

# INFRARED IMAGE ENHANCEMENT WITH NONLINEAR SPATIO-TEMPORAL FILTERING

Wilhelm MEIER, H.D. vom STEIN

Universität der Bundeswehr Hamburg,  
Allgemeine Nachrichtentechnik, Holstenhofweg 85, D-2000 Hamburg 70,  
Fed. Rep. Germany,  
Phone: 0049/040/6541-2545, Fax: 0049/040/6530413  
ISPRS Commission II

This article describes a nonlinear spatio-temporal filter method for infrared images aimed to produce image data with low noise and high accurate, prominent edges.

We present a novel, robust and fast one-dimensional phase correlation algorithm to determine the translation displacement from frame to frame in image sequences. The image sequence is then compensated according to its displacements and filtered with a new edge preserving smoothing algorithm. As a consequence of this filter algorithm simple edge detectors such as Roberts gradient could then be used.

Apart from that we present an application of the one-dimensional phase correlation algorithm to detect a rotation around arbitrary points combined with a translation between two frames in the image sequence.

To all of these algorithms results are given.

Key Words: Image enhancement, Correlation, Nonlinear spatio-temporal filter

## 1 INTRODUCTION

The automatic analysis of infrared images with digital computers makes great demands on the combined picture preprocessing especially the image enhancement with regard to later feature extraction. Due to the sensors used infrared images are of minor quality compared to those derived with sensors for the visual channel (CCD-sensors). They suffer at most from low signal to noise ratio, low contrast at object boundaries and image artifacts produced by the sensor. The usual steps in image preprocessing such as linear and nonlinear filtering performed on single images are not sufficient enough to guarantee almost errorfree results of the following segmentation procedures. Especially edge detection is a difficult task in IR-images.

A substantial amount of quality enhancement one can get with filtering the image signal not only in space but also in time. One problem which arises is the movement of the object(s) during the acquisition time of the image sequence. Before one can perform appropriate filtering this movement has to be calculated and compensated. There exists a wealth of algorithms to detect translational movements, most of them with high computational amount. In particular the outstanding performance of the 2D phase correlation method should be mentioned [3].

If the relevant objects have special features in the IR signal, such as hot spots, it is possible to use a *one dimensional phase correlation function* (1D-PCF) to determine its movements. This leads to very low computational costs

combined with the accurate results as in the 2D-PCF [3]. The presented algorithm requires an appropriate correlation window in the image sequence which includes one hot spot (Candidate Target Detection Algorithm, [2]). The movement of the hot spot then represents the movement of the object, which is a ship in our case.

After the image sequence has been compensated according to the detected movements it can be filtered. For this purpose we present a novel nonlinear spatio-temporal filter. This 'edge preserving smoothing' algorithm (EPS) has the great advantage not to smooth out the edges as filters - such as the simple lowpass filter - do. Therefore it is particularly suitable for edge detection.

Supposing the relevant object not only performs a translation but a rotation, the movement can also be calculated by the 1D-PCF applied to the absolute values of the 2D Fourier transform of the correlation window.

## 2 IMAGE ALIGNMENT WITH THE 1D-PHASE CORRELATION

The basis of the phase correlation is the fact that two identical but shifted signals differ in the Fourier domain only by a linear phase. This corresponds with the observation that the phase in the Fourier domain contains most information about the signal [5].

Let  $x_1(n)$  and  $x_2(n)$  be two discrete signals of length  $N$  with  $x_2(n)$  a cyclically shifted copy of  $x_1(n)$ . Then their corresponding Fourier transforms  $X_1(k)$  and  $X_2(k)$  are re-

lated as follows

$$x_1(n) \longleftrightarrow X_1(k) \quad (1)$$

$$x_2(n) = x_1((n - n_0))_N \longleftrightarrow X_2(k) \quad (2)$$

$$\begin{aligned} X_2(k) &= X_1(k)e^{-j\frac{2\pi}{N}n_0k} \\ &= X_1(k)W_N^{n_0k} \end{aligned} \quad (3)$$

$$x_1((n - n_0))_N : \text{cyclically shifted } x_1(n)$$

A correlation in the original domain corresponds to a conjugate complex product in the Fourier domain. If we divide this product by its absolute values and perform the inverse Fourier transformation, we get the phase correlation function (PCF) of the signals  $x_1(n)$  and  $x_2(n)$  which is a delta impuls in the ideal case.

