$

and the parameter vector $\hat{\mathbf{T}}$ is changed until the image difference $e(\hat{\mathbf{T}})$ between $I_m(\mathbf{x}, \hat{\mathbf{T}})$ and $I_2(\mathbf{x})$ and thus an error criterion $J\{e(\hat{\mathbf{T}})\}$ reaches a minimum. The error criterion is chosen to be the expectation value of the squared image difference e

$$J\{e(\hat{\mathbf{T}})\} = E\{e^2\} = E\{(I_m(\mathbf{x}, \hat{\mathbf{T}}) - I_2(\mathbf{x}))^2\} \tag{3}$$

which is assumed to be twice differentiable. For ill conditioned images where the error criterion has no dominant parabolic shape near the optimum appropriate filtering may be used.

For stationary stochastic signals $J\{e(\hat{\mathbf{T}})\}$ equals twice the negative cross-correlation function of $I_m(\mathbf{x}, \hat{\mathbf{T}})$ and $I_2(\mathbf{x})$ plus an additive constant. A direct solution for the minimization of the error criterion $J\{e(\hat{\mathbf{T}})\}$ is given by a global search for the optimal parameter vector $\hat{\mathbf{T}}=\hat{\mathbf{T}}^*$ resulting in the calculation of the three-dimensional cross-correlation function $R(\phi, d_1, d_2)$ in conjunction with a tremendous numerical complexity.

We use instead an iterative minimizing strategy changing the parameter vector $\hat{\mathbf{T}}$ well-directed until $J\{e(\hat{\mathbf{T}})\}$ is a minimum. This algorithm is an extension of a one-dimensional modified Newton-Raphson-algorithm [10] to estimate several parameters based on two-dimensional signals. The main structure of the algorithm is given by the iteration

$$\hat{\mathbf{T}}^{K+1} = \hat{\mathbf{T}}^K - \mathbf{H}^{-1}(\hat{\mathbf{T}} = \hat{\mathbf{T}}^* = \mathbf{T}) \cdot \mathbf{g}(\hat{\mathbf{T}}^K), \tag{4}$$

with the Hessian $\mathbf{H}$ as the second derivative and the gradient vector $\mathbf{g}$ as the first derivative of the error criterion $J\{e(\hat{\mathbf{T}})\}$ with respect to the parameter vector $\hat{\mathbf{T}}$. Note that the Hessian is always used at the optimum $\hat{\mathbf{T}} = \hat{\mathbf{T}}^* = \mathbf{T}$ and not at the actual iteration point $\hat{\mathbf{T}}^K$ as in the original Newton-Raphson-algorithm. Based on that the Hessian has to be computed only once which can be done before starting the iteration. This leads to several advantages which will be discussed in detail below.

Because of the fact that rotation and translation are not independent and therefore do not commute eq.(4) has to be slightly modified to preserve the advantages of the algorithm. Therefore we introduce a running coordinate system $\{\mathbf{x}^K\} = \{x^K, y^K\}$ and $\tilde{\mathbf{T}}^{K+1} = (\tilde{\phi}^{K+1}, \tilde{d}_1^{K+1}, \tilde{d}_2^{K+1})^T$ as an additional motion vector describing the estimate of the K+1-th iteration on the basis of the coordinate system $\{\mathbf{x}^K\}$ of the K-th iteration.

Thus the relation between the model image $I_m(\mathbf{x})$ of the K-th iteration step and the model image $I_m(\mathbf{x})$ of the K+1-th iteration step is introduced as following

$$\begin{aligned} I_m(\mathbf{x}, \hat{\mathbf{T}}^{K+1}) &= I_m(\mathbf{x}, \hat{\mathbf{T}}^K, \tilde{\mathbf{T}}^{K+1}) = S(\mathbf{x}^{K+1}) \\ &= S(x^K \cos\tilde{\phi}^{K+1} - y^K \sin\tilde{\phi}^{K+1} - \tilde{d}_1^{K+1}, ...) \\ &= S(x\cos\hat{\phi}^{K+1} - y\sin\hat{\phi}^{K+1} - \hat{d}_1^{K+1}, ...). \end{aligned} \quad (5)$$

