Edge and Line Feature Extraction Based on Covariance Models

Ferdinand van der Heijden

Abstract — **Image segmentation based on contour extraction usually involves three stages of image operations: feature extraction, edge detection and edge linking. This paper is devoted to the first stage: a method to design feature extractors used to detect edges from noisy and/or blurred images. The method relies on a model that describes the existence of image discontinuities (e.g. edges) in terms of covariance functions. The feature extractor transforms the input image into a "log-likelihood ratio" image. Such an image is a good starting point of the edge detection stage since it represents a balanced trade-off between signal-to-noise ratio and the ability to resolve detailed structures. For 1-D signals, the performance of the edge detector based on this feature extractor is quantitatively assessed by the so called "average risk measure." The results are compared with the performances of 1-D edge detectors known from literature. Generalizations to 2-D operators are given. Applications on real world images are presented showing the capability of the covariance model to build edge and line feature extractors. Finally it is shown that the covariance model can be coupled to a MRF-model of edge configurations so as to arrive at a maximum *a posteriori* estimate of the edges or lines in the image.**

Index Items — **Edge detection, line detection, image processing, image segmentation, feature extraction, MRF.**

I. Introduction

In a broad sense, the term "edge detection" refers to the process that involves the discovery of three dimensional edges (such as object boundaries, shadow boundaries, abrupt changes in surface orientation, or material properties) in a scene given one or more images of that scene. Often, these 3-D edges are seen in the image data by discontinuities of certain local image properties: irradiance, chromaticity values, texture, etc. The importance of edge detection in computer vision follows from the fact that 3-D edges are clues for the characterization of the scene.

In a narrow sense, the term "edge detection" refers to only one aspect of the process mentioned above, the detection and localization of local image features in the image plane. Most workers in the field of edge detector design apply themselves to an even more restrictive definition: the detection and localization of discontinuities of the irradiance in the image plane. In this paper, the last definition will be used.

A. Previous Work

Most edge detection techniques consist of two stages: a feature extraction stage and a stage in which the actual assignment of points to edges or non-edges takes place. Sometimes the procedure is followed by a third stage in which the detected edges are examined once again for instance to obtain closed contours, e.g. edge linking. The term feature extraction comes from pattern classification theory. Its goal is twofold. Firstly, to transform the original image data into features that are more useful for classification than the original data. Secondly, to reduce the computational complexity of a detection or classification problem. In our case, *edge feature extraction* refers to the transform of the observed image into one or more generalized images on which the actual detection of edges is based.

The second stage, the assignments of points to edges and non-edges, could simply be based on the comparison of the edge features with a threshold. In most cases, this leads to large localization errors and many multiple detections of a single edge. Therefore, the comparison operation (thresholding) is often combined with the detection of local maximums of the edge features in some suitably chosen direction in the image plane (i.e. non-local maximum suppression), or with the detection of zero-crossings of a (directional) derivative.

Often, the design of an edge detector is based on a model of a 1-D signal. Such a signal could be, for instance, a cross section or a projection from 2-D image data. Under the assumption that boundaries in the scene are seen in the image plane as transitions of the irradiance, the detection of these boundaries transforms to finding the locations of the transitions in the 1-D signal. Edges in the image are often modeled as step functions in the projection. Lines are supposed to manifest themselves in the projection as pulse shaped functions of finite extent. Sometimes the transitions are considered to be abrupt. In these cases the edge model is a Heaviside step function or *ideal* step function. In more realistic models, the spatial resolution of the imaging device is taken into account. In such cases, a transition corresponding to an edge occurs more smoothly. The corresponding step function is referred to as the *edge profile,* sometimes modeled by the edge spread function of the imaging device. Likewise, lines are modeled by the line spread function, here referred to as *line profile*.

Intuitively, differential operators seem to be well suited to enhance step functions and other discontinuities in observed signals. However, since differentiation is an ill-posed problem, all edge feature extractors relying on differential operators (like the Laplacian, the gradient and higher order directional derivatives) need a filtering stage in order to suppress noise

Manuscript received Aug. 20, 1993; revised Mar. 7, 1994. Recommended for acceptance by Dr. Terry Caelli.

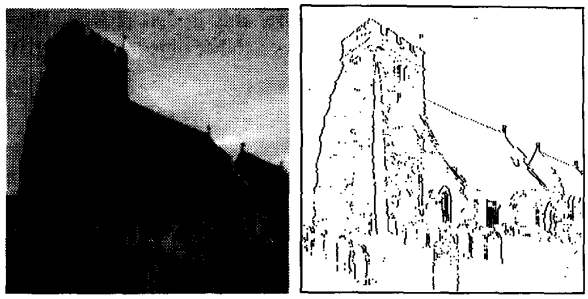
F. van der Heijden is with the Department of Electrical Engineering, University of Twente, PO 217, 7500AE, Enschede, The Netherlands; e-mail: ferdi@mi.el.utwente.nl.

IEEE Log Number P95012.

[27]. Often the design of this type of edge detectors boils down to finding an appropriate cost function, the minimization of which implicitly defines the response of the "ideal" noise filter. A number of authors has made claims to optimality in their solutions for the edge detection problem:

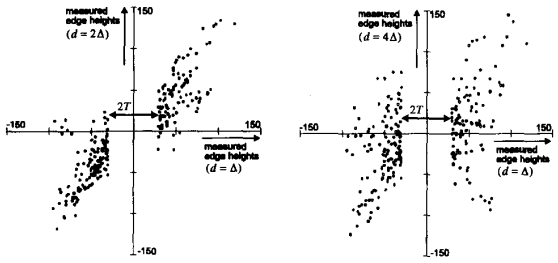
- Shanmugan et al. [22] considered a 1-D step function. Their objective was to arrive at a filter that has a response to a step function with maximum energy within a given interval in the vicinity of the edge. The filter is subjected to the constraint that it is strictly band-limited in the frequency domain, and that its response is even. Extensions to two dimensions are made by rotation of the 1-D filter transfer.
- Marr and Hildreth [18] advocated the use of the Laplacian of a 2-D Gaussian as the filter response for edge detection. The optimality follows from the argument that Gaussians are the only smoothing functions that maximize the product of spatial width and frequency bandwidth.
- Canny [6] considered a 1-D step function immersed in Gaussian noise. A filter is designed, such that its response to the step function and noise maximizes the product of the SNR (signal-to-noise ratio) measured at the position of the step, and the inverse of the root-mean-square localization error. Extensions to two dimensions are made by projection operators. A number of modifications to Canny's work are given: [24], [20], [8], [21], [28].
- Boie and Cox [4], [5] use the SNR and the inverse mean-square localization error separately to arrive at a detection-filter and a localization-filter for the detection of a 1-D step function. Generalizations to two dimensions are also based on projection operators.
- Shen and Castan [23] also considered filters for 1-D step function detection. Their optimality criterion relates only to SNR. The criterion does not quantify the localization ability.

Most authors cited above consider a single 1-D edge profile with fixed step height immersed in noise. The premise of our work is that this model is overly restrictive for two reasons. Firstly, in most images a large variability of edge heights is encountered. Therefore, random step heights with a suitably chosen probability density would be favorable. Secondly, the presence of neighboring objects in the scene ruins the model of having a *single* step function. In fact, for the imaging device most often used (i.e. the CCD-camera), the influence of neighboring objects in the scene has often more impact on the detection process than the noise of the camera (see example below). Of course, most filters mentioned above have spatially limited kernel sizes, but since the existence of the neighboring objects is not explicitly mentioned in the criteria, it is questionable whether these solutions are optimal in dealing with these interfering factors. An exception is with Shen-Castan [23] who affirm that a multi-edge model is needed. Their model is a binary random telegraph signal [19] with fixed step heights and Gaussian white noise. The authors show that their noise suppression filter minimizes the mean-square error between estimated signal and the telegraph signal. Unfortunately, the authors do not clarify why this criterion is well suited to avoid multiple detections. Our experiments (Section 3) indicate that it is not.



(a) (b)

Fig. 1(a). Grey level image. (b) Map with detected vertical edges.



(c) (d)

Fig. 1(c). and (d) Scatter diagrams of edge heights measured near the detected edges.

The assumptions stated above are illustrated in Fig. 1. A gray level image is shown together with 1(b) an accompanying map of detected edges with vertical orientations.[1] Fig. 1(c) and 1(d) show scatter diagrams derived from a random selection of detected edges. For each detected edge, the corresponding step height is estimated from the original image by calculating the difference of the gray levels at the two sides of the detected edge. This is done at a distance of one pixel period (the horizontal axis in both figures), at two pixel periods (Fig. 1(c), vertical), and at four pixel periods (Fig. 1(d), vertical). The gap in the horizontal direction has a width $2T$ corresponding to the threshold applied. Clearly, these scatter diagrams do not comply with a model of having a single step function with fixed step height. Instead, the step heights seem to obey a more or less zero-mean, normal distribution. Edges with step heights below the threshold are not detected. Therefore, the central part of the distribution is hidden. But this does not imply that these edges do not exist. In addition, the diagrams suggest that

[1]The detector used in this example is the first order symmetric difference operator $[-1\ 0\ 1]$ together with non-maximum suppression and hysteresis thresholding [6]. Of course, this detector is inferior to any other reasonable detector, but for this example this doesn't matter much.

near an edge, the signal is highly correlated, while moving away from the edge, the signal decorrelates more and more. (The correlation coefficients in Fig. 1(c) and 1(d) are 0.88 and 0.48, respectively). The reason for this is that moving away from the edge, the chance to cross another object boundary increases.

B. The Outline of This Work

The purpose of this paper is to build a feature extractor for signals with multiple edges (or lines) with varying heights. As such, this work is an extension of Canny's work and others. Since we don't want to prejudice the design towards a filter, we prefer to use the term *feature extractor* rather than *detection filter*.

The starting point in the development of the feature extractor is a model in which the occurrence of edges in 1-D signals is described in terms of conditional autocovariance functions. Application of the Bayes criterion (minimum risk) with unit cost function for both the detection and the localization of the edges results in a feature extractor, the output of which can be interpreted as a sequence of log-likelihood ratios associated with the input signal. As with Canny and others, non-maximum suppression and thresholding this sequence yields a final map with detected edges. In Section II, this feature extractor is developed.

Usually, the performance of an edge detector is expressed as the signal-to-noise ratio and the root-mean-square of the localization error. However, unlike the "single-edge case," the performance measure of a detector for multiple edges is not trivial. There are several factors which determine the quality of the detection. In Section III, an evaluation criterion is defined. Furthermore, in this section the performance of the detector is assessed and compared with the performance of edge detectors known from literature.

The 1-D edge feature extractor (Section 2) can be easily generalized to two dimensions using the projection technique of Canny [6]. This is optimal only in the case of linear edge segments, but not for curved segments or corners. Therefore, we prefer to apply the criterion function in two dimensions, and to develop the extractor in this domain instead. This is most easily done if spots are to be detected instead of edges or line segments. Section IV starts with a discussion about spot detection. Then, in this section the model is generalized to cases with chained elements such as edges. With a minor modification the design procedure for edge detection also applies to line detection. This topic is also dealt with in Section IV. The section concludes with a discussion on various classification schemes including non local maximum suppression techniques and schemes based on Markov random fields. In Section V conclusions are drawn.

II. One Dimensional Detection

A. Single Step Function

The starting point of this section is the stipulation of a model of a single edge or line profile in a 1-D observed signal $\underline{w}(x)$. The variable x denotes the space coordinate. Assume that the functional form of the edge or line profile is given by $s(x)$. For the moment, the actual shape of $s(x)$ is unimportant, but it is assumed that $s(x)$ is located at $x=\underline{\xi}$.The observed signal is given by the following expression (see Fig. 2):

$$\underline{w}(x)=\underline{m}\cdot\underline{a}\cdot s(x-\underline{\xi})+\underline{b}+\underline{n}(x) \tag{1}$$

The profile $s(x)$ occurs with prior probability P_1. This is modeled with a discrete random variable $\underline{m}$ taking values of either 0 or 1 with probability P_m. The variable $\underline{a}$ is the height of the profile. Most authors assume a fixed height, but in this paper $\underline{a}$ is a random variable. The position of the profile is given by the unknown variable $\underline{\xi}$.

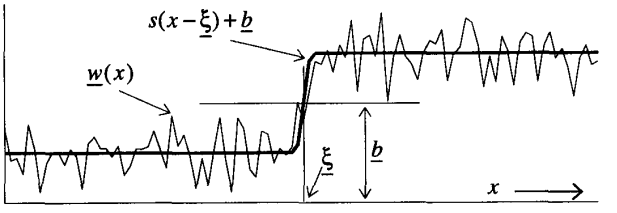


Fig. 2. A single 1-D edge profile and the observed signal.