$$\frac{X_1(k) \cdot X_2^*(k)}{|X_1(k) \cdot X_2^*(k)|} = W_N^{-n_0k} \longleftrightarrow \delta(n + n_0) \quad (4)$$

To align two images in [3] it is suggested to perform a 2D-PCF on the entire images. Since this is very time consuming the first step in our computation scheme is to set an appropriate correlation window of a smaller spatial extent depending on the relevant objects. In order to achieve reliable results such correlation windows should include image areas with high contrast.

In the case of processing IR-images containing objects with hot areas (hot spots), as are ships in our case, such windows are easily determined using well known candidate target detection algorithms (CTDA) [2]. The hot spot should be placed in the middle of the window and two signals  $x_1(n)$  and  $y_1(m)$  are then calculated projecting the pixel grey values in the window towards its borders. The signals  $x_i(n)$  and  $y_i(m)$  of the following images  $i, 2 \leq i \leq L$  are computed in exact the same way. If the objects move from frame to frame the signals  $x_i(n)$  and  $y_i(m)$  are *aperiodically* shifted copies of the signals  $x_1(n)$  and  $y_1(m)$  in contrast to (2) (Figure 1). The direct computation of the phase correlation functions  $PCF(x_1(n), x_i(n))$  and  $PCF(y_1(m), y_i(m))$  leads in most cases to a wrong position of the maximum in the PCF. The Fourier transforms of real signals decrease approximately with  $f^{-1}$  towards high frequencies. The normalization procedure inside the PCF (4) process ‘whitens’ the spectrum of the signals and as a consequence the high frequency components gain more influence on the result. Thus the PCF is heavily dominated by discontinuities from the left to the right border of the signals. The PCF of the two signals  $x_1(n)$  and  $x_2(n)$  in figure 1 would result in a wrong peak at location  $n_0 = 0$  although  $x_2(n)$  is shifted 4 samples to the right.

To avoid this it is absolutely necessary to use a weighting function for the input signals  $x_i(n)$  and  $y_i(m)$  to the PCF which forces them to zero at their borders.

Apart from that the signals  $\tilde{x}_i(n)$  and  $\tilde{y}_i(m)$  with

$$\tilde{x}_i(n) = x_i(n) - m_{xi} \quad (5)$$

$$\tilde{y}_i(m) = y_i(m) - m_{yi} \quad (6)$$

$$m_{xi}, m_{yi} : \text{mean of signals } x_i(n), y_i(m)$$

should be used instead of  $x_i(n)$  and  $y_i(m)$ . Otherwise the PCF could be dominated by the shape of the weighting

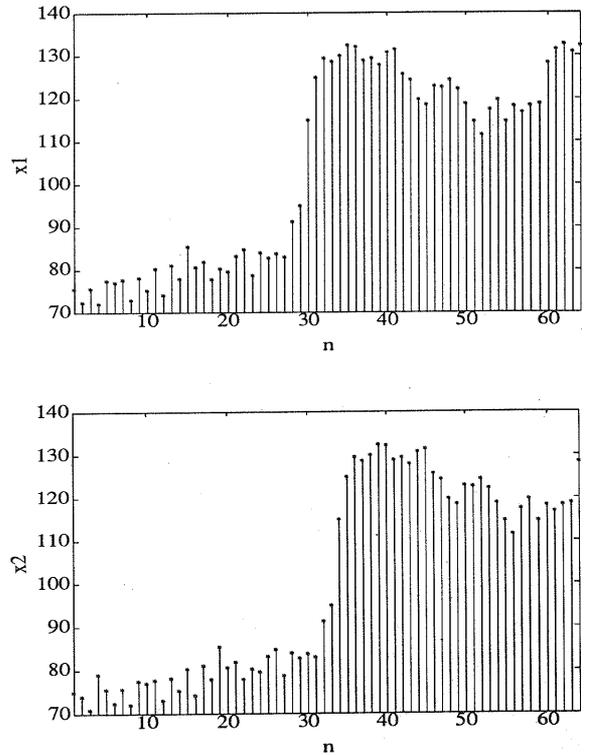


Figure 1: Signals  $x_1(n)$  (top) and  $x_2(n)$  (bottom) of a real image sequence

