

Contour Estimation by Array Processing Methods

Salah Bourennane^{1,2} and Julien Marot^{1,2}

¹ GSM, Institut Fresnel/CNRS-UMR 6133, Université Aix-Marseille III, D.U. de Saint Jérôme,
13397 Marseille Cedex 20, France

² École Généraliste d'Ingénieurs de Marseille (EGIM), Technopôle de Château-Gombert, 38 rue Joliot Curie,
13451 Marseille Cedex 20, France

Received 8 February 2005; Revised 16 November 2005; Accepted 29 December 2005

Recommended for Publication by Gloria Menegaz

This work is devoted to the estimation of rectilinear and distorted contours in images by high-resolution methods. In the case of rectilinear contours, it has been shown that it is possible to transpose this image processing problem to an array processing problem. The existing straight line characterization method called subspace-based line detection (SLIDE) leads to models with orientations and offsets of straight lines as the desired parameters. Firstly, a high-resolution method of array processing leads to the orientation of the lines. Secondly, their offset can be estimated by either the well-known method of extension of the Hough transform or another method, namely, the variable speed propagation scheme, that belongs to the array processing applications field. We associate it with the method called “modified forward-backward linear prediction” (MFBLP). The signal generation process devoted to straight lines retrieval is retained for the case of distorted contours estimation. This issue is handled for the first time thanks to an inverse problem formulation and a phase model determination. The proposed method is initialized by means of the SLIDE algorithm.

Copyright © 2006 Hindawi Publishing Corporation. All rights reserved.

1. INTRODUCTION

The estimation of the characteristics of lines and object contours from a sequence of binary images has been a widely studied problem over the past few years [1–3]. This type of problem is faced in robotic way screening, in the measurement of wafer track width in microelectronics, and generally in aerial image analysis. The image contains straight lines compound of black pixels with value “1,” over a white background with pixels value “0.” The detection and localization of these straight lines are essential issues in image processing [4].

The Hough transform can be used for this purpose [1, 2]. Although this method gives a good resolution even in presence of a relatively strong noise, some restrictions remain in its use. These restrictions are due to the dependence on the choice of the quantization step and the computational cost for the bidimensional search of the maxima.

Array processing methods consist in conjugating the parameters of both arrays and received signals. Their efficiency has been improved and led to efficient algorithms [5]. In order to keep the resolution and reduce the computational cost, the array processing methods [6, 7] have recently been

adapted to give the characteristics of multiple straight lines out of an image.

In this paper, we first recall in Section 2 how to adapt the estimation of straight lines as a classical array processing problem as was developed earlier in the SLIDE algorithm [8–10]. A straight line in an image is characterized by two parameters that are successively estimated. In order to estimate the orientation of the straight lines, the SLIDE algorithm [6] employs a constant speed propagation scheme and a high-resolution method [11, 12] that is based on the computation of a covariance matrix [13, 14]. Two different methods devoted to the estimation of the offsets are set forth; the first one is the extension of the Hough transform [1–3], and the second one employs the spectral analysis method “modified forward-backward linear prediction” (MFBLP) [15] after setting a *variable speed* propagation scheme [9] for the transcription of the content of the image as a signal.

The study dedicated to straight lines retrieval will be used as a basis for the distorted contours estimation for which we propose in Section 3 a new algorithm called ECAPMO (estimation of contours by array processing methods and optimization). By using this method the estimation of *distorted contours* is obtained thanks to the contribution of array

processing methods and numerical optimization by formulating an inverse problem, starting from the data generated from an image.

In Section 4, we show that it is possible to generalize this method for single curve detection to a method for the retrieval of the characteristics of several curves in an image. In Section 5, several examples illustrate the performances of the proposed algorithms. In the case of straight lines estimation, results obtained on binary images are presented, concerning especially polygonal contours. Then the results obtained by the ECAPMO method on distorted curves are commented. Several examples of practical applications in various domains are quoted.

2. STRAIGHT CONTOURS ESTIMATION

2.1. Data model: generation of the signals out of the image data

Let $I(x, y)$ be the recorded image (see [4, Figure 1 (a)]). We consider that $I(x, y)$ is a compound of d straight lines and an additive uniformly distributed noise. Moreover, in this model, image $I(x, y)$ is supposed to contain only pixel values “1” or “0” [6]. Pixels “1,” which form the straight lines, are called “useful pixels,” whereas “0” pixels are associated to the background. The image size is $N \times C$: it contains N lines and C columns. Each straight line within an image is associated to an offset x_0 on the X axis and to angle θ , between this straight line and the line of equation $x = x_0$ (Figure 1(b)).