The product of two variables $\underline{a}$ and $\underline{m}$ seems to over specify the model. However, the motivation of using this product is twofold. Firstly, $\underline{a}$ is a continuous random variable while $\underline{m}$ is discrete. If we substitute $\underline{c}=\underline{a}\,\underline{m}$, the probability distribution of this new variable becomes a mixture of a continuous and a discrete distribution. This complicates the detection problem unnecessarily. Secondly, the factorization of $\underline{c}$ into $\underline{a}\,\underline{m}$ expresses exactly what we try to model. The variable $\underline{m}$ describes the *existence* of an edge. Together with the position of the edge it is merely related to the geometry of the object being imaged. The variable $\underline{a}$ describes the *appearance* of the edge. It is merely related to the radiometry and imaging conditions of the object. It might well be the case that an edge exists (i.e. $\underline{m}=1$) while this is not visible in the image (i.e. $\underline{a}\approx 0$). Of course when $\underline{a}\approx 0$ the quality of the detection breaks down.

The observed signal is contaminated by two unknown factors: a constant (but unknown) background level $\underline{b}$ and a noise term $\underline{n}(x)$. The random variable $\underline{b}$ models the background illumination found in most imaging systems. The noise (e.g. thermal noise, reset noise) is assumed to be Gaussian, stationary and signal independent.

The problem is composed of both a detection and an estimation problem: given a realization $w(x)$ of the observed signal $\underline{w}(x)$ over a finite interval, test the hypothesis that the profile $s(x)$ is present (i.e. $\underline{m}=1$); and on assertion, estimate its position $\underline{\xi}$. A minimum risk solution can be derived by adoption of a cost function and by seeking the solution which minimizes the expectation of this cost function. A cost function $C(m, \xi|\underline{m}, \underline{\xi})$ quantifies the cost which is involved if the response of a detector/estimator is given by m and ξ, while the real (unknown) variables are $\underline{m}$ and $\underline{\xi}$. The

TABLE I.
LIST OF SYMBOLS

Symbol	Denotation	Probability Density
$\underline{a}$	Height	Zero mean, Gaussian distribution with variance σ_a^2
$\underline{\xi}$	Position of $s(x)$	Uniform distribution within a closed interval X
$\underline{b}$	Random background level of $\underline{w}(x)$	Gausian distribution with variance σ_b^2
$\underline{n}(x)$	Noise	Zero mean, stationary, Gaussian random process with autocovariance function (ACF) $R_{nn}(x) = \mathrm{E}\{\underline{n}(\tau + x)\underline{n}(\tau)\}$ and variance $\sigma_n^2 = R_{nn}(0)$

expectation of this cost, known as conditional risk, takes the form [9]:

$$\mathrm{E}\left\{C\left(m,\ \xi | \underline{m},\ \underline{\xi}\right) | \underline{w}(x) = w(x)\right\} \tag{2}$$

The detector/estimator which minimizes this risk is optimal according to Bayes.

Ideally, the cost function is chosen such that it quantifies the damage that is involved if m and ξ differ from $\underline{m}$ and $\underline{\xi}$. To achieve mathematically tractable solutions, and because it is hard to quantify the cost anyway, often simplified cost functions are introduced. In this context, a popular one is the uniform cost function: unit cost if any error occurs; zero cost if the solution is fully correct. This cost function may be a bit blunt, especially in its assignments of cost to small localization errors, but its choice enables an easy way of mathematical handling. On adoption of this function, the optimal detector/estimator must maximize the posterior probability density $f(\underline{m} = m,\ \underline{\xi} = \xi | \underline{w}(x) = w(x))$. Hence, the Bayes solution with uniform cost function is equivalent to the Maximum a Posterior (MAP) detector/estimator [12].

The probability densities of the various random variables considered in this paper are given in Table I. We assume that the background level $\underline{b}$ contains no information concerning the presence and position of an edge. Therefore, we require a constrained solution invariant to $\underline{b}$. One way to achieve that is to assume a Gaussian probability density of $\underline{b}$ with $\sigma_b >> \sigma_a$.

Application of Bayes rule for conditional probabilities reveals that the MAP solution is equivalent to the values of the parameters m and ξ which maximize:

$$f\left(\underline{w}(x) = w(x) | \underline{m} = m,\ \underline{\xi} = \xi\right) P_m f(\underline{\xi} = \xi) \tag{3}$$

Here, $f(\underline{w}(x) = w(x) | \underline{m} = m,\ \underline{\xi} = \xi)$ is the conditional probability density (also called the likelihood function) of the observed signal conditioned on the state of $\underline{m}$ and $\underline{\xi}$. Since it is assumed that the position $\underline{\xi}$ is uniformly distributed within a closed interval X, it suffices to maximize

$$f(\underline{w}(x) = w(x) | \underline{m} = m,\ \underline{\xi} = \xi) P_m$$

within that interval. Thus the MAP estimate of $\underline{\xi}$ becomes equivalent to its MLE (maximum likelihood estimate).

A closed form solution is obtained by further development of the likelihood function. To this end, we represent the observed signal $\underline{w}(x)$ by a finite, sampled record. Let $\vec{\underline{w}}$ denote an N-D random vector, the elements $\underline{w}_i$ of which are samples of the continuous signal observed in the interval $0 \le x < N\Delta$; i.e. $\underline{w}_i = \underline{w}(i\Delta)\ \ i = 0, \cdots, N-1$. The sampling period is denoted Δ. In addition, we define the signal vector $\vec{s}(\xi)$ and the noise vector $\vec{n}$ with elements $s_i(\xi) = s(i\Delta - \xi)$ and $\underline{n}_i = \underline{n}(i\Delta) + \underline{b}$, respectively. The observed vector $\vec{\underline{w}}$ consists of the signal vector and the noise vector:

$$\vec{\underline{w}} = \underline{m} \cdot \underline{a} \cdot \vec{s}(\underline{\xi}) + \vec{\underline{n}} \tag{4}$$

With fixed $\underline{\xi} = \xi$ the random vector $\underline{a} \cdot \vec{s}(\xi)$ has covariance matrix $\mathbf{R}_s(\xi)$ given by

$$\mathbf{R}_s(\xi) = \mathrm{E}\{\underline{a} \cdot \vec{s}(\xi) \cdot \underline{a} \cdot \vec{s}^t(\xi)\} = \sigma_a^2 \vec{s}(\xi) \vec{s}^t(\xi) \tag{5}$$

The elements of the covariance matrix $\mathbf{R}_n$ (see Table 1) of the noise vector become

$$\mathbf{R}_{n,ij} = \sigma_b^2 + R_{nn}((i-j)\Delta) \tag{6}$$

With $\underline{m} = m$ and $\underline{\xi} = \xi$ fixed, the vector $\vec{\underline{w}}$ is a zero mean Gaussian vector with covariance matrix:

$$\mathrm{R}_w(m,\ \xi) = m\mathrm{R}_s(\xi) + \mathrm{R}_n \tag{7}$$

The conditional probability density of the vector $\vec{\underline{w}}$ is given by

$$f\left(\vec{\underline{w}} = \vec{w} | \underline{m} = m,\ \underline{\xi} = \xi\right) = \frac{1}{2\pi^{N/2}\sqrt{|\mathrm{R}_w(m,\ \xi)|}} \exp\left(-\frac{1}{2}\vec{w}^t \mathrm{R}_w^{-1}(m,\ \xi)\vec{w}\right) \tag{8}$$

In order to find the best estimates of $\underline{m}$ and $\underline{\xi}$ we have to evaluate

$$f\left(\vec{\underline{w}} = \vec{w} | \underline{m} = m, \underline{\xi} = \xi\right)$$

for every possible combination

$$(m, \xi) \in \{0,\ 1\} \times X$$

and to select the combination which maximizes

$$f(\vec{\underline{w}} = \vec{w} | \underline{m} = m,\ \underline{\xi} = \xi) P_m .$$

Of course the estimate of ξ is meaningful only once the signal has been detected, i.e. when $m = 1$. Taking the logarithm of (8), canceling irrelevant terms and rearranging the result we conclude that the MAP estimator/detector must set ξ such that

$$\vec{w}^t \left(\mathbf{R}_w^{-1}(0,\ \xi) - \mathbf{R}_w^{-1}(1,\ \xi) \right) \vec{w} \qquad (9)$$

is maximized within X. The profile has been detected if (9) exceeds a threshold there. The derivation of equation (9) is lengthy and not very interesting. To keep the length of this paper within reasonable limits the derivation is omitted. It can be found in a technical report [15].

In equation (5) the covariance matrix $\mathbf{R}_s()$ has only one non-zero eigenvalue. A corollary of this is that the detection proedure can be simplified to the check whether the maximum of

$$\left(\vec{w}^t \mathbf{R}_n^{-1} \vec{s}(\xi) \right)^2 \qquad \xi \in X \qquad (10)$$

exceeds a suitably chosen threshold [15]. In that case, the signal has been detected.

Example. If the noise $\underline{n}(x)$ is such that the corresponding noise vector is white, the covariance matrix of noise and background is $\mathbf{R}_n = \mathbf{I} + \sigma_b^2 \mathbf{E}$ (with $\mathbf{E}$ an $N \times N$-matrix completely filled with 1). This matrix has rank N. The Fourier matrix $\mathbf{W}$ with elements $\mathbf{W}_{nm} = \exp(-\mathbf{j} 2\pi nm/N)/\sqrt{N}$ applied to $\vec{\underline{w}}$ would yield an uncorrelated noise and background vector the elements of which all - except for the first one - have unity variances. This first element is the zero-th harmonic, the variance of which is $1 + \sigma_b^2$. In the Fourier domain the operation $\mathbf{R}_n^{-1}$ in equation (10) corresponds to a spectral transfer that is unity for all harmonics except for the zero-th, where the transfer is

$$1/\sqrt{1 + \sigma_b^2} .$$

Hence, if $\sigma_b^2 >> 1$ the operation $\mathbf{R}_n^{-1}$ virtually removes the zero-th harmonic, i.e. the average term in $\vec{\underline{w}}$. This can also be seen by direct calculation of $\mathbf{R}_n^{-1}$:

$$\mathbf{R}_n^{-1} \approx \left(\mathbf{I} - \frac{1}{N} \mathbf{E} \right) \qquad \text{if} \qquad \sigma_b^2 >> 1$$

If the noise is correlated, the matrix $\mathbf{R}_n$ is still Toeplitz - see equation (5), and so is $\mathbf{R}_n^{-1}$ (provided that $N\Delta$ is large enough). Therefore, the spectral transfer of R_n^{-1} follows from Fourier transformation. Consequently, the vector $\vec{w}^t \mathbf{R}_n^{-1}$ in equation (10) corresponds well to the observed signal $\vec{w}$ filtered such that the noise is whitened. Furthermore, the inner product of $\vec{w}^t \mathbf{R}_n^{-1}$ with $\vec{s}(\xi)$ is equivalent to convolution. Hence, except for the square law transfer, the Bayes solution given in equation (10) is virtually equal to Canny's solution. However, the optimization criterion used here differs a lot from the one of Canny. Furthermore, the underlying models differ very much. In Canny (and others) the profile is deterministic, whereas in our model the profile has a random height.

This is also a good moment to compare the Bayes solution with the method of residual analysis advocated by Chen *et al* [7]. The first step in this analysis is to calculate the residuals. With the notations used in this paper, the residual $\underline{r}(x)$ can be seen as the result of a linear filter with input $\underline{w}(x)$ and impulse response $h(x) = \delta(x) - h_{low}(x)$. Here $\delta(x)$ is the Dirac function, and $h_{low}(x)$ is the impulse response of a symmetric low-pass filter. If $s(x)$ is a step function, its location is seen in the residuals as a zero-crossing. Therefore, the next step is to localize the zero-crossing in $\underline{r}(x)$. Then, in each neighborhood of a zero-crossing the autocovariance function of $\underline{r}(x)$ is estimated by

$$\hat{\underline{R}}_{rr}(x) = \int \underline{r}(\tau) \underline{r}(\tau + x) d\tau$$

A zero-crossing is marked as an edge if $\underline{R}_{rr}(0)$ exceeds a threshold, and if $\underline{R}_{rr}(x)$ meets certain conditions; e.g. $\underline{R}_{rr}(\alpha) / \underline{R}_{rr}(0)$ must be large enough (α is a positive quantity). While this procedure may be robust (e.g. insensitive to model errors) it is certainly not optimal with respect to detection. To show this, we expand the estimated ACF substi-tuting for $\underline{r}(x) = h(x) * \underline{w}(x)$ and $\underline{w}(x) = \underline{a} s(x) + \underline{n}(x) + \underline{b}$:

$$\underline{R}_{rr}(x) = \underline{a} \int \left[h(\tau) * \underline{w}(\tau) \right] \left[h(\tau + x) * s(\tau + x) \right] d\tau + \int \left[h(\tau) * \underline{w}(\tau) \right] \left[h(\tau + x) * \underline{n}(\tau + x) \right] d\tau$$

Provided that the impulse response $h(x)$ is chosen appropriately the first term evaluated for $x = 0$ could correspond well to the Bayes solution given above. However, whatever the choice of $h(x)$ is, the second term is fully random. It can only lower the detection quality.