function if the constant part of the signals is large compared with the alternating parts and the maximum peak might be displaced towards  $n_0 = 0$ .

The previous considerations can now be formulated in the following computation steps to calculate the displacement vector from frame to frame:

1. Choose a correlation window of size  $N \times N$  (CTDA)
2. Compute the projections of the pixel grey values  $g_i(n, m)$  towards the window borders in each image  $i$  of the sequence

$$\begin{aligned} x_i(n) &= \sum_{m=0}^{N-1} g_i(n, m) \\ y_i(m) &= \sum_{n=0}^{N-1} g_i(n, m) \end{aligned}$$

3. Modify the signals

$$\begin{aligned} \tilde{x}_{wi}(n) &= w(n) [x_i(n) - m_{xi}] \\ \tilde{y}_{wi}(m) &= w(m) [y_i(m) - m_{yi}] \\ w(n) &: \text{weighting function} \\ m_{xi}, m_{yi} &: \text{mean of signals } x_i(n), y_i(m) \end{aligned}$$

4. Compute  $PCF(\tilde{x}_{w1}(n), \tilde{x}_{wi}(n))$  and  $PCF(\tilde{y}_{w1}(m), \tilde{y}_{wi}(m))$
5. Search in both PCFs the peak positions and store them as displacement  $(n_0, m_0)$ :

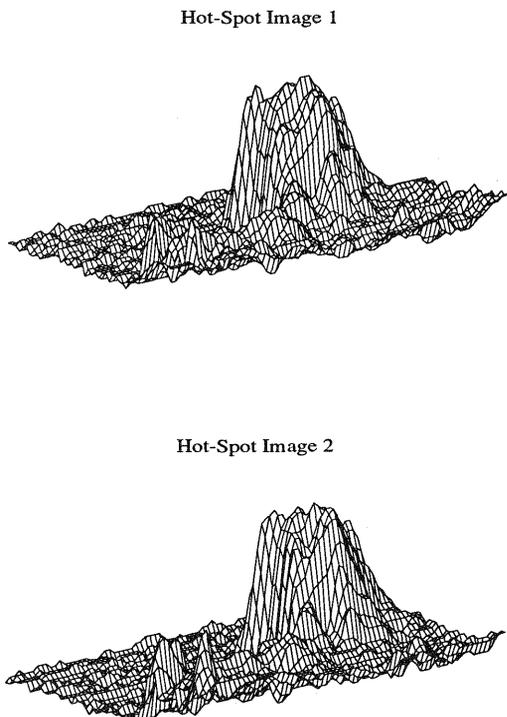


Figure 2: Correlation window of image 1 (top) and of image 2 (bottom) of a real image sequence

A *Hanning window* is selected as the weighting function  $w(n)$  in step 3 of the above algorithm which produces low distortions in the frequency domain. In step 4 the normalized conjugate complex product inside the PCF (4) could be weighted as well with an appropriate function (7).

$$PCF(x_1(n), x_2(n)) \longleftrightarrow \frac{X_1(k) \cdot X_2^*(k)}{|X_1(k) \cdot X_2^*(k)|} \cdot M(k) \quad (7)$$

$M(k) \quad :$

If high frequency noise is present in the signals, good results are achieved with a *Hanning window* which mutes high frequencies.