Thus the innovation $\tilde{\mathbf{T}}^{K+1}$ is given in the transformed coordinates $\{\mathbf{x}^K\}$ and only indirectly in the coordinates $\{\mathbf{x}\}$. Further details are given in [9]. Now the correct two-dimensional extension of the one-dimensional modified Newton-Raphson-algorithm is given by

$$\hat{\mathbf{T}}^{K+1} = \mathbf{A}^{K+1}\hat{\mathbf{T}}^K + \tilde{\mathbf{T}}^{K+1} = \mathbf{A}^{K+1}\hat{\mathbf{T}}^K - \tilde{\mathbf{H}}^{-1} \cdot \tilde{\mathbf{g}}(\hat{\mathbf{T}}^K). \quad (6)$$

The slightly modified Hessian $\tilde{\mathbf{H}}$ at the optimum and the slightly modified gradient vector $\tilde{\mathbf{g}}(\hat{\mathbf{T}}^K)$ can be expressed by the derivatives of the two images $I_1(\mathbf{x})$ and $I_2(\mathbf{x})$ with respect to the coordinates $\{\mathbf{x}\}$. Finally with the abbreviation $\partial/\partial\phi = y\cdot\partial/\partial x - x\cdot\partial/\partial y$ and the operator $\partial = (\partial_1, \partial_2, \partial_3)^T = (\partial/\partial\phi, \partial/\partial x, \partial/\partial y)^T$ we get

$$\tilde{H}_{ij} = 2\, E\{\partial_i I_1(\mathbf{x}) \partial_j I_1(\mathbf{x})\} \qquad i,j = 1,2,3; \quad (7)$$

$$\tilde{g}_i(\hat{\mathbf{T}}^K) = -2 \sum_{j=1}^{3} B_{ij}^K\, E\{(I_m(\mathbf{x}, \hat{\mathbf{T}}^K) - I_2(\mathbf{x}))\partial_j I_2(\mathbf{x})\}. \quad (8)$$

The weighting matrices

$$\mathbf{A}^K = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\tilde{\phi}^K & -\sin\tilde{\phi}^K \\ 0 & \sin\tilde{\phi}^K & \cos\tilde{\phi}^K \end{pmatrix} \quad (9)$$

and

$$\mathbf{B}^K = \begin{pmatrix} 1 & \hat{d}_1^K \sin\hat{\phi}^K - \hat{d}_2^K \cos\hat{\phi}^K & \hat{d}_1^K \cos\hat{\phi}^K + \hat{d}_2^K \sin\hat{\phi}^K \\ 0 & \cos\hat{\phi}^K & -\sin\hat{\phi}^K \\ 0 & \sin\hat{\phi}^K & \cos\hat{\phi}^K \end{pmatrix} \quad (10)$$

describe the interrelation of rotation and translation.

A special property of the algorithm of eq.(6) is the fact that the Hessian is always taken at the optimum and not at the actual iteration point. This has two impacts. First the performance is improved, i.e. the algorithm combines the advantages of the gradient-technique (large region of stability) with those of the normal Newton-Raphson-technique (good signal adaptive convergence property near the optimum which is independent of the image bandwidth). The convergence rate of an adequate error norm $\varepsilon = \| \hat{\mathbf{T}}^K - \mathbf{T} \|$ is at least of second order [9]. Second the algorithmic expense keeps quite small as will be discussed in the following.