B. Multiple Step Functions: Covariance Models

In this section, the signal and noise model will be extended to the case of multiple edges within one realization of the observed signal. Starting point in this model is a special random process, called an inhibited Poisson impulse process. Such a process consists of an infinite train of Dirac functions at locations $\underline{\xi}_k$ with intensities $\underline{a}_k$. It can be looked upon as an ordinary stationary Poisson impulse process [19] with a specific kind of distortion.

Without inhibition, the sequence $\left\{ \underline{\xi}_k \right\}$ is a Poisson distributed, stationary point process with density (= mean number of points per unit length) λ. The density is assumed to be constant

everywhere. The intensities $\underline{a}_k$ are considered to be Gaussian random variables with zero mean and variance σ_a^2. Furthermore, it is assumed that these variables are independent, i.e. $\mathrm{E}\{\underline{a}_k \underline{a}_n\} = 0 \quad n \neq k$.

The process is called inhibited if it is known that no point has a neighbor within a distance R_i (known as the inhibition distance). The introduction of inhibition serves the same purpose as the multiple response constraint of Canny: it prevents a double detection of a single edge. By assumption, the inhibition distance is small compared with the mean distance between two neighboring points: $R_i\lambda << 1$. In that case, the probability of having more than two points within an interval with length R_i is negligible, and consequently the process can be regarded globally as a Poisson impulse process in the strict sense.

The observed signal is a pile-up of profiles contaminated by noise and a random background level:

$$\underline{w}(x) = \sum_k \underline{a}_k s\left(x - \underline{\xi}_k\right) + \underline{b} + \underline{n}(x). \tag{11}$$

$\underline{\xi}_k$ is the position of the *k-th* profile, and $\underline{a}_k$ is the corresponding height. *s(x)* is the profile, as before. Fig. 3 shows a realization of the process $\underline{w}(x)$.

Fig. 3. Pile-up of edge profiles (thick line) and its noisy observation.

The problem is to estimate the positions

$$\underline{\xi}_k$$

of all random points, given the observed signal $\underline{w}(x)$, or given an infinite number of samples $\underline{w}(i\Delta)$ of this signal. The problem can be reduced to the detection/estimation problem of the previous section by partitioning the x-axis into overlapping intervals:

$$X_n = [(n-1)\Delta,\ (n+1)\Delta] \qquad n = \cdots,\ -1,\ 0,\ 1,\ \cdots.$$

If it is assumed that the length 2Δ of each interval is much smaller than the mean distance between points $2\lambda\Delta << 1$, the probability of having precisely one point in X_n equals $P_1 = 2\lambda\Delta$ and the probability of having more than one point is negligible. The problem is solved by application of the detector/estimator of the previous section to each interval X_n and by combining the results of all these detections.

In order to apply the detector/estimator of Section IIA, the availability of conditional ACFs is required. To adapt the model to the problem at hand, we have to introduce an extra term describing the interfering contribution of all the edge profiles lying outside the interval X_n. Let us assume that it is known that no point $\underline{\xi}_k$ exists within X_n. Then the conditional autocovariance function

$$R_{ww}\left(x_1,\ x_2 | \underline{\xi}_k \notin X_n\right)$$

is composed of contributions due to the noise, the background level, and a contribution $R_{pp}(.)$ representing the profiles with positions outside X_n:

$$\begin{aligned} &R_{ww}\left(x_1, x_2 | \underline{\xi}_k \notin X_n\right) = \\ &R_{nn}(x_1 - x_2) + \sigma_b^2 + R_{pp}\left(x_1, x_2 | \underline{\xi}_k \notin X_n\right) \end{aligned} \tag{12}$$

If, on the other hand, it is known that a point, say $\underline{\xi}_k$, exists within X_k, then we have four contributions to the conditional ACF: the noise $R_{nn}(.)$, the background level, the contribution $R_{ss}(.)$ of the single point, and the remainder $R_{pp}(.)$ of the Poisson process:

$$\begin{aligned} &R_{ww}\left(x_1, x_2 | \underline{\xi}_k \in X_n\right) = R_{nn}(x_1 - x_2) + \sigma_b^2 + \\ &R_{ss}\left(x_1, x_2 | \underline{\xi}_k \in X_n\right) + R_{pp}\left(x_1, x_2 | \underline{\xi}_k \in X_n\right). \end{aligned} \tag{13}$$

Expressions for $R_{pp}(\ )$ and $R_{ss}(\ )$ are given in Appendix A.

C. Covariance Model Operators

The purpose of this section is to find a detector/estimator which detects the existence of an edge or line profile within the interval X_n, and (in case of a detection) estimates its position within X_n. The detector has an infinite, countable number of samples $\underline{w}(i\Delta)$ at its disposal. We assume that samples taken from a position far away from X_n, i.e. $\lambda|i - n|\Delta >> 1$, do not contain information concerning the existence of profiles in X_n. Therefore, it is safe to restrict the samples which are actually used to a finite set. Let this set be arranged in a vector $\underline{\vec{w}}_n$ with elements

$$\underline{\vec{w}}_{n_i} = \underline{w}\left((n + i - (N-1)/2)\Delta\right),\ i = 0,1,\cdots,N-1.$$

It is presumed that N is chosen odd.

Since the statistical definition for each subinterval X_n is position invariant, it suffices to restrict the discussion of the detector/estimates to the one of X_0 without loss of generality. Like in Section IIA let m and ξ be the MAP estimators of the existence $\underline{m} \in \{0,1\}$ and the position $\underline{\varsigma} \in X_0$ of a profile within X_0. Following the discussion in Section 2A the MAP detector will detect an edge within X_0 if

$$v_0(\xi) = \vec{w}_0^t\left(\mathrm{R}_w^{-1}(0,\xi) - \mathrm{R}_w^{-1}(1,\xi)\right)\vec{w}_0 \tag{14}$$

exceeds a suitably chosen threshold. Here, $\mathbf{R}_w(m, \xi)$ is the covariance matrix of the vector $\vec{w}_0$, conditioned on the state of $\underline{m} = m$ and $\underline{\xi}_k = \xi$ with $\underline{\xi}_k \in X_0$. These matrices follow from (12) and (13):

$$\mathbf{R}_{w_{ij}}(0, \xi) = R_{ww}\left((i - N/2 - 1/2)\Delta, (j - N/2 - 1/2)\Delta | \underline{\xi}_k \notin X_0\right) \quad (15)$$

$$\mathbf{R}_{w_{ij}}(1, \xi) = R_{ww}\left((i - N/2 - 1/2)\Delta, (j - N/2 - 1/2)\Delta | \underline{\xi}_k \in X_0\right) \quad (16)$$

The signal $v_0(\xi)$ defined in (14) is called the log-likelihood ratio of the interval X_0. The estimated position ξ of a hypothesized profile within X_0 is the argument which maximizes $v_0(\xi)$ within X_0. The search for this maximum can be kept quite simple. If subpixel accuracy is needed it is necessary to evaluate $v_0(\xi)$ for some discrete values of ξ within X_0 and to locate the maximum by an appropriate interpolation. If subpixel accuracy is not needed it suffices to check whether $v_0(0) > v_0(\Delta)$ and $v_0(0) \geq v_0(-\Delta)$. With $\xi = \Delta$, the likelihood ratio given the vector $\vec{w}_0$ must be close to the likelihood ratio given the vector $\vec{w}_1$. That is, $v_0(\Delta) \approx v_1(0)$. The reason for this is that the vectors $\vec{w}_0$ and $\vec{w}_1$ differ only in their first and last element. Likewise $v_0(-\Delta) \approx v_{-1}(0)$. Therefore, the procedure given above is close to checking to see whether:

$$v_0(0) > v_1(0) \text{ and } v_0(0) \geq v_{-1}(0) \text{ and } v_0(0) > \textit{threshold} \quad (17)$$

The detection of all profiles is performed by application of the detector to each interval X_n. Hence, if for the sake of brevity $v_n(0)$ is denoted by v_n, it suffices to evaluate v_n for all n to locate all local maximums in the sequence v_n and to compare these local maximums with a suitably chosen threshold. A profile is detected whenever a local maximum exceeds this threshold.

It is conceivable that the information-carrying part of the observed vector $\underline{\vec{w}}_n$ is kept in a small (linear) subspace of the N-dimensional observation space only. If this is the case, linear feature extraction may be applied in order to reduce the dimensionality of the space, and to supply features that are more useful than the original data. Such a linear feature extraction consists of a multiplication of the observed vector $\underline{\vec{w}}_n$ by a $D \times N$-matrix $\mathbf{T}$. Principal component analysis [9] gives us the tool to determine the dimension D and to design the matrix $\mathbf{T}$. The first step in this analysis is to determine a $N \times N$-matrix such that when applied to the vector $\underline{\vec{w}}_n$ the resulting covariance matrices $\mathrm{R}_w(0,0)$ and $\mathrm{R}_w(1,0)$ become the unity $N \times N$-matrix $\mathbf{I}$ and a diagonal $N \times N$-matrix Γ, respectively:

$$\vec{u}_n = \mathbf{T}\vec{w}_n \qquad v_n = \vec{u}_n^t\left(\mathbf{I} - \Gamma^{-1}\right)u_n \quad (18)$$

The effect of the operation $\vec{u}_n = \mathbf{T}\vec{w}_n$ is that each element in $\vec{u}_n$ independently contributes to v_n. Suppose that γ_i is the i-th diagonal element in γ, then the effectiveness of each element in $\vec{u}_n$ can be expressed by the Bhattacharyya distance of the underlying conditional probability densities:

$$J_b(N) = \sum_{i=0}^{N-1} J_i \qquad J_i = \frac{1}{2}\log\frac{1}{2}\left(\sqrt{\gamma_i} + 1/\sqrt{\gamma_i}\right) \quad (19)$$

Now, features in $\vec{u}_n$ can be selected by disregarding all elements in $\vec{u}_n$, of which the contributions to $J_b(N)$ are negligible. Therefore, we sort the rows in $\mathbf{T}$ such that $J_{i-1} \geq J_i$, and maintain only the first D rows, where D is chosen as small as possible, but such that $J_b(D)$ still approximates $J_b(N)$. As a result, $\mathbf{T}$ will become a $D \times N$-matrix, $\mathbf{I}$ and Γ will reduce to $D \times D$, matrices and the vector $\vec{u}_n$ will be D-dimensional.

The transformation $\vec{u}_n = \mathbf{T}\vec{w}_n$ with $n = \cdots, -1, 0, 1, \cdots$ corresponds to a parallel bank of D digital FIR filters. The input of each filter is the sequence $w(n\Delta)$, the output of the *i-th* filter is the sequence u_{n_i}. The impulse response of the *i-th* filter is given in the *i-th* row in $\mathbf{T}$. According to (18), the output of each filter is squared, multiplied by a weight $c_i = 1 - 1/\gamma_i$, and finally summed together to yield the log-likelihood-ratio v_n. Non-local maximum suppression and thresholding complete the detection process. This is shown in Fig. 4.

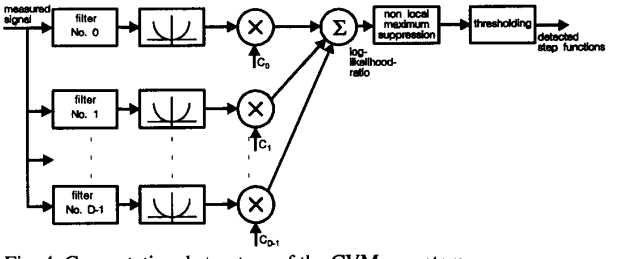


Fig. 4. Computational structure of the CVM operators.

It is difficult to give analytical expressions of the impulse responses. In our experiments, these responses are obtained numerically. The procedure to get the responses and the corresponding weights is summarized in Appendix B.

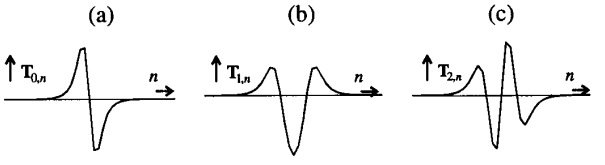


Fig. 5 (a). Impulse response of the first filter of a cvm-operator.
(b). Impulse response of the second filter of a cvm-operator.
(c). Impulse response of the third filter of a cvm-operator.