It is possible to establish the analogy between the localization of sources [7, 12, 16] in array processing and the recognition of lines in image processing. For this purpose some signals are generated out of the image data [10]: we create artificially, out of the N lines of the image matrix, N inputs to a linear array composed of N equidistant sensors ranged along the image side. The position of each pixel on a given line has an influence on the signal received by the corresponding sensor. We can therefore define the signal received by the i th sensor as the superposition of the useful pixels belonging to the corresponding line. When d lines are present in the image, there are d nonzero pixels on the i th line of the image-matrix, localized on the columns x_1, \dots, x_d , respectively. The signal received by the sensor in front of the i th line, when no noise is present in the image, is written as [4]

$$z(i) = \sum_{k=1}^d \exp(-j\mu x_k(i)), \quad i = 1, \dots, N, \quad (1)$$

where μ is a propagation parameter [9, 10] that can be constant or variable: we can consider a constant or variable parameter propagation scheme. First we consider the case of only one line with angle θ and offset x_0 , as it is shown in Figure 1(b). Supposing that the width, along the X axis, of each line is equal to one pixel, the horizontal coordinate of a straight line pixel in front of the i th sensor is

$$x(i) = x_0 - (i - 1) \tan(\theta). \quad (2)$$

Hence the signal received on the i th sensor is written as

$$z(i) = \exp(-j\mu x(i)), \quad (3)$$

$$z(i) = \exp(-j\mu x_0) \exp(j\mu(i - 1) \tan(\theta)).$$

In this expression we took into account the possible values of each pixel “1” or “0.” In the presence of d different straight lines in the image and an additive noise, the signal received on the sensor i is

$$z(i) = \sum_{k=1}^d \exp(j\mu(i - 1) \tan(\theta_k)) \exp(-j\mu x_{0k}) + n(i), \quad (4)$$

where $n(i)$ is the noise on the i th line that can be due to several useful pixels. As a consequence of the presence of noisy nonzero pixels, the linear variation of the phase in expression (3) is no longer verified.

Defining

$$a_i(\theta_k) = \exp(j\mu(i - 1) \tan(\theta_k)), \quad s_k = \exp(-j\mu x_{0k}), \quad (5)$$

expression (3) becomes

$$z(i) = \sum_{k=1}^d a_i(\theta_k) s_k + n(i), \quad i = 1, \dots, N. \quad (6)$$

Equation (6) gives the signal model that will be employed in the following, and that fully characterizes the d lines within the noisy image.

2.2. Estimation of the angles: overview of the SLIDE method

The method for angles estimation falls into two parts: the estimation of a covariance matrix and the application of a total least squares criterion.

Numerous works have been developed in the frame of the research of a reliable estimator of the covariance matrix when the duration of the signal is very short or the number of realizations is small. This situation is often encountered, for instance, with seismic signals. To cope with it, numerous frequency and/or spatial means are computed to replace the temporal mean. In this study the covariance matrix is estimated by using the spatial mean [17]. From the observation vector we build K vectors of length M with $d < M \leq N - d + 1$. In order to maximize the number of subvectors, we choose $K = N + 1 - M$. By grouping the whole subvectors obtained in matrix form, we obtain

$$\mathbf{Z}_K = [\mathbf{z}_1, \dots, \mathbf{z}_K], \quad (7)$$

where

$$\mathbf{z}_l = \mathbf{A}_M(\theta) \mathbf{s}_l + \mathbf{n}_l, \quad l = 1, \dots, K. \quad (8)$$