As mentioned before the Hessian taken at the optimum has to be calculated only once before starting the iteration, because according to the special model structure the Hessian at the optimum (eq.(7)) can be expressed only by derivatives of the image $I_1(\mathbf{x})$ with respect to the coordinates $\{\mathbf{x}\}$. These derivatives are independent of the actual parameter vector $\hat{\mathbf{T}}$ and can therefore be computed off-line. The only high-dimensional data dependent expression to be updated within the iteration is the gradient vector $\tilde{\mathbf{g}}(\hat{\mathbf{T}}^K)$ given in eq.(8). Therefore four main steps are necessary. First the new model image $I_m(\mathbf{x}, \hat{\mathbf{T}}^K) = S(\mathbf{x}^K)$ has to be generated either from $I_1(\mathbf{x})$ by rotating and translating with the parameter vector $\hat{\mathbf{T}}^K$ or from the preceeding model image $I_m(\mathbf{x}, \hat{\mathbf{T}}^{K-1})$ by rotating and translating with the parameter vector $\tilde{\mathbf{T}}^K$. In a second step $I_2(\mathbf{x})$ has to be subtracted from $I_m(\mathbf{x}, \hat{\mathbf{T}}^K)$; i.e. the image difference has to be built. Third this image difference has to be multiplied by the partial derivatives of $I_2(\mathbf{x})$ according to eq.(8). Note that these derivatives - like the Hessian - can be calculated at the beginning of the iteration because they are also independent of the actual parameter vector $\hat{\mathbf{T}}$. In a fourth step the expectation values have to be computed, which can be done either directly by summing up the calculated values over a region of interest or indirectly by using the histogram of the calculated values. The rotation and translation of the images may be realized either internally by a digital signal processor or externally by moving the camera. Under the assumption that a dedicated hardware is available especially for the rotation of image frames at TV-rate one iteration step could be implemented in 8 TV-cycles, namely one cycle to generate $I_m(\mathbf{x}, \hat{\mathbf{T}}^K)$, one cycle for the subtraction, two cycles for multiplication and expectation operation for each component using the histogram analyzer. Therefore approximately 320 msec are necessary for one iteration step.

3. MODIFICATIONS OF THE ALGORITHM

There are several possibilities to reduce the complexity of the identification algorithm furthermore. It may be observed that in many applications the off-diagonal elements of the Hessian may be negligible. Therefore instead of six elements only the three diagonal elements have to be calculated and the inversion is trivial. Nevertheless because of the possibility to calculate the Hessian before starting the iteration no significant computational savings within the iteration are attained.

Another possibility to simplify the algorithm is to process only the sign of the dc-free images. If only one image is quantized we get relais-correlation, if both signals are quantized we get polarity-correlation. Basis for relais-correlation is the fact that - at least for gaussian signals - the relais-correlation function $R_{sgn(u),v}(\tau)$ is proportional to the normal correlation function $R_{u,v}(\tau)$ [11]

$$R_{sgn(u),v}(\tau) = E\{sgn(u(t+\tau))v(t)\} = \sqrt{\frac{2}{\pi}}\frac{1}{\sigma_u} R_{u,v}(\tau). \quad (11)$$

Because of the monotone relation between the normal and the relais-correlation function, maximizing $R_{u,v}(\tau)$ or $R_{sgn(u),v}(\tau)$ leeds to the same estimate of the motion parameters. The same holds for the polarity- correlation $R_{sgn(u),sgn(v)}(\tau)$ [11]

$$\begin{aligned} R_{sgn(u),sgn(v)}(\tau) &= E\{sgn(u(t+\tau))sgn(v(t))\} \\ &= \frac{2}{\pi}\arcsin\frac{R_{u,v}(\tau)}{\sigma_u\sigma_v}. \end{aligned} \quad (12)$$

The prerequisite of gaussian distributed grey-scale images is too strong for many applications. It is sufficient that $R_{sgn(u),v}(\tau)$ and $R_{sgn(u),sgn(v)}(\tau)$ are arbitrary monotone functions of $R_{u,v}(\tau)$ so that the maxima of these functions are achieved at the same parameter value τ. With these assumptions the following simplifications of the algorithm of eq. (6) are possible.

Relais case:

$$\hat{\mathbf{T}}^{K+1} = \mathbf{A}^{K+1}\hat{\mathbf{T}}^{K} - \tilde{\mathbf{H}}_R^{-1} \cdot \tilde{\mathbf{g}}_R(\hat{\mathbf{T}}^{K}) \tag{13}$$

with the modified Hessian $\tilde{\mathbf{H}}_R$

$$\tilde{H}_{Rij} = -2E\left\{\frac{\partial^2 I_1(\mathbf{x})}{\partial x_i \partial x_j} sgn(I_1(\mathbf{x}))\right\} \tag{14}$$

and the modified gradient vector $\tilde{\mathbf{g}}_R(\hat{\mathbf{T}}^K) = \mathbf{B}^K \cdot \mathbf{g}_R(\hat{\mathbf{T}}^K)$,

$$g_{Ri}(\hat{\mathbf{T}}^K) = -2E\left\{(sgn(S(\mathbf{x}^K)) - sgn(I_2(\mathbf{x}))\frac{\partial I_2(\mathbf{x})}{\partial x_i}\right\} \tag{15}$$

where $S(\mathbf{x})$ and $I_2(\mathbf{x})$ are assumed to have a zero mean value. The main advantage of this algorithm is the fact that within the iteration loop the image difference of the clipped signals may only assume the three values -2, 0, +2. Therefore no full realization of the multiplication is necessary, it is sufficient to realize polarity controlled addition and subtraction.