An illustration is given in Figs. 5 and 6. The Poisson process in Fig. 3 is given by: $\lambda\Delta = 0.06$ and $R_i = 4\Delta$. The profile is an erf-function, the derivative of which has width $\sigma_p = 0.5\Delta$. The noise is assumed to be white with standard deviation $\sigma_n = 0.2\sigma_a$. The chosen operator size of the filters is $N = 39$. In this example the number of filters can be reduced

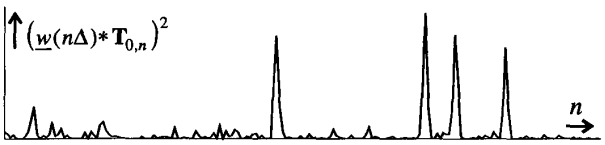


Fig. 6(a). Squared output of first CVM-filter.

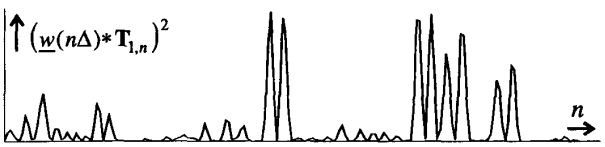


Fig. 6(b). Squared output of second CVM-filter.

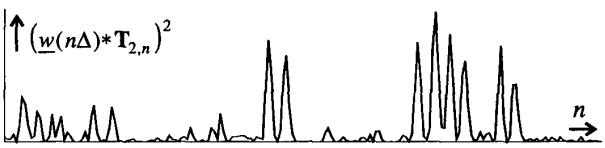


Fig. 6(c). Squared output of third CVM-filter.

Fig. 6(d). Log-likelihood ratios obtained by linear combinations of (a), (b) and (c). Step functions are detected at positions of local maximums exceeding a suitably chosen threshold

Fig. 6. Application of a CVM operator to the signal in Fig. 3.

to $D = 3$ without loosing too much information. The resulting filter impulse responses and corresponding (squared) outputs are given in Figs. 5 and 6, respectively. These outputs are linearly combined with weights $c_0 = +0.37$, $c_1 = -0.28$, $c_2 = -0.19$, respectively. This yields log-likelihood ratios (Fig. 6(d)), the local maximums of which indicate the positions of possible step functions. Non local maximum supression and thresholding completes the detection process. Note that if $R_i = 0$, the number D of useful filters reduces to one. In that case, the feature extractor essentially resembles the one of Canny and others.

The impulse responses of the given example can be given the following interpretation. The impulse response of the first filter is such that the first derivative of the signal is estimated. This is quite similar to the detection filters of other authors, (e.g. Canny, Boie-Cox, etc.). The weight of this filter is positive. However, the two other filters have negative weights, and correspondingly, their squared (and thus positive) outputs are subtracted from the squared output of the first filter. Moreover, the responses of the last two filters are zero at the position of a (step edge) profile. Hence, these filters serve to sharpen the peaks in the log-likelihood ratios without affecting the maximums of these peaks. Therefore, these filters help to localize the step functions. Furthermore, it can be seen that the last two filters also improve the *SNR* in areas in which the signal does not show a transition.

At first glance, the second and third filters look like the second and third derivative of a Gaussian ($\partial^2\text{gauss}(x)/\partial x^2$ and $\partial^3\text{gauss}(x)/\partial x^3$), but in fact they may be quite different. In the example given this holds true especially for the third filter. Close examination of Fig. 6(c) reveals that at the position of a step edge the output of this filter is zero, while the extremes are found in the side lobs of the step response. As explained above the corresponding weight is negative, and thus the side lobes help to localize the edge. If the impulse response of the third filter would be similar to $\partial^3\text{gauss}(x)/\partial x^3$, its step response would be close to $\partial^2\text{gauss}(x)/\partial x^2$. In that case the corresponding weight should have been positive, because this response finds its extreme at the position of the edge. The noise at the output of the third filter is independent to the noise at the outputs of the other filters: the information given by the third filter is extra.

The detector/estimator developed in this section is named CVM-detector, since its existence is based on a covariance model of signal and noise.

III. Performance Analysis

A. Evaluation Criterion

The objective of this section is to arrive at a quantitative evaluation measure which enables us to experimentally test the various detection schemes. The performances of the detectors will be assessed using artificially generated test signals with accompanying reference maps (showing the true positions of the profiles) and by defining an evaluation criterion which measures the distance between reference map and the detections delivered by the detector-under-test.

The quality of a detector of single events can be quantified by its error rate E. This is the probability of a detection error, either a missed event or a false alarm. In Canny's model the signal-to-noise ratio *SNR* decreases monotonically with E. An important performance measure for estimators is the root-mean-square error, defined as

$$RMS = \sqrt{E\left\{\left(\underline{\xi} - \underline{\hat{\xi}}\right)^2\right\}}$$

with $\underline{\hat{\xi}}$ the estimate of the random variable $\underline{\xi}$. Canny uses the reciprocal $LOC = 1/RMS$ as a quality criterion for the localization ability.

If a detector has to find many profiles in a signal, the *SNR* as defined by Canny does not fully define the quality of this detector. *SNR* is defined for one and only one position. False alarms, however, occur at local maximums of the filtered signal. These maximums may occur anywhere. Therefore, the probability P_{fa} of false alarms should be related to the length of an interval (a more consistent measure to assess the quality is the density of false alarms). Canny tries to handle this aspect by formulating his multiple response constraint, stating that the mean distance $\bar{x}$ between neighboring local maximums of the filtered noise must be set to some constant.

The performance measure of Canny is the quantity:

TABLE II.
PERFORMANCE OF VARIOUS OPERATORS

Parameters of Test Signal: $\breve{\lambda} = 0.06[1/\Delta]$, $\breve{R}_i = 2[\Delta]$ $\breve{\sigma}_p = 0.5[\Delta]$

Operator	Parameter(s) of Operator	AVR	$\lambda(\omega_4)$ Missed Edges	$\lambda(\omega_3)$ False Alarms	$\lambda(\omega_2)$ Multiple Detections	$\lambda(\omega_{1,\Delta})$ Loc. Errors	$\lambda(\omega_{1,2\Delta})$ Loc. Errors	$\lambda(\omega_{1,3\Delta})$ Loc. Errors
CVM	$\lambda = 0.06[1/\Delta]$ $\sigma_p = 0.5$ [Δ] $\sigma_n = 0.4[\sigma_a]$ $R_i = 3$ [Δ]	0.0228	0.0185	0.0016	0.0002	0.0066	0.0006	0.0002
Sarkar-Boyer	$\phi = 1.0$ $\beta = 0.9$ $\alpha = 0.7$ [1/Δ]	0.0230	0.0187	0.0015	0.0002	0.0071	0.0005	0.0001
Boie-Cox	$\sigma = 1.6$ [Δ]	0.0233	0.0190	0.0015	0.0001	0.0077	0.0005	0.0001
Canny	$k = 1.0$ $2W = 11$ [Δ]	0.0235	0.0186	0.0014	0.0001	0.0088	0.0008	0.0002
Petrou-Kittler	$2w = 11$ [Δ] $s \to \infty$ [1/Δ]	0.0235	0.0188	0.0016	0.0001	0.0084	0.0006	0.0001
Spacek	$2W = 9$ [Δ]	0.0236	0.0186	0.0012	0.0000	0.0098	0.0009	0.0002
Shen-Castan	$\alpha = 0.40$ [1/Δ]	0.0242	0.0190	0.0016	0.0011	0.0063	0.0006	0.0002

$SNR \times LOC$. Used as an evaluation criterion in an experimental set–up, the objections against this measure are at least twofold:

- *SNR* does not involve the density of false alarms (see the discussion above).
- The relationship between $SNR \times LOC$ and the cost that is involved by an erroneous detection is unclear. For instance, the unit of $SNR \times LOC$ is distance^{-1}. The use of this, as the unit of a quality measure, is doubtful.

Spacek [24] and others [20] use $SNR \times LOC \times \bar{x}$ as a performance measure. However, this quantity suffers from the same defects.

We will develop an evaluation criterion which is based on the average risk. As stated earlier, a detector is prone to several types of errors. We distinguish the following types:

$\omega_{1,\varepsilon}$: A localization error over a distance ε

ω_2: A multiple detection. A single profile has given rise to an extra detected edge.

ω_3: A falsely detected profile.

ω_4: A missed profile.

By definition a localization error is distinguished from a falsely detected profile together with a missed profile if the separation distance between these profiles is less than 3.5Δ.

Suppose that the function $C(\omega_i)$ quantifies the cost of an error of type ω_i, and that these costs add up independently. Then, a useful measure to evaluate a detector is the expectation of the cost per unit length [25], [13]. I.e., if the output of a detector contains a density $\lambda(\omega_i)$ of errors of type ω_i then the expected cost per unit length is given by:

$$AVR = \int_{\varepsilon} \lambda(\omega_{1,\varepsilon}) C(\omega_{1,\varepsilon}) d\varepsilon + \sum_{i=2,\,3,\,4} \lambda(\omega_i) C(\omega_i). \qquad (20)$$

The abbreviation *AVR* stands for average risk. In our experiments, this error measure is calculated by estimation of the densities $\lambda(\omega_i)$ and by numerical evaluation of (20); see also [14]. For that purpose realizations of the test signal $\underline{w}(x)$ (equation 11) are used. These test signals are (artificially) generated with $s(x)$ an erf-function with standard deviation $\breve{\sigma}_p$. The variance of the height σ_a^2 is kept constant. Δ is the sampling period. The following parameters are used:

$\breve{\lambda}$ (density of profiles; unit = [1/Δ]).

$\breve{R}_i$ (inhibition distance; unit = [Δ]).

$\breve{\sigma}_b$ (standard deviation of white noise; unit = [σ_a]).

$\breve{\sigma}_p$ (standard deviation of erf-function; unit = [Δ]).

In our experiments, the *AVR* uses the following settings: $C(\omega_{1,\varepsilon})$=0.3, 0.7 and 1.4 for $\varepsilon = \Delta$, $\varepsilon = 2\Delta$ and $\varepsilon = 3\Delta$, respectively. $C(\omega_i) = 1.0$ for $i = 2$, 3, and 4. The density of the types of the errors are estimated by classification of the detected profiles into the categories ω_i. For that purpose a reference map showing the true positions of the profiles is used. The estimates are based on realizations in an interval with 2×10^6 samples.

B. Experiment and Results

The experiment presented in this section is designed to verify the behavior of the newly defined *CVM* detectors and to compare them with the detectors known from literature. For that purpose six operators are selected from literature. All of these operators are claimed to be optimal. However, as stated earlier, the criteria used differ from ours. Furthermore, all these operators involve a parametric detection filter, the output of which must be squared, non-local maximums must be suppressed and, finally, the local maximums must be thresholded.

The set of operators included in the experiment are given in Table II. The Canny operator optimizes the product $SNR \times LOC$ under the constraint that the spatial width of the detection filter equals 2W and the mean distance $\bar{x}$ between neighboring local maximums is set to some fraction k of W. The Sarkar-Boyer operator [21] uses the same criteria. How-

ever, here the spatial width is not limited. This enables an IIR-realization of the filter. Sarkar-Boyer advocate the use of two causal, second order IIR filters, described by three parameters: two shape parameters ϕ and β and a scaling-parameter α. The Deriche operator [8] is contained in the Sarkar-Boyer operator. Therefore, there is no need to include the Deriche operator in the evaluations. The Boie-Cox filter [4], [5] for step edges is the first derivative of a Gaussian with width σ. This filter has also been claimed to be optimal (on various grounds) by other authors [18], [17]. Spacek's modification of the Canny criterion (i.e. $SNR \times LOC \times k$) applied to an ideal step function leads to a filter of which only the scale (parametrised by the spatial width 2W) is tunable. Petrou-Kittler [20] have extended this work to a certain class of non-ideal step functions, the steepness of which is parametrized by a variable s. Shen-Castan [23], finally, also define an optimization criterion. The resulting filter is parametrized by a scaling parameter α solely.

A test signal with known statistical parameters is generated and applied to the various operators. For each operator the filter parameters (and threshold) are adjusted so as to minimize their measured *AVR*. The results (optimized parameters, measured *AVR*, and corresponding error densities) for a typical setting of the parameters of the test signal are given in Table II.

C: Discussion

In Table II, the *CVM*-operator appears to have the best *AVR* amongst all other operators. This is to be expected since the *CVM*-model incorporates multiple edge profiles, whereas other operators are built on simple single edge models. Furthermore, amongst all optimization criteria the one of the *CVM*-operator (i.e. uniform cost) is closest to the evaluation criterion. The sole difference between the *AVR* evaluation criterion and the *CVM* (design) criterion is in the assignment of cost to localization errors. In the *CVM*-criterion, all localization errors are punished equally. Therefore, the optimal parameters of the *CVM*-operator in Table II differ a little from the corresponding parameters of the test signal. This holds especially for the noise parameter $\left(\sigma_n^2 \text{ is two times } \breve{\sigma}_n^2\right)$. Such enlargement favors the detection ability above the localization ability.