Figure 2 - Figure 4 show the results of the PCF applied to real IR-signals. Clearly visible are the very sharp peaks at the registration points which are only about one resolution element in width. The PCF can not be disturbed by convolutional degradations to the signals (8) and it can be shown that it is fairly insensitive to unshifted superimposed degradations such as faults in the detector elements of the sensor.

$$\begin{aligned} \hat{x}_1(n) = x_1(n) * g(n) &\longleftrightarrow \hat{X}_1(k) \\ \hat{x}_2(n) = x_2(n) * g(n) &\longleftrightarrow \hat{X}_2(k) \\ g(n) &\longleftrightarrow G(k) \\ \frac{\hat{X}_1(k) \cdot \hat{X}_2^*(k)}{|\hat{X}_1(k) \cdot \hat{X}_2^*(k)|} &= \frac{X_1(k) \cdot X_2^*(k) |G(k)|^2}{|X_1(k) \cdot X_2^*(k)| |G(k)|^2} \quad (8) \end{aligned}$$

Table 1 compares the computational amount of the 1D-PCF with the modified 2D-PCF according to [3] with weighting functions in original and frequency domain.

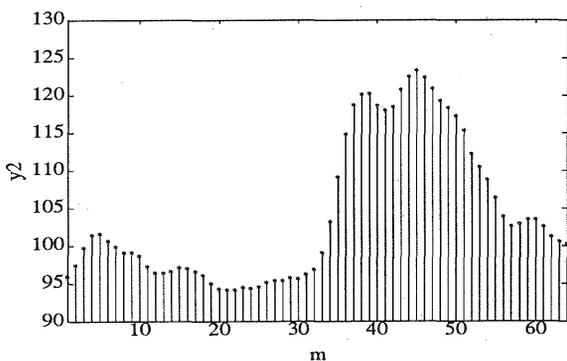
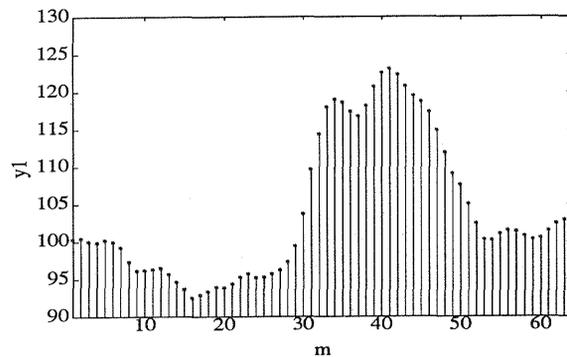


Figure 3: Projections of the correlation windows 1 (top) and 2 (bottom) in y-direction. See figure 1 for x-projections

### 3 EDGE PRESERVING SMOOTHING

After the image sequence has been compensated so that the relevant object is at the same position in all images, a nonlinear spatio-temporal filter is applied to the image sequence. In order to avoid blurring of edges, averaging must not be applied to areas containing edges. Thus a simple lowpass filter, which averages over an isotrope mask volume through the image sequence, is fairly unsuitable. It is rather necessary to find the most homogeneous mask volume around the point to be smoothed. The variance of a mask volume can serve as a measure of inhomogeneity [4]. We use the eight volumina shown in Figure 5 ( $M_1 \dots M_8$ ) and the isotrope volume  $M_9$ , which includes the 8-neighbourhood of  $p_e(n, m)$  and its pixels in the following images.

Now we can summarize the edge preserving smoothing algorithm as follows:

- For each point  $p_e(n, m)$  of the first of L images do:
  1. Compute the 9 mask volume variances  $Var(M_1) \dots Var(M_9)$
  2. Find the minimum variance mask volume  $M^* : Var(M^*) \leq Var(M_i); 1 \leq i \leq 9$
  3. Set  $p_o(n, m) = \frac{1}{N^*} \sum_{M^*} p(n, m)$  in the output image ( $N^*$ : Number of pixels of mask volume  $M^*$ )

Table 2 shows the result of the EPS-filter compared with a conventional lowpass filter for a test sequence corrupted