Polarity-case:

A further reduction of the numeric complexity is possible with

$$\hat{\mathbf{T}}^{K+1} = \mathbf{A}^{K+1}\hat{\mathbf{T}}^{K} - \tilde{\mathbf{H}}_P^{-1} \cdot \tilde{\mathbf{g}}_P(\hat{\mathbf{T}}^{K}) \tag{16}$$

with the modified gradient vector $\tilde{\mathbf{g}}_P(\hat{\mathbf{T}}^K) = \mathbf{B}^K \cdot \mathbf{g}_P(\hat{\mathbf{T}}^K)$,

$$g_{Pi}(\hat{\mathbf{T}}^K) = -2E\left\{(sgn(S(\mathbf{x}^K)) - sgn(I_2(\mathbf{x}))sgn\left(\frac{\partial I_2(\mathbf{x})}{\partial x_i}\right)\right\} \tag{17}$$

and the Hessian $\tilde{\mathbf{H}}_P$ which has to be expressed by differentiating the gradient vector g_{Pj} with respect to $\tilde{T}_i$

$$\tilde{H}_{Pij} = -\frac{\partial}{\partial \tilde{T}_i} 2E\left\{(sgn(S(\mathbf{x},\tilde{\mathbf{T}})) - sgn(S(\mathbf{x}))) \times sgn\left(\frac{\partial S(\mathbf{x})}{\partial x_j}\right)\right\}\Big|_{\tilde{\mathbf{T}}=0}. \tag{18}$$

Calculating $\tilde{\mathbf{H}}_P$ directly from the derivatives of the images is not possible because clipping as a nonlinear operation and differentiating do not commute.

The realization of this algorithm is quite simple. Because of the fact that the image difference in eq.(18) may assume only the three values -2, 0, +2 and the derivatives only the two values -1 and +1 calculating $\mathbf{g}_P(\hat{\mathbf{T}}^K)$ reduces to simple up/down-counting.

Comparing the three algorithms (eq.(6), eq.(13), eq.(16)) we may observe a similar structure. The characteristic performance as well as the large stability region and the signal adaptive convergence rate remain unchanged. The difference between the algorithms consists in the complexity of the arithmetic operations and in a simplified, namely binary image memory. Above that there might be an additional impact on the characteristics of the error functions.

4. EXPERIMENTS WITH REAL IMAGE DATA

The algorithms (eq.(6), eq.(13), eq.(16)) have been tested with real image data. Image 1 and 2 show two typical scenes digitized by 512 x 512 pixels which were used as $I_1(\mathbf{x})$. The figures 1, 2 and 3 give the result of characteristic experiments. The images were translated and rotated in the computer by definite motion vectors $\mathbf{T}$ to generate $I_2(\mathbf{x})$ using bilinear interpolation. The rotation always was around the centre of the marked areas and the smallest of these areas (51 x 51 pixels) were used as region of interest to calculate the expectation values within the identification of the given motion vectors T.