From all operators based on Canny's criterion (or variants) the one with highest degree of freedom (no FIR constraint) is most competitive to *CVM*. The Sarkar-Boyer filter performs better than the Boie-Cox (Gauss) filter. Both are IIR filters, but the Sarkar-Boyer filter has three parameters to tune, while the Gaussian has only one.

Multiple responses appear to be a problem in none of the operators of the Canny family. Therefore, optimization of the multiple response criterion k, as advocated by Spacek and Petrou-Kittler, does not give a noticeable improvement to the original Canny filter.

It is interesting to compare the Shen-Castan operator with the *CVM*-operator. The Shen-Castan filter response resembles the response of the first *CVM*-filter (Fig. 5). Both filter responses have a discontinuity at $x = 0$. This results in a very good localization ability (the *LOC*-criterion of Canny tends to go to infinity). On the other hand the detection ability of the Shen-Castan filter is modest. In addition, the filter produces many multiple responses. In the *CVM*-operator, however, these defects are compensated by the contributions of the other filters, while in the Shen-Castan operator the overall performance is poor due to the bad detection ability and the many multiple responses.

We finally note that the last column of Table 2 shows that none of the tested operators show a localization error significantly larger than 2Δ.

IV. Two Dimensional Extensions

The usual extension from 1-D edge and line operators to two dimensions is the application of a projection operator along an edge segment and a 1-D edge detector perpendicular to the edge segment. This procedure can also be applied to the *CVM*-operators. However, we prefer to proceed in another way. In this section, the 1-D model will be extended to two dimensions. Next, from this model, 2-D edge and line operators will be derived.

The signal and noise model from Section II can easily be extended so as to describe the existence of spots in 2-D imagery. Spots are points in the image plane which correspond to objects in the scene which are so small that they are represented in the image by the PSF (point spread function) of the imaging device. Let (x, y) be the coordinates of the image plane, and let $s(x, y)$ denote the PSF. Then in our model the (non-sampled) image is given by

$$\underline{w}(x,y) = \sum_k \underline{a}_k s(x - \underline{\xi}_k, y - \underline{\eta}_k) + \underline{b} + \underline{n}(x,y) \qquad (21)$$

The random process $\Sigma \underline{a}_k s(x - \underline{\xi}_k, y - \underline{\eta}_k)$ represents the signal. The position of each spot is given by the 2-D Poisson point process $(\underline{\xi}_k, \underline{\eta}_k)$ The height of each spot is given by the random variables $\underline{a}_k$. The process is statistically described by its density (λ = mean number of points per unit area) and its inhibition distance R_i. The image is corrupted by Gaussian distributed noise consisting of a constant background level $\underline{b}$ and a fluctuating part $\underline{n}(x, y)$. Furthermore, the image is received as a sampled record on an orthogonal grid with sampling periods Δ in both directions.

The problem of estimating the positions $(\underline{\xi}_k, \underline{\eta}_k)$ of all spots is solved by a straightforward generalization of the detector/estimator from Section II. The first step is to partition the image plane into overlapping subareas centered at positions $(n\Delta, m\Delta)$. Each subarea comprises its center pixel and its eight neighbor pixels. Then the assumption that at most one spot is generated within such a subarea ($4\lambda\Delta^2 << 1$) allows us to consider the remaining part of the Poisson process as interfering noise described by its ACF. The presence and the absence, respectively, of a spot within the subarea is statistically described by two conditional ACFs. These ACFs follow from a straightforward generalization of equation (12) and (13). The 1-D independent variables, x_1, x_2, etc. must be replaced by their 2-D analogues. Line integration must be replaced by area integration.

The next step is to construct a measurement vector $\vec{w}_{nm}$ of dimension N for each subarea at position $(n\Delta, m\Delta)$. The vector comprises all samples within an $M \times M$ neighborhood centeredat$(n\Delta, m\Delta)$. Elements of the vector $\vec{w}_{nm}$ are defined by:

$$\begin{aligned} \underline{w}_{nm,\,i} &= \underline{w}(n\Delta + x_i,\ m\Delta + y_i) \qquad i = 0,\ \cdots,\ N-1 \\ x_i &= (i/M - M/2)\Delta \qquad N = M \times M \\ y_i &= (i\%M - M/2)\Delta \end{aligned} \tag{22}$$

Here, "%" denotes the modulus operator, and "/" is integer division.

Once the image samples are arranged in a vector the corresponding conditional covariance matrices can be derived immediately from the conditional ACFs. The development of the desired detector follows the same course as in Section 2. Analogous to equation (18) the following statistic is (near) sufficient to our detection problem:

$$\vec{u}_{nm} = \mathbf{T}\vec{w}_{nm} \qquad v_{nm} = \vec{u}_{nm}^{t}(\mathbf{I} - \Gamma^{-1})\vec{u}_{nm} \tag{23}$$

Here too, the transformation matrix $\mathbf{T}$ and the diagonal matrix Γ result from a principal component analysis of the conditional covariance matrices. See Appendix B.

The detection process consists of evaluating the 2-D sequence:

$$v_{nm}\ n = \cdots,\ -1,\ 0,\ 1,\ \cdots \qquad m = \cdots, -1, 0, 1, \cdots,$$

finding its local maximums and checking to see whether these maximums exceed a threshold. This is a straightforward 2-D generalization of the process shown in Fig. 4.

A. Rotational Invariant Edge Feature Extraction

In this subsection, we discuss the detection of object boundaries which are characterized in the image by steplike discontinuities. The generalization from 1-D step function detection to 2-D edge detection is difficult for two reasons. The first complication is that, in two dimensions, edges occur with a certain (generally unknown) orientation, i.e. the tangential direction of the boundary of the imaged object. Hence, besides the edge height $\underline{a}$ and the background level $\underline{b}$, a third nuisance parameter must be faced: the edge orientation $\underline{\varphi}$.

The second problem is that edge elements are non-isolated points. The configurations of edge elements are restricted by two properties: *thinness* and *continuation*. Thinness requires an absence of side neighbors across an edge segment (except for some branch points) and connectedness of edge elements along the edge segment. Continuation requires consistency in the direction of a set of neighboring edge elements. This last requirement is based on the observation that the permissible raggedness of an edge segment is limited by the resolving power of the imaging device. In general, neighboring edge elements will have heights and orientations which are highly correlated.

These complications are very difficult to model and to analyze mathematically. However, the *CVM*-model can be adapted such that the complications are partly dealt with. For that purpose, the ESF (edge spread function) will be used. The ESF $s_e(x, y, \varphi)$ is defined as the response of the imaging device on a dark and a light half plane, separated by an abrupt transition. This transition is aligned on a straight line crossing the origin at an angle φ.

Suppose that it is known that a boundary of an imaged object passes the origin at an orientation φ. The property of continuation demands that in the close surroundings of the origin, the image data is given by $\underline{a} \cdot s_e(x, y, \varphi) + \underline{b} + \underline{n}(x, y)$. The farther we move from the origin, the more uncertain we are concerning the exact course of the boundary. In order to express this we introduce a *forgiving* function $p(x, y, \varphi)$. This function must be unity for $(x, y)=(0, 0)$ and must decay smoothly to zero elsewhere. Ideally, the actual choice of the function is based on the statistical behavior of the curvature of the boundary. A first guess, used in this paper, is a 2-D Gaussian function with width σ_t in the direction of the boundary and width σ_r in the orthogonal direction:

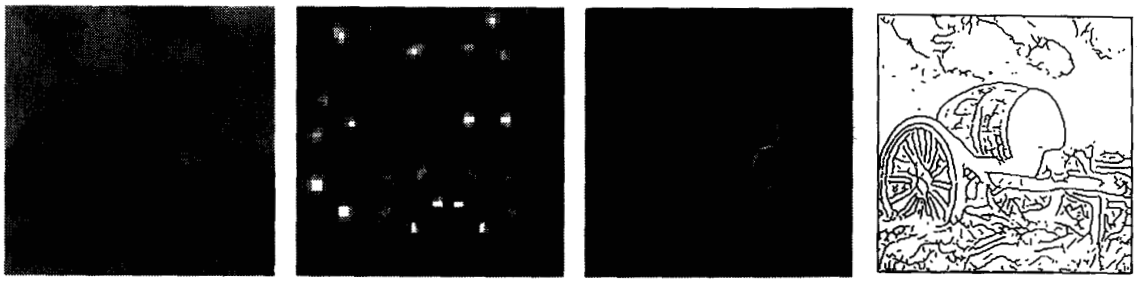
$$p(x,y) = \exp\left(\frac{-(x\cos\varphi + y\sin\varphi)^2}{2\sigma_t^2} + \frac{-(y\cos\varphi - x\sin\varphi)^2}{2\sigma_r^2}\right) \tag{24}$$

The *CVM*-model becomes:

$$\begin{aligned} &\underline{w}(x,y) = \\ &\sum_k \underline{a}_k s(x - \underline{\xi}_k,\ y - \underline{\eta}_k,\ \underline{\varphi}_k) + \underline{b} + \underline{n}(x, y) \\ &s(x, y, \varphi) = s_e(x, y, \underline{\varphi}_k)p(x, y, \varphi) \end{aligned} \tag{25}$$

The random sequence $\{(\underline{\xi}_k,\ \underline{\eta}_k)\}$ $k = 0.1, \cdots$ is assumed to be an inhibited Poisson point process with density λ and inhibition distance R_i. The variables $\{\underline{a}_k\}$ are independent Gaussian random variables. The $\underline{\varphi}_k$s give the orientations of the edge elements. They are assumed to be independent and uniformly distributed within $[0, 2\pi]$.

Some of the above stated assumptions are crude. The connectedness of edge segments implies that, knowing an edge element exists at $(\underline{\xi}_k, \underline{\eta}_k)$ a neighboring edge element must exist with about the same parameters. Therefore, it would be more precise to assume some dependency in the sequences $\{\underline{a}_k\}$ and $\{\underline{\varphi}_k\}$. Another inaccuracy is in the modeling of inhibition distance. From the results in Sections 2 and 3 it is clear that the concept of inhibition is useful, since it helps to prevent multiple responses across the edge segment. However, chosen dependency of the inhibition distance on the orientation. Such a refinement involves a too complex mathematical description, not easy to handle, and is therefore left from consideration. Of course this is at the cost of full optimality. Autocovariance functions can be used to give a statistical description of the edge model. However, the ACFs in equation (12) and (13) have to be adapted, since we have to cope



(a) (b) (c) (d)

Fig. 7(a). Blurred image. (b) Rotational invariant CVM-operator. The figure shows 9 filter impulse responses. The weight factor c_i for each filter is depicted as the length of a vertical bar right to the filter. White bars correspond to positive weights; black bars to negative. (c) Resulting log-likelihood ratio. (d) Edge map derived from (d).

with the (random) orientation of each edge element. Expressions for the conditional ACFs are given in Appendix C. With this second order moment description of signal and noise the design procedure of a feature extractor follows exactly the one given in Section 2. It must be noted, though, that the given statistical description is now incomplete. In the 1-D case, the random process $\underline{w}(x,y)$ is Gaussian, and therefore entirely determined by its second order statistics, i.e. its ACF. The introduction of random edge orientations, however, induces a non-Gaussian random process. A complete determination of the process involves higher order statistics. Therefore the given feature extractor looses optimality.

As an example, Fig. 7(a) shows an image blurred by a Gaussian PSF with $\sigma_p = 2\Delta$. The image is contaminated by white noise with deviation $\sigma_n = 5$. This is about 0.15 times the standard deviation of the gray levels of the image. Fig. 7(b) depicts a *CVM*-operator for the rotational invariant extraction of edge features in the blurred image. Its parameters are $\sigma_p = 2\Delta$, $\sigma_n = 0.15\sigma_n$, $\lambda = 0.0015/\Delta^2$, $R_i = 3\Delta$, $\sigma_r = \sigma_t = 3.5\Delta$ and M=13. Application of these parameters to the covariance model yields a set of nine filters. The filter impulse responses are similar to — but not exactly equal to — the generic neighborhood operations described in [17]. The filters combine differential operations ($\partial/\partial x$, $\partial^2/\partial x\partial y$, $\partial^3/\partial x^2\partial y$, etc. and rotated versions) with noise suppression. A merit of the covariance model is that it supplies these filters with a computational structure that tells us how to combine the outputs of these filters. For instance, the first four filters are differential operators of odd order. These filters contribute positively to the log-likelihood ratio. The next three filters are of even order and yield negative contributions. Fig. 7(c) shows the log-likelihood ratio resulting from Fig. 7(a) using the CVM-operator of Fig. 7(b).