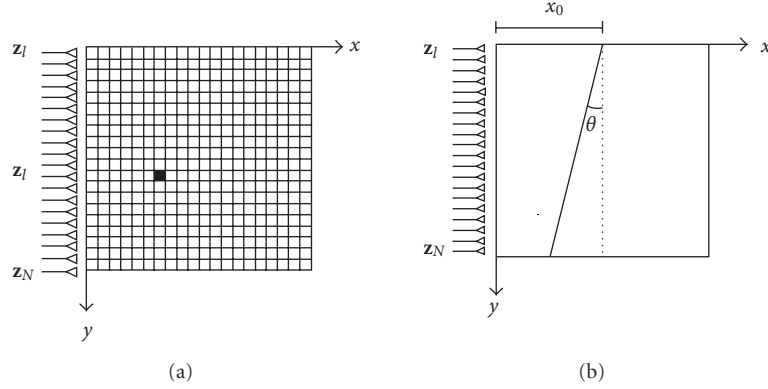


FIGURE 1: The image model (see [4]). (a) The image-matrix provided with the coordinate system and the rectilinear array of N equidistant sensors. (b) A straight line characterized by its angle θ and its offset x_0 .

$\mathbf{A}_M(\theta) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_d)]$ is a Vandermonde-type matrix of size $M \times d$. The signal is supposed to be independent of the noise; the components of noise vector \mathbf{n}_l are supposed to be uncorrelated and to have identical variance. The covariance matrix can be estimated from the observation subvectors as it is performed in [4]. Using the subvectors in the forward and the backward sense leads to a better estimation of the covariance matrix [16, 18]. The eigen-decomposition of the covariance matrix is, in general, used to characterize the sources by subspace techniques in array processing. In the frame of image processing the aim is to estimate the angle θ of the d straight lines. Several high-resolution methods that solve this problem have been proposed in the literature [11–13]. SLIDE algorithm is applied to a particular case of an array consisting of two identical subarrays [9]. It leads to the following estimated angles [9]:

$$\hat{\theta}_k = \tan^{-1} \left[\frac{1}{(\mu * \Delta)} \operatorname{Im} \left(\ln \left(\frac{\lambda_k}{|\lambda_k|} \right) \right) \right], \quad (9)$$

where $\{\lambda_k, k = 1, \dots, M\}$ are the eigenvalues of a diagonal unitary matrix that relates the measurements from the first subarray to the measurements resulting from the second subarray. Parameter μ is the propagation constant, and Δ is the distance between two sensors. The determination of the offsets (x_{0_k}) of the rectilinear curves forms the last step of the method. It exploits the straight lines angles θ_k that have been estimated previously.

2.3. Estimation of the offsets

The aim of this part is to present two methods that lead to the estimation of the offsets of the straight lines, when their angle is known. The first one is the well-known “extension of the Hough transform” [14]. It is based on the projection of the image along the straight line angle values. The second proposed method remains in the frame of array processing: it employs a variable parameter propagation scheme [8–10] and uses a high-resolution method. This high-resolution

“MFBLP” method relies on the concept of forward and backward organization of the data [17–19].

2.3.1. “Extension of the Hough transform” Method

We consider the polar parametrization. We call the representation of the values taken by the Hough transform for all considered values of polar coordinates θ and ρ “sinogram”. For a fixed θ value, the sinogram depends only on the ρ variable. The two polar coordinates can define a straight line. The distance $\{\rho_k\}$ between the origin and the straight line indexed by k is estimated by projecting the image along the orientation of polar coordinate θ_k and by retrieving

$$\rho_k = \arg \max_{-\sqrt{2}N \leq \rho \leq \sqrt{2}N} \sum_{i=1}^{i=N_p} c(\rho - x_i \cos \theta_k - y_i \sin \theta_k), \quad k = 1, \dots, d, \quad (10)$$

where N is the size of the image, N_p is the number of useful pixels having components (x_i, y_i) , contained in the image and c is the real function defined for a given variable value r and a width parameter R by

$$c(r) = \begin{cases} \cos \left(\frac{\pi r}{2R} \right) & \text{if } |r| < R, \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

The offsets are obtained by the relation

$$\rho_k = x_{0_k} \cos \theta_k. \quad (12)$$

This method has a good behavior in the presence of noise. In practice we will take $R = 3$ pixels. This parameter can be reduced in order to improve the estimation of the offsets [14]. The drawback of the method is its numerical cost. When the number of nonzero pixels in the image is large, the

summation in (10) contains a large number of terms. A major property of the extension of the Hough transform is that the case of several straight lines for a given angle value can be treated if several local maxima of the sinogram are selected. These maxima are obtained for values of ρ_k which are proportional to all offset values for a given orientation.