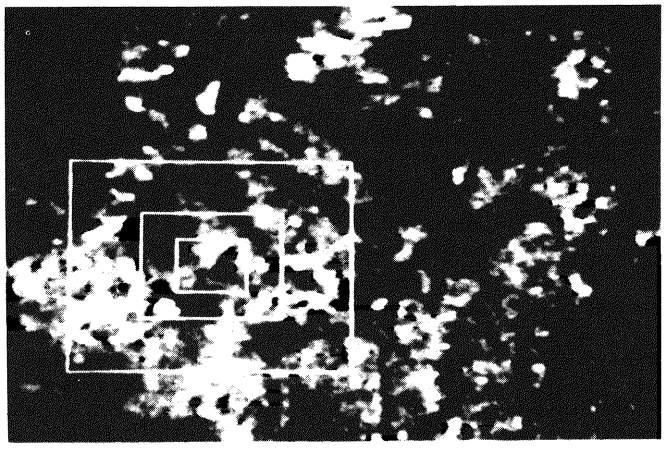

Image 1: Satellite image of 512 x 512 pixels

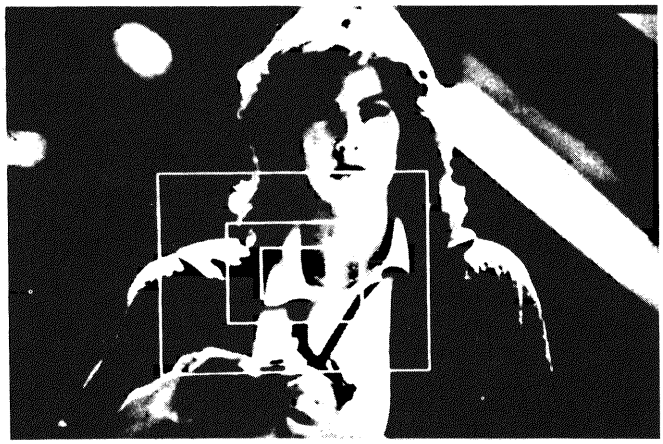

Image 2: Picture of a woman digitized by 512 x 512 pixels

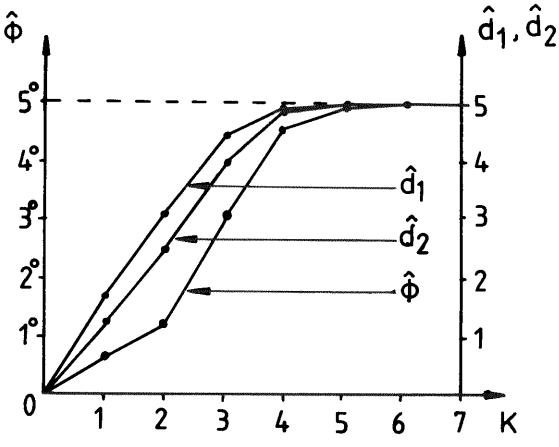

Fig. 1: Estimation of $\mathbf{T} = (5°, 5, 5)^T$ using image 1

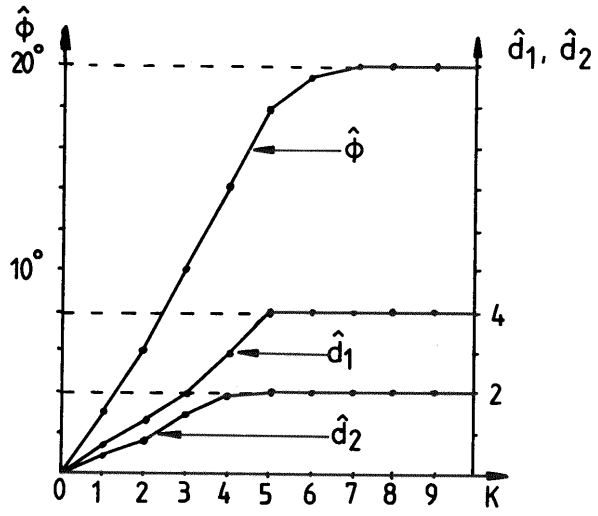

Fig. 2: Estimation of $\mathbf{T} = (20^o, 4, 2)^T$ using image 2

The first and second figure show a joint identification of rotation and translation using the normal modified Newton-Raphson-algorithm. In the first example using image 1 the motion vector to be identified was $\mathbf{T} = (5^o, 5, 5)^T$. In the second example image 2 was used with the motion vector $\mathbf{T} = (20^o, 4, 2)^T$. The estimated values $\hat{\phi}$ in degrees and $\hat{d}_1, \hat{d}_2$ in pixels are plotted versus the iteration number K. Both examples show that the given motion vector could be identified in a few iterative steps. Furthermore near the optimum the innovations are large and the estimation vector $\hat{\mathbf{T}}$ converges very fast to the true value $\mathbf{T}$ which states the good convergence property near the optimum. Especially example 2 shows the large region of stability up to more than 20° in this scene.