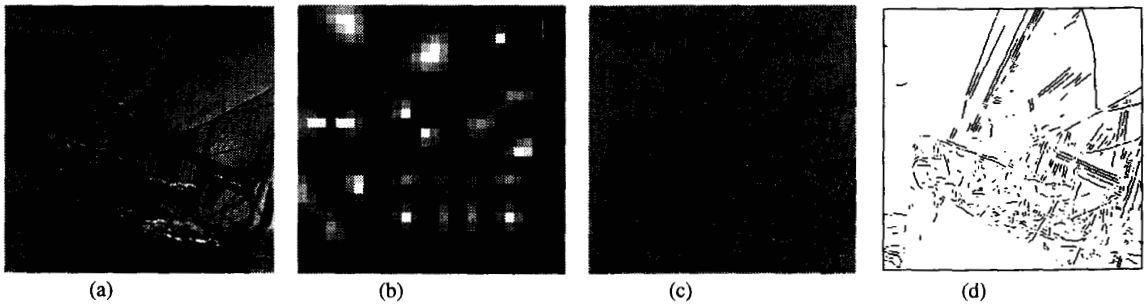
B. Rotational Invariant Line Geature Extraction

By definition, we call an elongated object "line-like" if from its image its width cannot be measured accurately. This occurs when the width of an object falls down below the spatial resolution of the imaging device.

A line-like object is reflected in the image by the LSF (line spread function) $s_l(x, y, \varphi)$. This function is the response of the imaging device on a dark background with a bright,straight line, crossing the origin at an angle φ. As in the case of edges we introduce a forgiving function $p(x,y,\varphi)$, but this time only working in the direction along the line segment:

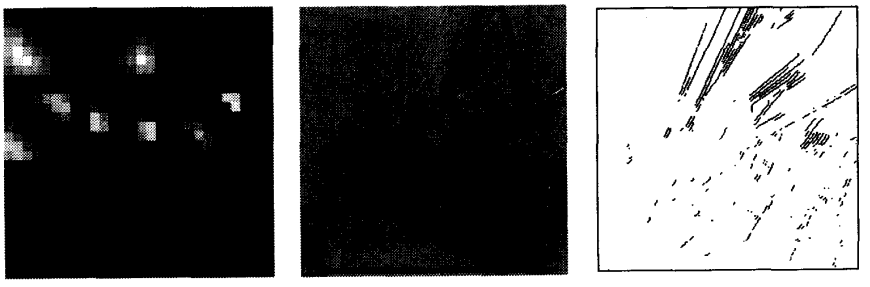
$$p(x, y) = \exp\left(\frac{-(x\cos\varphi + y\sin\varphi)^2}{2\sigma_t^2}\right) \tag{26}$$

Under the same assumptions as in Section IVA the design procedure of a line feature extractor matches the one for edge features.



(a) (b) (c) (d)

Figure 8(a). Image of an Yacht. (b) Rotational invariant CVM-operator. The figure shows 9 filter impulse responses. See text Fig. 7. (c) Resulting log-likelihood ratio. (d) Line map derived from (c).



(a) (b) (c)

Figure 9(a). Directional cvm-operator for line elements. (b) Corresponding log-likelihood ratio. (c) Line map derived from (b).

Fig. 8 shows a *CVM*-operator suitable for rotationally invariant extraction of line segments. The operator shown is based on a Gaussian LSF with width $\sigma_p = 0.7\Delta$, $\sigma_n = 1.3\sigma_a$, $\lambda = 0.05/\Delta^2$, $R_i = 3\Delta$, $\sigma_t = 4\Delta$, and $M = 7$. Here too, the filters found are similar to the generic neighborhood operators of Koenderink and van Doorn [17]. Operators of odd order are inhibitory, thereby suppressing gradients caused by, for instance, step functions or the sides of line segments. As expected, operators of even order give positive contributions to line segments.

C. Directional Feature Extraction

The covariance model can also be exploited if one is interested in the detection of line or edge segments with orientations within a certain range of angles. Suppose that this range is denoted Φ, while the complementary range (all angles not within Φ) is given by $\overline{\Phi}$. Then a line or edge element with (unknown) orientation $\varphi \in \Phi$ but with known position $(\xi,\eta) = \vec{\xi}$ will give rise to an ACF $R_{ss}(\vec{x}_1,\vec{x}_2|\vec{\xi})$ given by (see equation 5):

$$R_{ss}(\vec{x}_1,\vec{x}_2|\vec{\xi}) = \frac{\sigma_a^2}{\|\Phi\|}\int_{\varphi\in\Phi} s(\vec{x}_1-\vec{\xi},\varphi)s(\vec{x}_2-\vec{\xi},\varphi)d\varphi \qquad (27)$$

In Appendix C, similar expressions are given for the ACF of 2-D Poisson processes. With these ACFs the conditional autocovariance functions $R_{ww}(.,.|.)$ of equation (12) and (13) can be reformulated. In equation (15) and (16), the ACFs define the covariance matrices $\mathbf{R}_w(m,\xi)$. With these matrices directional *CVM*-operators can be built (Appendix B).

The *CVM*-operator shown in Fig. 9(a) is built to extract line elements within an orientation range of $\Phi = [23.5°, 62.5°]$, i.e. diagonal line elements. Other parameters used to design this operator are: $\sigma_p = \Delta$, $\sigma_n = 0.8\sigma_a$, $\lambda = 0.05/\Delta^2$, $R_i = 3\Delta$, $\sigma_t = 4\Delta$, and M=7. Among the set of six filters only one contributes positively to the log-likelihood ratio. This filter more or less corresponds to a matched filter. The remaining five filters serve to suppress artifacts that would result if the matched filter would have been the sole filter. For instance, the first two filters are of first order and suppress gradients. The fourth filter suppresses line segments not within the preferred orientation range (e.g. orthogonal to the desired orientation). Fig. 9(b) shows the resulting log-likelihood ratio.

D. Classification Schemes

In contrast with the 1-D case, the 2-D model adopted in this work does not provide directives how to transform the log-liklihood ratio into edge or line maps. Simply thresholding gives rise to thickened edge structures. A more sophisticated classification strategy is needed.

In the 1-D case, the concept of non-local maximum suppression is effective. This non-local maximum suppression technique can be generalized to two dimensions quite easily if edge or line elements with a small orientation range are sought after. The classification scheme becomes as follows:

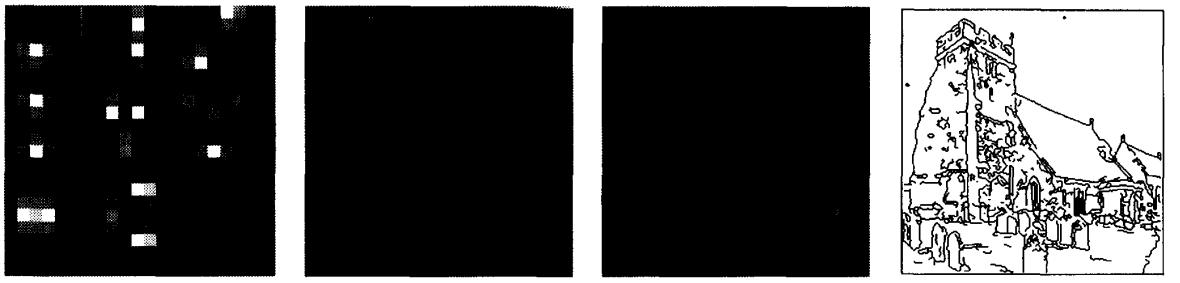
- If at a certain pixel its log-likelihood ratio exhibits a local maximum in the direction orthogonal to the main direction of the elements being sought after, and this ratio exceeds a threshold, then the pixel is marked as an edge (or line) element.

This classification scheme is applicable to the log-likelihood ratio shown in Fig. 9(b). Here, non-local maximum suppression must take place in a diagonal direction (top left to bottom right). The resulting line map is given in Fig. 9(c).

A non-local maximum suppression technique well suited to find all edges (regardless of their orientations) is difficult to define. The problem originates from the fact that edge ele ments are connected. Non-local maximum suppression tends to break down this connectedness, especially at branch points and corners. If rotationally invariant operators are used, as in Figs. 7 and 8, one strategy is to combine the local maximums found in two orthogonal directions. For instance:

- If at a certain pixel its log-likelihood ratio exhibits a local maximum in the vertical direction or in the horizontal direction and this ratio exceeds a threshold, then the pixel is marked as an edge (or line) element.
- Fig. 7(d) and 8(d) are edge and line maps that are obtained from this classification scheme. The maps indicate that the scheme performs well for segments with low curvature. However, near corners tiny spurs arise or the connectedness is affected. This also occurs at branch points.

Another approach is to use a combination of a set of directional operators. Essentially, this is the approach of Canny and



(a) (b) (c) (d)

Figure 10(a). Directional CVM-operator for horizontal edges. (b). Log-likelihood ratio of horizontal edges. (c) Log-likelihood ratio of vertical edges. (d). Non-directional edge map derived from combination of (b) and (c).

many others. In this classification scheme, each directional operator covers a small range of edge orientations. The whole set covers the full range. The strategy is to apply, at each pixel position, non-maximum suppression in the direction corresponding to the operator with maximum output. Then, a pixel is marked as an edge if the output of this operator exceeds a threshold.

Fig. 10 gives an example. Here the scheme is based on two directional operators, one for horizontal and another for vertical edge orientations. The scheme can easily be extended to more than two directions, but in pilot experiments [14] it appeared that this extension did not improve the performance. The *CVM*-operator shown in Fig. 10(a) is built to extract edge elements within an orientation range of $\Phi = [-45°, 45°]$, i.e. horizontal edge elements. Other parameters used to design this operator are $\sigma_p = \Delta$, $\sigma_n = 0.5\sigma_a$, $\lambda = 0.05/\Delta^2$, $R_i = 2\Delta$, $\sigma_i = 0.5\Delta$, $\sigma_r = 1.25\Delta$, and $M = 5$. These parameters are obtained by numerical optimization of the performance of the corresponding edge detector if applied to a well defined test image [14]. The operator does not fully suppress vertical edges, but this is quite harmless, since the ultimate goal is to detect all edges (regardless of their orientation). Fig. 10(b) shows the log-likelihood ratio resulting from the horizontal *CVM*-operator. The vertical operator is obtained by transposition of the kernels of the horizontal operator. Fig. 10(c) shows the final edge map.

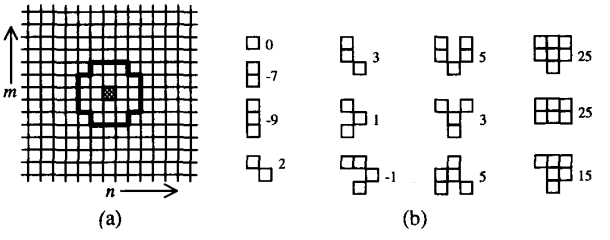
E. Conjunction With MRF-Models

The classification schemes discussed so far are all based on the assumption that the loci of edge elements form segments with low curvature. This assumption is violated near corners and branch points. A more scientific approach is to model all kinds of permissible edge configurations explicitly. One way to formulate the problem is by putting it within a statistical framework. For that purpose, define for each pixel $(n\Delta, m\Delta)$ a state ω_{nm} denoting whether the pixel is an edge or not: $\omega_{nm} \in \{\text{edge, non-edge}\}$. The set of all states in the image plane is denoted Ω. Then, the set of all permissible edge configurations can be made explicit by defining the probability of having a configuration Ω, i.e. $P(\underline{\Omega} = \Omega)$. Edge configurations that are not permitted (e.g. thickened edge structures, fragmented segments, etc.) have zero probability. In the remaining part of this section we will show that the *CVM*-model and its associated operators can be coupled to this notion of "permissible edge configurations."

Suppose that the "true" edge map of an acquired image $w(x, y)$ is given by Ω. Furthermore, suppose that the samples $w(n\Delta, m\Delta)$ of the image are collected in a set W. The probability of having such an edge map with associated image is $P(\underline{\Omega} = \Omega, \underline{W} = W)$. The problem is to find Ω given the image W. If the *uniform* cost function [12] is adopted, the Bayes solution is the MAP estimate $\hat{\Omega}$ which turns out to be the configuration that maximizes $P(\underline{\Omega} = \Omega)P(\underline{W} = W|\underline{\Omega} = \Omega)$.