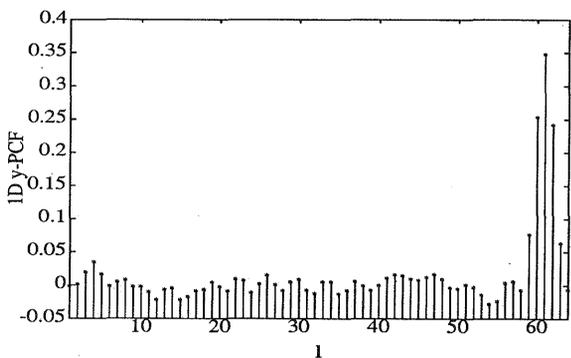
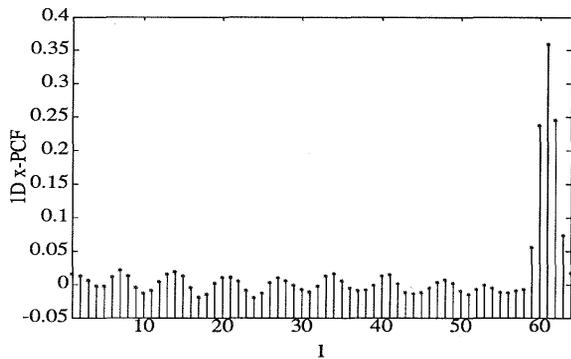
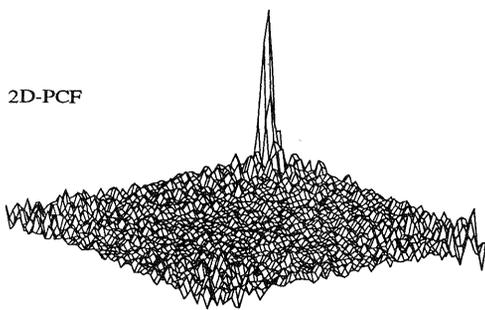


Figure 4: Results of the 2D-PCF (top) and 1D-PCF for x-direction (mid) and y-direction (bottom)

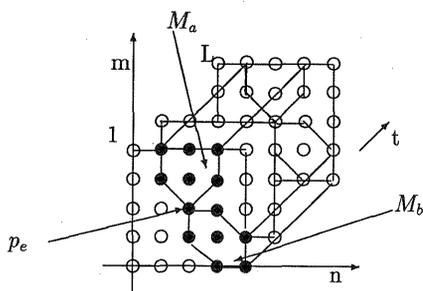


Figure 5: The mask volumina  $M_1 \dots M_8$  are constructed by  $M_a$  and  $M_b$ , each rotated 3 times by  $\frac{\pi}{2}$ . Each mask volume extends over  $L$  images.

Operation	2D-PCF	1D-PCF (both directions)
real multiplication	$N^2(3 + 8 \log_2(N^2))$	$2N(3 + 8 \log_2(N))$
real additions	$8N^2 \log_2(N^2)$	$2N(N - 1) + 16N \log_2(N)$
$N = 64$		
real multiplications	405.504	6.528

Table 1: Computational amount of the 1D-PCF and 2D-PCF including weighting in original and frequency domain and computation of the correlation window projections for the 1D-PCF

	original sequence	result of lowpass-filter $1 \times 1 \times L$	result of lowpass-filter $5 \times 5 \times L$	result of EPS filter
$\sigma$ of image	24.3	27.4	22.8	24.7
$\sigma$ of noise	40.0	13.5	10.9	8.5

Table 2: Image variances after various filters applied to the testsequence corrupted with gaussian noise  $\sigma = 40.0$ , ( $L$ : number of images in sequence)