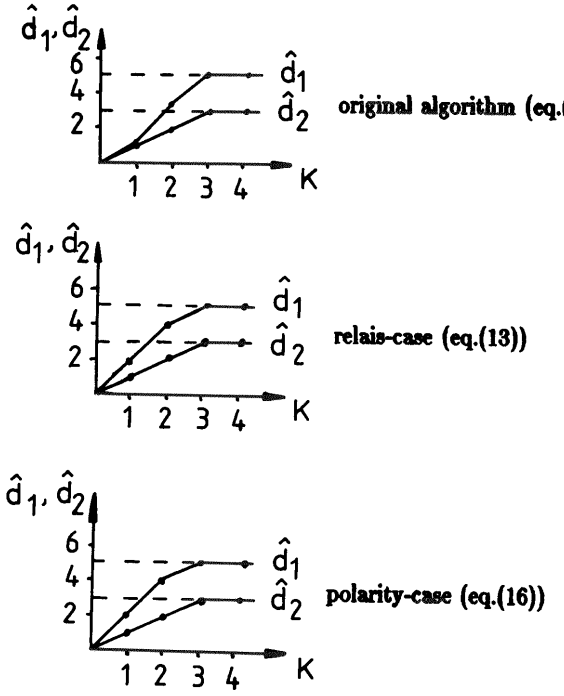

Fig. 3: Estimation of $\mathbf{T} = (0^o, 5, 3)^T$ using image 1

The third example using image 1 compares the three algorithms (eq.(6), eq.(13), eq.(16)) in the case that only translation is present. The given vector $\mathbf{T} = (0^o, 5, 3)^T$ is always identified in a few steps. There are no great differences between the normal algorithm, the relais- and polarity-case. Note that the innovations in contrast to the former examples are always in multiples of full pixels because here no subpixel interpolation was used and the estimated values were rounded.

5. CONCLUSIONS

A fast converging algorithm to jointly estimate rotation and translation in image sequences has been presented and tested using real image data. The special features of the algorithm, the high, image bandwidth adaptive convergence rate, the large region of stability and a low numeric complexity of the algorithm could be verified succesfully. An extension of the algorithm to estimate affine transform parameters is given in [9].

REFERENCES

[1] Nagel, H.H.: Image sequence analysis: What can we learn from applications? Huang, T.S. (ed.): Image Sequence Analysis. Springer, 1981.

[2] Nagel, H.H.: Analyse und Interpretation von Bildfolgen. Informatik Spektrum 8, 1985, pp. 178-200 and pp. 312-327.

[3] Huang, T.S. (ed.): Proc. Nato Advanced Study Inst. on Image Sequence Processing and Dynamic Scene Analysis. Braunlage 1982, Springer, 1983.

[4] Schalkoff, R.J.; McVey, E.S.: A model and tracking algorithm for a class of video targets. IEEE Trans. PAMI-4, 1982, pp. 2-10.

[5] Legters, G.R.; Young, T.Y.: Mathematical model for computer image tracking. IEEE Trans. PAMI-4, 1982, pp. 583-594.

[6] Axelsson, S.R.J.: On optimum algorithms for imaging tracking systems. Kunt, M.; de Coulon, F. (eds.): Signal Processing: Theories and applications. North-Holland, 1980, pp. 723-728.

[7] Lenz, R.; Gerhard, A.: Adaptive geometrische Transformationen zur Mustererkennung mit Hilfe eines linearen, lokalen Distanzmaßes. Niemann, H.: Mustererkennung 1985, 7. DAGM-Symposium, Informatik-Fachberichte 107. Springer, 1985, pp. 112-117.

[8] Jerian, C.; Jain, R.: Determining motion parameters for scenes with translation and rotation. IEEE Trans. PAMI-6, 1984, pp. 523-530.

[9] Burkhardt, H.; Diehl, N.: Simultaneous estimation of rotation and translation in image sequences. Young, I.T. et al. (eds.): Proceedings of EUSIPCO-86. The Hague, 1986, North Holland, 1986.

[10] Burkhardt, H.; Moll, M.: A modified Newton-Raphson-search for the model-adaptive identifications of delays. Isermann, R. (ed.): Identification and System Parameter Estimation. Pergamon, 1979, pp. 1279-1286.

[11] Papoulis, A.: Probability, random variables and stochastic processes. McGraw-Hill, 1984.