2.3.2. Proposed method: MFBLP

The variable speed propagation scheme method [8, 9] enables the estimation of the offsets with a lower computational load than the extension of the Hough transform. We associate to this specific signal generation scheme a high-resolution method called "MFBLP" (modified forward backward linear prediction). In a previous work (see [17]), the concept of using forward-backward averaging led to effective results when it was applied to the SLIDE algorithm.

The basic idea in this method is to associate a propagation speed which is different for each line in the image. By setting artificially a propagation speed that linearly depends on the index of the lines in the matrix, we will be able to apply a frequency retrieval method to compute the offset values. When the first orientation value is considered, the signal received on sensor i ($i = 1, \dots, N$) is then

$$z(i) = \sum_{k=1}^{d_1} \exp(-j\tau x_{0k}) \exp(j\tau(i-1) \tan(\theta_1)) + n(i); \quad (13)$$

d_1 is the number of lines with angle θ_1 . When τ varies linearly as a function of the line index, the measure vector \mathbf{z} contains a modulated frequency term. Indeed, we set $\tau = \alpha(i-1)$.

$$z(i) = \sum_{k=1}^{d_1} \exp(-j\alpha(i-1)x_{0k}) \times \exp(j\alpha(i-1)^2 \tan(\theta_1)) + n(i). \quad (14)$$

This is a sum of d_1 signals that have a common quadratic phase term but different linear phase terms. The first treatment consists in obtaining an expression containing only linear terms. This goal is reached by dividing $z(i)$ by the nonzero term $a_i(\theta_1) = \exp(j\alpha(i-1)^2 \tan(\theta_1))$. We obtain then

$$w(i) = \sum_{k=1}^{d_1} \exp(-j\alpha(i-1)x_{0k}) + n'(i), \quad i = 1, \dots, N. \quad (15)$$

The resulting signal appears as a combination of d_1 sinusoids with frequencies :

$$f_k = \frac{\alpha x_{0k}}{2\pi}, \quad k = 1, \dots, d_1. \quad (16)$$

Consequently, the estimation of the offsets can be transposed to a frequency estimation problem. Estimation of frequencies

from sources having the same amplitude was considered in [15]. In the following a high-resolution algorithm, initially introduced in spectral analysis [15], is proposed for the estimation of the offsets.

After adopting our signal model we adapt to it the spectral analysis method called MFBLP [15] for estimating the offsets.

We consider d_k straight lines with given angle θ_k and apply the MFBLP method. We consider d_k straight lines with given angle θ_k and apply the MFBLP method to the vector \mathbf{w} . For a convenient representation the components of \mathbf{w} will be written $[w_1, w_2, \dots, w_N]$.

(1) For an N -data vector \mathbf{w} , form the matrix \mathbf{Q} of size $2 \cdot (N-L) \times L$, where the subscript "*" indicates conjugate:

$$\mathbf{Q} = \begin{bmatrix} w_L & w_{L-1} & \cdots & w_1 \\ w_{L+1} & w_L & \cdots & w_2 \\ w_{L+2} & w_{L+1} & \cdots & w_3 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ w_{N-1} & w_{N-2} & \cdots & w_{N-L} \\ w_2^* & w_3^* & \cdots & w_{L+1}^* \\ w_3^* & w_4^* & \cdots & w_{L+2}^* \\ w_4^* & w_5^* & \cdots & w_{L+3}^* \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ w_{N-L+1}^* & w_{N-L+2}^* & \cdots & w_N^* \end{bmatrix}. \quad (17)$$

Build the length $2 \cdot (N-L)$ vector:

$$\mathbf{h} = [w_{L+1}, w_{L+2}, \dots, w_N, w_1^*, w_2^*, \dots, w_{N-L}^*]^T; \quad (18)$$

L is such that

$$d_k \leq L \leq N - \frac{d_k}{2}. \quad (19)$$

(2) Calculate the singular value decomposition of \mathbf{Q} :

$$\mathbf{Q} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^H. \quad (20)$$

(3) Form the matrix $\mathbf{\Sigma}$ by setting to 0 the $L - d_k$ smallest singular values contained in $\mathbf{\Lambda}$:

$$\mathbf{\Sigma} = \text{diag} \{ \lambda_1, \lambda_2, \dots, \lambda_{d_k}, 0, \dots, 0, 0, 0 \}. \quad (21)$$

(4) Form the vector \mathbf{g} from the following matrix computation:

$$\mathbf{g} = [g_1, g_2, \dots, g_L]^T = -\mathbf{V} * \mathbf{\Sigma}' * \mathbf{U}^H \mathbf{h}, \quad (22)$$

where $\mathbf{\Sigma}'$ is the pseudoinverse of $\mathbf{\Sigma}$.