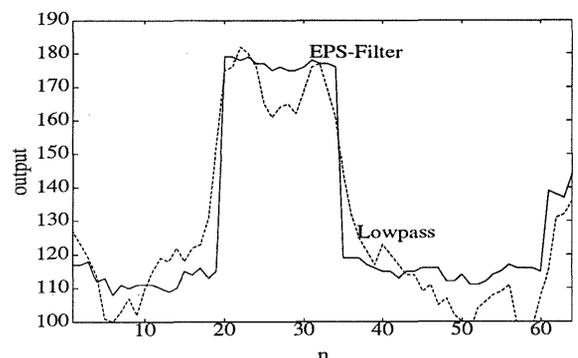
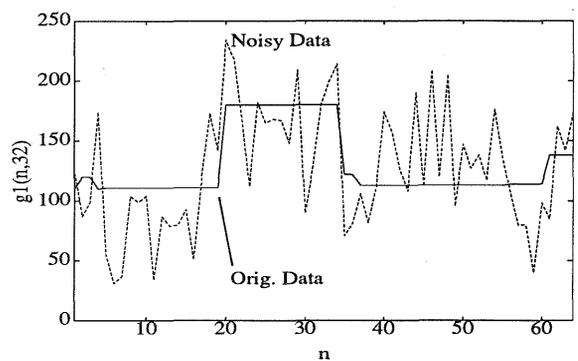


Figure 6: Cross section of the results of the EPS-filter applied to the testsequence corrupted with gaussian noise  $\sigma = 40.0$

with gaussian noise. Figure 6 points out the remarkable capability of the EPS-filter to preserve edges. As the EPS-filter achieves the best noise-reduction combined with the outstanding characteristic to preserve edges in position and slope it is well suited for edge detection.

#### 4 ROTATION DETECTION

Let us assume  $g(x, y)$  are the grey values of an image and  $F(u, v)$  is its Fourier transform (FT). With the Fourier theorem

$$g(a_1x + b_1y, a_2x + b_2y) \longleftrightarrow \frac{1}{|a_1b_2 - a_2b_1|} F(A_1u + A_2v, B_1u + B_2v) \quad (9)$$

$$\begin{bmatrix} A_1 & B_1 \\ A_2 & B_2 \end{bmatrix} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix}^{-1}$$

it follows

$$g(x \cos \varphi + y \sin \varphi, -x \sin \varphi + y \cos \varphi) \longleftrightarrow F(\omega_x \cos \varphi + \omega_y \sin \varphi, -\omega_x \sin \varphi + \omega_y \cos \varphi) \quad (10)$$

That is, the FT rotates by the same angle as the image itself. An additional translation of the image affects its FT only by a bilinear phase. In other words, the magnitude of the FT is shift-invariant.

$$g(x - x_0, y - y_0) \longleftrightarrow F(\omega_x, \omega_y) \exp(-j\omega_x x_0) \exp(-j\omega_y y_0) \quad (11)$$

Let  $|F_1(r, \theta)|$  and  $|F_2(r, \theta)|$  denote the magnitude of the FTs in polar coordinates of two images  $g_1(x, y)$  and  $g_2(x, y)$  of an image sequence. Then we can define two radial projections of the magnitude functions

$$\tilde{a}_1(\theta) = \int_0^\infty |F_1(r, \theta)| r \, dr \quad (12)$$

$$\tilde{a}_2(\theta) = \int_0^\infty |F_2(r, \theta)| r \, dr \quad (13)$$

which are periodic with respect to  $\theta$ . If image  $g_2(x, y)$  is a rotated (and perhaps translated(!)) copy of the image  $g_1(x, y)$ , then  $\tilde{a}_2(\theta) = \tilde{a}_1((\theta + \Delta\theta))_{2\pi}$  is a cyclically shifted version of  $\tilde{a}_1(\theta)$ .

Therefore the angle  $\Delta\theta$  could be easily determined using techniques such as the 1D-PCF. But as a consequence of the digital geometry  $\tilde{a}_1(\theta)$  and  $\tilde{a}_2(\theta)$  are difficult to calculate.