One possibility to find a solution is to model the probability $P(\underline{\Omega} = \Omega)$ with the concept of Markov random fields (MRF). The basic assumption is that the conditional probability for a state $\underline{\omega}_{nm}$ (i.e. edge or non-edge) for a pixel with coordinates $(n\Delta, m\Delta)$ given all other states in the image, is the same as the conditional probability of $\underline{\omega}_{nm}$ given the states in only some local neighborhood [1]. Fig. 11(a) gives an example of a neighborhood.

The merit of a MRF is that $P(\underline{\Omega} = \Omega)$ can be broken down into a composite of simple terms, thereby offering an enormous reduction of the complexity. To show this, *cliques* must be introduced. A clique $C_i(n, m)$ is a subset of the neighborhood of pixel (n, m) such that each pixel in the clique is in the neighborhoods of all other pixels in



(a) (b)

Figure 11(a). A neighbourhood. (b) Some cliques with associated potentials.

that clique. Some examples of cliques of the neighborhood of Fig. 11(a) are shown in Fig. 11(b). It is proven that for all MRFs the probability $P(\underline{\Omega} = \Omega)$ can be written as

$$P(\underline{\Omega}=\Omega)=\frac{1}{Z}\exp\left(\frac{-U(\Omega)}{T}\right) \tag{28}$$

T and Z are constants. $U(\Omega)$ is called the energy function. It is defined as

$$U(\Omega)=\sum_{n,m}\sum_{i}F_i\{C_i(n,m)\} \tag{29}$$

The terms $F_i\{C_i(n,m)\}$ are functions of the states of the pixels in the cliques associated with the pixel at (n,m). In Fig. 11(b), potentials are tabulated for each clique shown. If the configuration of edge elements within the neighborhood of (n, m) matches a particular clique $C_i(\cdot)$, the function $F_i\{\cdot\}$ returns the associated potential. If not, the function returns zero.

The consequence of this is that if the potential of a clique is large, the probability of having such a configuration of edge elements becomes low (and *vice versa*). The potentials in Fig. 11(b) are chosen such that: a) thickened edge structures become very unlikely, b) unfragmented segments are favored, and c) branch points are allowed.

Equations (28) and (29) show that, with given potentials and given Ω a numerical evaluation of $P(\underline{\Omega}=\Omega)$ is feasible. In fact, numerical methods are known with which the configurations can be found that maximize $P(\underline{\Omega}=\Omega)$. An example is simulated annealing [10]. These methods are computationally expensive. Other methods are known that give suboptimal solutions, but with the advantage that they are less computationally expensive (for instance: iterative conditional modes estimation [1], [26]).

All these methods rely on the calculation of conditional state probabilities. Let Ω_{nm} denote all pixels in the neighborhood of (n,m) except ω_{nm} itself. A corollary of equation (28) and (29) is that

$$P(\underline{\omega}_{nm}=\omega_{nm}|\underline{\Omega}_{nm}=\Omega_{nm})=\frac{1}{\breve{Z}}\exp\left(-\frac{1}{T}U_{nm}\right) \tag{30}$$

with $\breve{Z}$ a normalizing constant and

$$U_{nm}=\sum_{i}F_i\{C_i(n,m)\} \tag{31}$$

Finding the configuration Ω that maximizes $P(\underline{\Omega}=\Omega)$ boils down to repeatedly visiting the pixels (n,m) and changing their states in dependency on their conditional state probability, or equivalently, on their relative change of energy ΔU_{nm}

$$\frac{\Delta U_{nm}}{T}=\log\left(\frac{P(\underline{\omega}_{nm}=\text{edge}|\underline{\Omega}_{nm}=\Omega_{nm})}{P(\underline{\omega}_{nm}=\text{non-edge}|\underline{\Omega}_{nm}=\Omega_{nm})}\right) \tag{32}$$

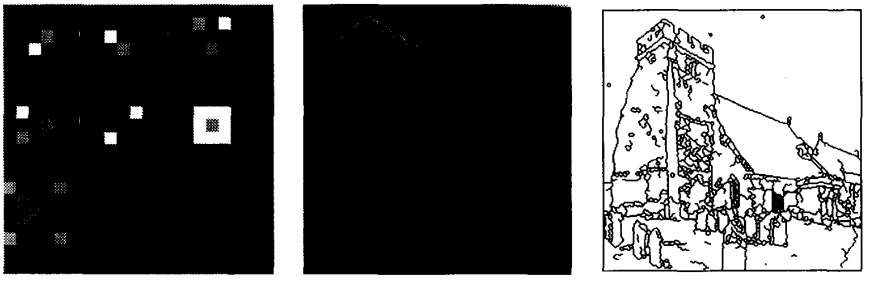
We now return to the problem of maximizing $P(\underline{\Omega}=\Omega)P(\underline{W}=W|\underline{\Omega}=\Omega)$. Finding an exact expression of $P(\underline{W}=W|\underline{\Omega}=\Omega)$ is very difficult. Heuristic methods are presented in [11], [16]. The *CVM*-model offers a suboptimal solution in which $P(\underline{W}=W|\underline{\Omega}=\Omega)$ is replaced by $P(\underline{W}=W|\underline{\omega}_{nm}=\omega_{nm})$. In that case, the solution is found by augmenting the change of energy in equation (32) with the log-likelihood ratio given in equation (8) and (23), i.e.

$$\log\left(\frac{P(\underline{W}=W|\underline{\omega}_{nm}=\text{edge})}{P(\underline{W}=W|\underline{\omega}_{nm}=\text{non-edge})}\right)\approx-\log\left(\sqrt{|\Gamma|}\right)+\frac{1}{2}v_{nm} \tag{33}$$

The change of energy to be used in a numerical optimization procedure becomes

$$\frac{\Delta U_{nm}}{T}-\log\left(\sqrt{|\Gamma|}\right)+\frac{1}{2}v_{nm} \tag{34}$$

Fig. 12 gives an example in which this strategy is applied. The *CVM*-operator shown is a rotationally invariant edge feature extractor with parameters: $\sigma_p=0.5\Delta$, $\sigma_n=0.05\sigma_a$, $\lambda=0.05/\Delta^2$, $R_i=2\Delta$, $\sigma_i=0.5\Delta$, $\sigma_r=1.25\Delta$ and $M=5$. The log-likelihood ratio corresponding to Fig. 1(a)is given in Fig. 12(b). The edge map in Fig. 12(c) results from application of simulated annealing (the Gibbs sampler [10]).



(a) (b) (c)

Figure 12(a) Rotationally invariant cvm-operator. (b) Log-likelihood ratios for edges. (c) Edge map derived from b) and the MRF from Fig. 11 using stochastic optimisation.

V. Conclusion

The basic assumption in this work is that the most disturbing factors in edge and line detection are image discontinuities coming from neighboring objects. The consequence of this is twofold. First, it demands that the existence of these neighboring discontinuities are modeled explicitly. Second, since in general the contrasts between neighboring objects vary from object to object, a variability of strengths of image discontinuities must be envisaged. In fact, in most applications a zero mean, unimodal distribution function of these strengths is quite realistic.

Most methods to design edge and line features rely on a model describing a single discontinuity immersed in Gaussian noise. Usually, the criterion function for both design and performance evaluation is given in terms of the signal-to-noise ratio, root-mean-square of the localization error, and the mean distance between neighboring detections. However, it is doubtful whether these criterion functions are applicable in cases with multiple image discontinuities.

The *CVM*-operators developed in this work are based on a model in which the image data is described in terms of conditional covariances. Such a model covers some factors that are not met in other models: explicit modeling of multiple image discontinuities, and modeling of varying strengths. The model also encompasses non-white noise and non-ideal discontinuities. The *CVM*-operators are Bayes detectors/estimators with unit cost function for both classification errors and localization errors.

In order to evaluate the newly defined operators and to compare them with other operators a performance measure is developed based on the average risk. This measure is a weighted sum of the densities of several types of errors (e.g. false alarms, localization errors). The measure can be used to optimize the parameters of the various operators. It can also be used to analyze the behavior of an operator.

Experiments have shown that within the operators of the Canny family the one without FIR constraint (Sarkar-Boyer) performs best. Optimization of the mean distance between neighboring detections (Spacek, Petrou-Kittler) does not appear to be an appropriate criterion. Application of the Shen-Castan criterion leads to a detector with excellent localization ability, but at the cost of many multiple responses, false alarms and missed events. The *CVM*-operator consists of a parallel bank of filters. Taken together these filters combine the localization ability of the Shen-Castan operator with the multiple response suppression and detection ability of the Canny-like filters. In our experiments the *CVM*-operator outperforms all other operators. This indicates that our basic assumption holds.

A full generalization of the *CVM*-model to two dimensions appears to be difficult because of the connectedness and mutual dependencies of edge elements. However, an approximate solution (with some loss of optimality) appears to be feasible. It has been shown that the 2-D *CVM*-model leads to a set of filters that are useful in edge or line feature extraction.

The output image of a *CVM*-operator can be placed in the context of a probabilistic framework. The Geman and Geman method of discovering discontinuities (and related work) is based on such a framework, i.e. a MRF model. Provided that the clique potentials in a MRF are determined properly this model is well suited to cover the geometric properties of edge elements. However, the radiometric description of edges — e.g. the edge spread function — in MRFs becomes either quite complex or inaccurate. The *CVM*-model provides a systematic way to build operators that transform image data into log-likelihood ratios. Conceptually, these ratios fit well in a MRF. Therefore, the *CVM*-operators are complementary to MRF-based edge- and line detectors.

Appendix A: Autocovariance Functions of Inhibited Poisson Processes

We consider a zero mean random process $\underline{w}(x)$ given by:

$$\underline{w}(x) = \sum_k \underline{a}_k s(x - \underline{\xi}_k)$$

$\left(\underline{\xi}_k\right)$ is a Poisson point process with non-stationary density $\lambda(x)$. $\{\underline{a}_k\}$ is an uncorrelated Gaussian random sequence with variance σ_a^2. With fixed coefficients $\underline{a}_k = a$ the autocorrelation of $\underline{w}(x)$ is given by [19]:

$$R_{ww}(x_1, x_2) = \mathrm{E}\{\underline{w}(x_1)\underline{w}(x_2)\} = a^2 \int \lambda(\xi) s(x_1 - \xi) s(x_2 - \xi) d\xi \tag{a.1}$$

If $\lambda(x) = \lambda$ everywhere, except for an interval X in which $\lambda(x) = 0$ this integral degenerates into:

$$R_{ww}(x_1, x_2) = a^2 \lambda \int_{\alpha \notin X} s(x_1 + \alpha) s(x_2 + \alpha) d\alpha \tag{a.2}$$

The case in which the sequence $\{\underline{a}_k\}$ is random is discussed in [2] and [3]. If independence is assumed in the random variables $\underline{a}_k$, then:

$$\begin{aligned} R_{ww}(x_1, x_2) = {} & \sigma_a^2 \lambda \int_{\alpha=-\infty}^{\infty} s(x_1 - x_2 + \alpha) s(\alpha) d\alpha \\ & - \sigma_a^2 \lambda \int_{\alpha \in X} s(x_1 + \alpha) s(x_2 + \alpha) d\alpha \end{aligned} \tag{a.3}$$

If the inhibition distance is small, i.e. $R_i \lambda << 1$, then the probability of having two or more points in an interval R_i is negligible. Hence, the ACF of an inhibited Poisson process conditioned that no point exists in an interval X equals:

$$\begin{aligned} R_{pp}(x_1, x_2 | \underline{\xi}_k \notin X) \approx {} & \sigma_a^2 \lambda \int_{\alpha=-\infty}^{\infty} s(x_1 - x_2 + \alpha) s(\alpha) d\alpha - \\ & \sigma 2 \lambda \int_{\alpha \in X} s(x_1 + \alpha) s(x_2 + \alpha) d\alpha \end{aligned} \tag{a.4}$$

On the other hand, if it is known that a point exist at $\xi_k \in X_k$ the ACF corresponding to that single point is

$$R_{ss}(x_1, x_2 | \xi_k \in X_n) = \sigma_a^2 s(x_1 - \xi_k) s(x_1 - \xi_k) \tag{a.5}$$

This single point will inhibit other points within an interval around ξ_k. Therefore, the remaining part of the inhibited Poisson process has ACF:

$$R_{pp}(x_1, x_2 | \underline{\xi}_k \in X_n) \approx \sigma_a^2 \lambda \int_{\alpha=-\infty}^{\infty} s(x_1 - x_2 + \alpha) s(\alpha) d\alpha - \sigma_a^2 \lambda \int_{\alpha=\xi_k - R_i}^{\xi_k + R_i} s(x_1 + \alpha) s(x_2 + \alpha) d\alpha \tag{a.6}$$

Note that in the case of edges, the first integral in both (a.4) and (a.6) has an anomaly caused by the infinite, non-zero overlap between $s(x_1 - x_2 + \alpha)$ and $s(\alpha)$. In fact, for edges, $\Sigma \underline{a}_k s(x - \underline{\xi}_k)$ is a random process with stationary increments. One way to avoid this anomaly is to window the edge profile $s(x)$ by a symmetrical window function, which is unity at $x=0$, and which decays smoothly to zero as x varies from -∞ and ∞. The width of this window function must be chosen large compared with $1/\lambda$. By doing so, the long range influence of far-away edges (which cause only changes in the background level) is removed, while the influence of nearby edges is maintained.