(5) Determine the roots of the polynomial function H , where

$$H(z) = 1 + g_1 z^{-1} + g_2 z^{-2} + \cdots + g_L z^{-L}. \quad (23)$$

(6) d_k zeros of H are located on the unit circle. The complex arguments of these zeros are the frequency values; according to (16), these frequency values are proportional to the offsets, the proportionality coefficient being $-\alpha$. The main advantage of this method comes from its low complexity. Indeed, the complexity of the variable parameter propagation scheme associated to MFBLP is much less than the complexity of the extension of the Hough transform as soon as the number of nonzero pixels in the image is increased. This algorithm enables the characterization of straight lines with same angle and different offset.

3. ESTIMATION OF NONRECTILINEAR CONTOURS IN AN IMAGE AS AN INVERSE PROBLEM

In the previous sections we will recalled a specific formalism for image representation and presented an application of the high-resolution methods of array processing to the retrieval of straight lines in an image.

In this section, we keep formalism retained for straight lines retrieval. The more general case of distorted contour estimation is proposed. As in the previous sections, this problem can be considered as an array processing problem in which a wave front has to be estimated. It is possible to make an analogy with a physical phenomenon that can be observed in wave physics. We suppose that the distorted curve actually contained in the image can be assimilated to a distorted wave front. Such a distorted front can be observed when the propagation medium is not isotropic. In order to estimate the wave front distortion, we propose to apply a recursive algorithm. In [20], a similar problem is solved, in the case of a plane wave received by a distorted antenna.

We propose a new method called ECAPMO (estimation of contours by array processing methods and optimization) for the estimation of continuous nonrectilinear contours. It relies on the formulation of an inverse problem over the generated signals and the determination of phase fluctuations.

3.1. Retrieval of a general phase model

The ECAPMO method relies on the idea of a continuous phase model. We propose to extend the formalism proposed in [6] that sets the analogy between the phase model used in array processing and a contour in image processing. Instead of assuming that the phase model is known, that is, that there exists a predefined model for the contour that we aim at retrieving, we create an artificial evolution of the wave front and of the corresponding received signal. By setting a recursive algorithm, we modify the phase of a current signal until it is equal to the input signal generated out of the image. The proposed ECAPMO method leads to the phase parameters characterizing the distorted wave front.

In order to retrieve the characteristics of the wave front corresponding to the distorted curve, we can start from an initial signal corresponding to a plane wave front. We will modify recursively the components of a current signal until it becomes equal to the signal actually generated out of the image.

3.2. Initialization of the proposed algorithm

Our recursive optimization algorithm needs to be initialized. For this purpose, we choose as initialization parameters the phase values corresponding to a plane wavefront. Through the signal generation formalism that we adopted, this plane wavefront corresponds to a straight line in the image. Therefore, in order to initialize our recursive algorithm, we apply the SLIDE algorithm, which is supposed to return the straight line that fits best the distorted contour which is present in the image. In this section we consider only the case where the estimated number d of curves is equal to one. The parameters angle and offset recovered by the straight line retrieval method are employed to build an initialization vector \mathbf{x}_0 , containing the position of the pixels of the initialization straight line:

$$\mathbf{x}_0 = [x_0, x_0 - \tan(\theta), \dots, x_0 - (N-1)\tan(\theta)]^T. \quad (24)$$

Figure 2 presents a distorted curve and also presents an initialization straight line that fits this distorted curve.