Since the signals in (12), (13) are periodic functions of  $\theta$  it is possible to calculate the coefficients  $A_{1k}$  and  $A_{2k}$  of their Fourier series representation. It's noteworthy for the implementation that the discrete Fourier series (DFS) of a periodic signal equals the DFT of the aperiodic signal.

$$A_{1k} = \frac{1}{2\pi} \int_0^{2\pi} \int_0^\infty |F_1(r, \theta)| r \, dr \exp\left(-j\frac{2\pi}{2\pi} k\theta\right) \partial\theta$$

$$= \frac{1}{2\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |F_1(u, v)| \sqrt{u^2 + v^2} \exp\left(-jk \arccos(z)\right) \begin{vmatrix} \frac{\partial r}{\partial u} & \frac{\partial r}{\partial v} \\ \frac{\partial \theta}{\partial u} & \frac{\partial \theta}{\partial v} \end{vmatrix} \partial u \partial v \quad (14)$$

$$\text{with } z = \frac{u}{r}; \quad r = \sqrt{u^2 + v^2}$$

$$\theta = \arccos\left(\frac{u}{\sqrt{u^2 + v^2}}\right)$$

For the discrete implementation follows

$$A_1(k_\theta) = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1} |F_1(n, m)| \exp(-jk_\theta \arccos(y)) \quad (15)$$

$$\text{with } y = \frac{n}{\sqrt{n^2 + m^2}}$$

With the two FTs  $A_1(k_\theta)$  and  $A_2(k_\theta)$  a normal cross-correlation (CCF) or phase correlation (PCF) can be computed.

$$PCF_{rot}(l) \longleftrightarrow \frac{A_1(k_\theta) \cdot A_2^*(k_\theta)}{|A_1(k_\theta) \cdot A_2^*(k_\theta)|} \quad (16)$$

$$CCF_{rot}(l) \longleftrightarrow A_1(k_\theta) \cdot A_2^*(k_\theta) \quad (17)$$

The position of the peak in  $PCF_{rot}(l)$  or  $CCF_{rot}(l)$  reveals the angle  $\Delta\theta$  of the rotation.

The direct computation of (15) is very time consuming. But it can be computed very effectively by a recursive procedure using Chebyshev polynomials [1].

Practical tests of the algorithm have shown that only a few samples of the FT in (15), that is only few Fourier coefficients have to be determined (Figure 7). The appropriate interpolation to achieve a sufficient accuracy for the result can be attained by an enlarged length of the inverse FT in (16) and (17). Thereby less time is needed to evaluate (15) whereas a greater length of the inverse FT is not of great importance if the FFT algorithm is used.

To detect an additional translation it is suggested in [1] to perform a backrotation of  $F_2(\omega_x, \omega_y)$  and then to calculate the displacement via a 2D-PCF of  $F_1(\omega_x, \omega_y)$  and the backrotated  $F_2(\omega_x, \omega_y)$ . Because the 2D-PCF is fairly sensitive to phase errors, it is very difficult to find an appropriate interpolation algorithm for the backrotation of  $F_2(\omega_x, \omega_y)$ . Much better and less timeconsuming results are achieved if we perform the backrotation on  $g_2(x, y)$  and align the images via the 1D-PCF method.

Figure 7 shows the results of the rotational correlation in the case of rotated checkerboard ( $256 \times 256$ ) and rotated real images using various number of Fourier samples. The practical tests with real images have shown that the algorithm is robust against image degradations. It's noteworthy that for a checkerboard pattern 4 possible symmetries exist, as detected in Figure 7. The two major peaks in Figure 8 are explainable if we recall that the magnitude of the 2D-FT is used in order to be independent of image translations. Therefore rotations in the intervals  $0 \leq \Delta\theta \leq \pi$  and  $\pi \leq \Delta\theta \leq 2\pi$  are indistinguishable.