Appendix B: How To Build CVM-Operators

Input Parameters:

λ	$[1/\Delta]$	Density of step or pulse functions
R_i	$[\Delta]$	Inhibition distance
σ_a^2		Variance of amplitudes
$s(x)$		Step or line profile
σ_b^2		Variance background level
$R_{nn}(x)$		Autocorrelation function of noise
Δ		Sampling period

Note: if λ and R_i are normalized appropriately, the sampling period Δ can be set to 1. Likewise $R_{nn}(x)$ and σ_b^2 can be normalized to σ_a^2.

Algorithm:

1) Choose N (number of samples in a convolution kernel), such that $N\lambda\Delta >> 1$.
2) Calculate the conditional covariance matrices $\boldsymbol{R}_0 = \boldsymbol{R}_w(0,0)$ and $\boldsymbol{R}_1 = \boldsymbol{R}_w(1,0)$ by numerical evaluation of equation (15) and (16).
3) Principal component analysis:
   3.1) Determine a $N \times N$ matrix $\mathbf{V}$, such that $\mathbf{V}\mathbf{R}_0\mathbf{V}^t = \mathbf{I}$ (Note: $\mathbf{V}$ can be determined by use of an eigenvector analysis of $\mathbf{R}_0$)
   3.2) Determine the orthonormal matrix $\mathbf{U}$ containing the eigenvectors of the matrix $\mathbf{V}\mathbf{R}_1\mathbf{V}^t$. Determine the eigenvalues γ_i, $i = 0, \cdots, N-1$ belonging to the eigenvectors in $\mathbf{U}$.
4. Feature Selection:
   4.1) Calculate the Bhattacharyya distances $J_b(D)$, $D = 0, \cdots, N-1$ according equation (19).
   4.2) Sort the eigenvectors in $\mathbf{U}$ and the eigenvalues γ_i, such that: $J_b(D-1) \geq J_b(D)$. Choose D as small as possible, but such that still $J_b(D) \approx J_b(N-1)$.
   4.3) Select only the first D eigenvectors in $\mathbf{U}$ (i.e. $\mathbf{U}$ becomes a $N \times D$ matrix).
   4.4) Calculate $\mathbf{T} = \mathbf{U}^t\mathbf{V}$.
5. The convolution kernels are given in the rows of the matrix $\mathbf{T}$. The weights are given by $1 - 1/\gamma_i$.

Appendix C: Autocovariance Functions of 2-Dimensional Poisson Processes

We consider a zero mean 2-D random process $\underline{w}(x,y)$ given by:

$$\underline{w}(x,y) = \sum_k \underline{a}_k s(x - \underline{\xi}_k, y - \underline{\eta}_k)$$

$(\underline{\xi}_k, \underline{\eta}_k)$ is a Poisson point process with non-stationary density $\lambda(x,y)$ and $\{\underline{a}_k\}$ is an uncorrelated Gaussian random sequence with variance σ_a^2. With fixed coefficients $\underline{a}_k = a$ the autocorrelation of $\underline{w}(x,y)$ is given by [19]:

$$R_{ww}(x_1, y_1, x_2, y_2) = \mathrm{E}\{\underline{w}(x_1, y_1)\underline{w}(x_2, y_2)\} = a^2 \iint \lambda(\xi, \eta) s(x_1 - \xi, y_1 - \eta) s(x_2 - \xi, y_2 - \eta) d\xi d\eta$$

If $\lambda(x,y) = \lambda$ everywhere, except for a region A in which $\lambda(x,y) = 0$ this integral degenerates into:

$$R_{ww}(x_1, y_1, x_2, y_2) = a^2 \lambda \iint_{(\alpha,\beta) \notin A} s(x_1 + \alpha, y_1 + \beta) s(x_2 + \alpha, y_2 + \beta) d\alpha d\beta$$

Using the notation $\vec{x}$ and $\vec{\alpha}$ for the coordinates (x,y) and (α,β), respectively, this expression is rewritten into

$$R_{ww}(\vec{x}_1, \vec{x}_2) = a^2 \lambda \iint_{\vec{\alpha}} s(\vec{x}_1 - \vec{x}_2 + \vec{\alpha}) s(\vec{\alpha}) d\alpha d\beta - a^2 \lambda \iint_{\vec{\alpha} \in A} s(\vec{x}_2 + \vec{\alpha}) s(\vec{x}_2 + \vec{\alpha}) d\alpha d\beta$$

The generalization to the case in which the sequence $\{\underline{a}_k\}$ is random is straightforward as long as independence is assumed in the random variables $\underline{a}_k$:

$$R_{ww}(\vec{x}_1,\vec{x}_2) = \sigma_a^2\lambda\iint_{\vec{\alpha}} s(\vec{x}_1-\vec{x}_2+\vec{\alpha})s(\vec{\alpha})d\alpha d\beta - \sigma_a^2\lambda\iint_{\vec{\alpha}\in A} s(\vec{x}_2+\vec{\alpha})s(\vec{x}2+\vec{\alpha})d\alpha d\beta \quad \text{(c.1)}$$

Next we consider the random process, given by:

$$\underline{w}(x,y) = \sum_k \underline{a}_k s(x-\underline{\xi}_k, y-\underline{\eta}_k, \underline{\varphi}_k)$$

with $\{\underline{\varphi}_k\}$ random variables which affect the shape of the signals $s(.,.,\varphi_k)$. If the sequences $\{\underline{a}_k\}$ and $\{\underline{\varphi}_k\}$ are independent then:

$$R_{ww}(\vec{x}_1,\vec{x}_2) = \sigma_a^2\lambda\iint_{\vec{\alpha}} \mathrm{E}\{s(\vec{x}_1-\vec{x}_2+\vec{\alpha},\underline{\varphi}_k)s(\vec{\alpha},\underline{\varphi}_k)\}d\alpha d\beta - \sigma_a^2\lambda\iint_{\vec{\alpha}\in A} \mathrm{E}\{s(\vec{x}_2+\vec{\alpha},\underline{\varphi}_k)s(\vec{x}_2+\vec{\alpha},\underline{\varphi}_k)\}d\alpha d\beta$$

where the expectation in the integrand is taken with respect to $\underline{\varphi}_k$. That is, if $\underline{\varphi}_k$ is uniformly distributed within an interval Φ then

$$R_{ww}(\vec{x}_1,\vec{x}_2) = \frac{\sigma_a^2\lambda}{\|\Phi\|}\iint_{\vec{\alpha}}\int_{\varphi\in\Phi} s(\vec{x}_1-\vec{x}_2+\vec{\alpha},\varphi)s(\vec{\alpha},\varphi)\}d\varphi d\alpha d\beta - \frac{\sigma_a^2\lambda}{\|\Phi\|}\iint_{\vec{\alpha}\in A}\int_{\varphi\in\Phi} s(\vec{x}_2+\vec{\alpha},\varphi)s(\vec{x}_2+\vec{\alpha},\varphi)d\varphi d\alpha d\beta \quad \text{(c.2)}$$

REFERENCES

[1] J. Besag, "Spatial Interaction and the Statistical Analysis of Lattice Systems," *J. Royal Statistical Society*, Ser. B. Vol. 36, 19two-236, 1974.

[2] F.J. Beutler, O.A.Z. Leneman, "The Spectral Analysis of Impulse Processes," *Inform. and Control*, Vol. 12, No. 3, pp. 236-258, 1968.

[3] F.J. Beutler, O.A.Z. Leneman, "On the Statistics of Random Pulse Processes," *Inform. Control*, Vol. 18, No. 4, pp. 326-341, 1971.

[4] R.A. Boie, I.J. Cox, P. Rehak, "On Optimum Edge Recognition using Matched Filters," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 100-108, 1986.

[5] R.A. Boie, I.J. Cox, "Two Dimensional Optimum Edge Recognition using Matched and Wiener Filters for Machine Vision," *Proc. Int. Conf. Computer Vision*, London, pp. 450-456, 1987.

[6] J. Canny, "A Computational Approach to Edge Detection," *IEEE Tr PAMI*, Vol. 8, pp. 679-698, November 1986.

[7] M.H. Chen, D. Lee, T. Pavlidis, "Residual Analysis for Feature Detection," *IEEE Tr. PAMI*, Vol. 13, pp. 30-40, January 1991.

[8] R. Deriche, "Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detection," *Int. J. Computer* Vision, pp. 167-187, 1987.

[9] P.A. Devijver, J. Kittler, "Pattern Recognition - a Statistical Approach," Prentice Hall, Englewood Cliffs, New Jersey, 1982.

[10] S. Geman, D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Tr. PAMI*, Vol. 6, pp. 721-741, November 1984.

[11] D. Geman, S Geman, C. Graffigne, "Locating Texture and Objects," *Pattern Recognition Theory and Practice*, P.A. Devijver and J. Kittler, Eds. Springer Verlag, Heidelberg, 1987.

[12] R.M. Haralick, "Decision Making in Context," *IEEE Tr. PAMI*, Vol. 5, pp. 417-428, July 1983.

[13] F. van der Heijden, "Evaluation of Edge Detection Algorithms," *Proc. Third Int. Conf. on Image Processing, IEE*, Warwick, 1989.

[14] F. van der Heijden, "A Statistical Approach to Edge and Line Detection in Digital Images," Ph.D. Thesis, Faculty of Electrical Engineering, University of Twente, Enschede, 1992.

[15] F. van der Heijden, "Detection and Localisation of a Single Profile Function with Random Height," University of Twente, internal report, BSC 93-M271, 1993.

[16] S. Hongo, M. Kawato, T. Inui, S. Miyake, "Contour Extraction of Images on Parallel Computer - Local, Parallel and Stochastic Algorithm which Learns Energy Parameters," *Proc. IJCNN*, Vol. 1, Washington D.C., pp. 161-167, 1989.

[17] J.J. Koenderink A.J. van Doorn, "Generic Neighborhood Operators," *IEEE Tr. PAMI*, Vol. 14, pp. 597-605, June 1992.

[18] D. Marr, E.C. Hildreth, "Theory of Edge Detection," *Proc. Royal Soc.* London B, Vol. 207, pp. 187-217, 1980.

[19] A. Papoulis, "Probability, Random Variables and Stochastic Processes," McGraw-Hill, Tokyo, 1965.

20. M. Petrou, J. Kittler, "Optimal Edge Detectors for Ramp Edges," *IEEE Tr PAMI*, Vol. 13, pp. 483-491, May 1991.

[21] S. Sarkar, K.L. Boyer, "On Optimal Infinite Impulse Response Edge Detection Filters," *IEEE Tr PAMI*, Vol. 13, pp. 1154-1171, November 1991.

[22] K.S. Shanmugam, F.M. Dickey, J.A. Green, "An optimal Frequency Domain Filter for Edge Detection in Digital Pictures," *IEEE Tr.* Vol. 1, Vol. 1, pp. 37-49, January 1979.

[23] J. Shen, S. Castan, "An Optimal Linear Operator for Step Edge Detection," *CVGIP*, Vol. 54, pp. 112-133, 1992.

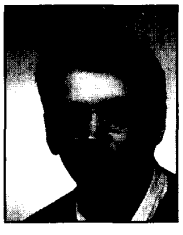
[24] L.A. Spacek, "Edge Detection and Motion Detection," *Image and Vision Computing*, Vol 4, pp. 43-56, February 1986.

[25] L.J. Spreeuwers, F. van der Heijden, "Evaluation of Edge Detectors Using Average Risk," *11th IAPR International Conference on Pattern Recognition*, The Hague, 1992.

[26] C.W. Therrien, "An Estimation-Theoretic Approach to Terrain Image Segmentation," *CVGIP*, Vol. 22, pp. 313-326, 1983.

[27] V. Torre, T.A. Poggio, " On Edge Detection," *IEEE Tr PAMI*, Vol. 8, pp. 147-163, March 1986.

[28] D. Ziou, "Line Detection Using an Optimal IIR Filter," *Pattern Recognition*, Vol 24, pp. 465-678, 1991.



Ferdinand van der Heijden received the ingenieur degree in 1981 from the University of Twente, the Netherlands. In 1982 he joined the Group for Measurement Science and Instrumentation of that same university. Until 1989 his main activities concerned the development of image based measurement systems with industrial applications. He received the Ph.D. degree in 1992. His current research interests are in computer vision, image analysis and pattern recognition