3.3. Distorted curve: proposed algorithm

We aim at determining the N unknowns $x(i), i = 1, \dots, N$ of the image, forming a vector $\mathbf{x}_{\text{input}}$, each of them taken into account respectively at the i th sensor:

$$z(i) = \exp(-j\mu x(i)), \quad \forall i = 1, \dots, N. \quad (25)$$

The observation vector is

$$\mathbf{z}_{\text{input}} = [\exp(j\varphi_1), \dots, \exp(j\varphi_N)]^T \quad (26)$$

with $\varphi_i = -\mu x(i)$ representing the phase of sensor i . So we try to recreate the signal from which we ignore the N parameters. We start from the initialization vector \mathbf{x}_0 , characterizing a straight line that fits a locally rectilinear portion of the curve to be studied. Then, with k indexing the steps of this recursive algorithm, we aim at minimizing

$$J(\mathbf{x}_k) = \|\mathbf{z}_{\text{input}} - \mathbf{z}_{\text{estimated for } \mathbf{x}_k}\|^2, \quad (27)$$

where $\|\cdot\|$ represents the norm induced by the usual scalar product of \mathbb{C}^N . For this purpose we use gradient methods with fixed step type. The vectors of the series are obtained by the relation

$$\forall k \in \mathbb{N} : \mathbf{x}_{k+1} = \mathbf{x}_k - \lambda \nabla(J(\mathbf{x}_k)), \quad (28)$$

where $0 < \lambda < 1$ is the step for the descent. The recurrence loop is

$$\mathbf{x}_k \longrightarrow \mathbf{z}_{\text{estimated for } \mathbf{x}_k} \longrightarrow J(\mathbf{x}_k). \quad (29)$$

The gradient is estimated using finite differences. We stop when the gradient becomes lower than a threshold.

At this point, by minimizing the function J , we find the components of vector \mathbf{x} leading to the signal \mathbf{z} which is the closest to the input signal. Nevertheless, by employing the criterion of minimum square error between signals,

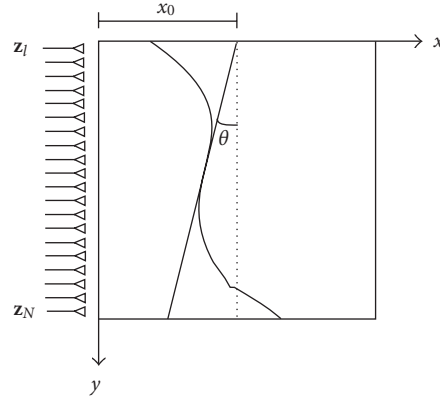


FIGURE 2: A model for an image containing a distorted curve.

a phase indetermination over the input signal is remaining. Therefore, we propose an algorithm that aims at canceling the phase indetermination induced by the criterion that we chose. This algorithm is based upon the continuity of the phase of the signal which is actually generated out of the image. Moreover we prove that the series $(\mathbf{x}_k)_{k \in \mathbb{N}}$ actually converges toward, a local minimum argument of the criterion J . By defining as \hat{x}_l the components of $\hat{\mathbf{x}}$ and starting from the relation

$$\begin{aligned} \forall p \in \mathbb{Z}, \quad \exp(j\varphi_l) &= \exp(-j\mu x_l) = \exp(-j\mu \hat{x}_l) \\ &= \exp\left(-j\mu\left(\hat{x}_l + \frac{2p\pi}{\mu}\right)\right), \end{aligned} \quad (30)$$

we deduce that there exists an N -uplet of relative integers denoted by p_l such that $\mathbf{x}_{\text{input}} = \hat{\mathbf{x}} + (2\pi/\mu)[p_1, p_2, \dots, p_N]^T$.

This relation is equivalent to a shift which is proportional to $2\pi/\mu$ between x_l and \hat{x}_l for each line l of the image, or to a phase delay appearing on the signals obtained on the lines of the image matrix. The general formulation of the N -components vectors which minimize the cost function J are defined by

$$\begin{aligned} \text{Arg min}_j &= \hat{\mathbf{x}} + \frac{2\pi}{\mu} \mathbb{Z}^N = \left(\hat{x}_1 + \frac{2\pi p_1}{\mu}, \dots, \hat{x}_N + \frac{2\pi p_N}{\mu}\right), \\ (p_1, \dots, p_N) &\in \mathbb{Z}^N. \end{aligned} \quad (31)$$

The choice of the descent method towards such a minimum $\hat{\mathbf{x}}$ is such that

$$\hat{\mathbf{x}} \in \text{Arg min}_y \{ \|y - \mathbf{x}_0\| \}, \quad (32)$$

where $\|\cdot\|$ symbolizes here the norm induced by the scalar product in \mathbb{R}^N , and $y \in \text{Arg min}_j$. This implies that the main characteristics of $\hat{\mathbf{x}}$ are

- (1) to minimize the criterion J ,
- (2) to guarantee that its distance to \mathbf{x}_0 , that is, $\|\hat{\mathbf{x}} - \mathbf{x}_0\|$ is minimum with respect to the distances of \mathbf{x}_0 to the other solutions \mathbf{x} of J .