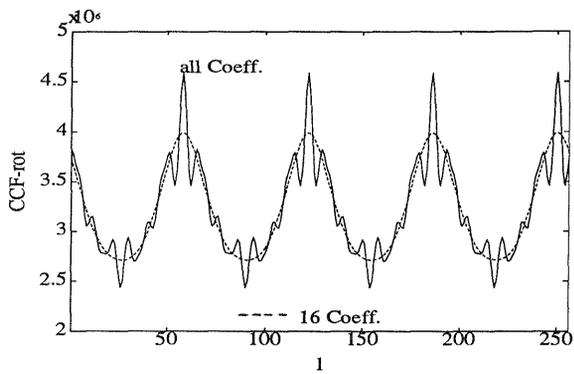


Figure 7: Rotational normal correlation (CCF) applied to checkerboard images

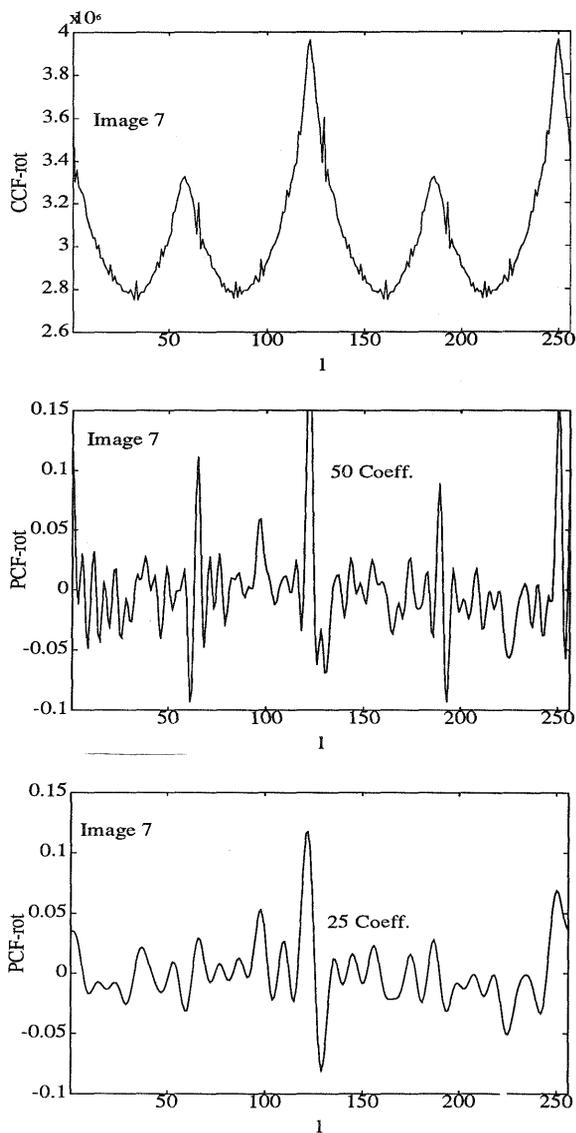


Figure 8: Rotational normal correlation (CCF) applied to real IR-images (top) and rotational phase correlation (PCF) applied to real IR-images with various number of Fourier coefficients (mid, bottom)

## 5 CONCLUSION

In this paper we describe a new spatio-temporal filter scheme for image sequences with moving objects. Before the filter algorithm can take place, the images are aligned via a fast and accurate 1D-phase correlation method especially well suited for IR-images. Thereafter a nonlinear spatio-temporal filter, which has the capability to preserve edges, is applied to the motion-compensated sequence. If there is the need to detect not only translations but also image rotations we propose an algorithm based on the 1D-phase correlation applied to projections of the magnitude of the 2D Fourier transform.

## REFERENCES

- [1] S. Alliney and C. Morandi. Measuring rotations and translations of digitized images. In *Third European Signal Proc. Conf.*, pages 897-901. EUSIPCO, September 1986.
- [2] M. Burton and C. Benning. Comparison of imaging infrared detection algorithms. *SPIE:Infrared Techn. for Target Detect. and Classif.*, 302:26-32, 1981.
- [3] C.D. Kuglin and D.C.Hines. The phase correlation image alignment method. *Proc. IEEE 1975 Int. Conf. Cybern. and Soc.*, pages 163-165, September 1975.
- [4] M. Nagao and T. Matsuyama. Edge preserving smoothing. *Computer, Graphics and Image Processing*, 9:394-407, 1979.
- [5] A.V. Oppenheim and J.S. Lim. The importance of phase in signals. *Proc. IEEE*, 69(5):529-541, May 1981.