The next step concerns the determination of the actual values of the vector $\mathbf{x}_{\text{input}}$. The uniqueness of the correct N -uplet for the reconstruction of the distorted wave requires determination of at least one of the components x_l of \mathbf{x} . At this stage of the method, the choice of an initialization by a convenient straight line and the interest of the work presented about the determination of the curves step in. The hypotheses of curve continuity is exploited for this purpose. A reconstruction method (going successively forward and backward over the lines) of the curve is proposed starting from a fixed point. Before going further, we choose as an arbitrary point i_{max} the maximum value of the set $\{i = 1, \dots, N\}$ such that $\hat{x}_i = x_0 - (i - 1) \tan \theta$, determined from the data \mathbf{x}_0 and $\hat{\mathbf{x}}$. We obtain

$$p_{i_{\text{max}}} = 0, \quad x_{i_{\text{max}}} = \hat{x}_{i_{\text{max}}}. \quad (33)$$

We set $\delta(\mu) = \text{Max}\{|\hat{x}_l - \hat{x}_{l-1}|, l = 2, \dots, N\}$. For the forward part over the remaining lines of the image matrix, for each line $l = i_{\text{max}}, \dots, N$, we determine successively $m_l \in \mathbb{N}$ such that $|\hat{x}_l - \hat{x}_{l-1} - (2\pi/\mu) \text{sign}(\hat{x}_l - \hat{x}_{l-1}) m_l| < \delta(\mu)$ and we set $p_{l-1} = \text{sign}(\hat{x}_l - \hat{x}_{l-1}) m_l$. For the descent method over the lines, we start anew for $l = i_{\text{max}} + 1, \dots, N - 1$ by increasing the index l for the two relations above.

3.4. Convergence of the gradient method

For $k \rightarrow +\infty$ the series \mathbf{x}_k converges towards a vector $\hat{\mathbf{x}}$ such that

$$\mathbf{z}_{\text{input}} = \mathbf{z}_{\hat{\mathbf{x}}}. \quad (34)$$

That is to say, $\hat{\mathbf{x}}$ is the argument a local minimum of J contained in the neighborhood (in the sense of topology) of \mathbf{x}_0 . Let us denote for all $k \in \mathbb{N}$, $R_k = \lambda \|\nabla J(\mathbf{x}_k)\|$. The order-one Taylor series of J over the R_k radius ball centered on \mathbf{x}_k allows us to write

$$\begin{aligned} \forall \omega \in \mathbb{R}^N, \quad |\omega| \leq R_k, \\ J(\mathbf{x}_k + \omega) = J(\mathbf{x}_k) + \langle \nabla J(\mathbf{x}_k), \omega \rangle + |\omega| \varepsilon(\omega). \end{aligned} \quad (35)$$

Thus for $\omega = -\lambda \nabla J(\mathbf{x}_k)$, with λ small enough so that ω have negligible norm, we obtain

$$J(\underbrace{\mathbf{x}_k - \lambda \nabla J(\mathbf{x}_k)}_{\mathbf{x}_{k+1}}) = J(\mathbf{x}_k) - \lambda \|\nabla J(\mathbf{x}_k)\|^2 \leq J(\mathbf{x}_k). \quad (36)$$

That is to say,

$$\forall k \geq 0, \quad J(\mathbf{x}_{k+1}) \leq J(\mathbf{x}_k). \quad (37)$$

The series $(\mathbf{x}_k)_{k \in \mathbb{N}}$ induces the decrease of J with

$$\lim_{k \rightarrow +\infty} \nabla J(\mathbf{x}_k) = \mathbf{0}. \quad (38)$$

This proves the convergence of the proposed optimization algorithm.

3.5. Summary of the proposed algorithm

An outline of the proposed distorted contour estimation method is given as follows:

- (1) derive artificial signals using (1);
- (2) apply SLIDE algorithm: estimate line angle and offset that fits best the distorted contour (see Sections 2.2 and 2.3);
- (3) initialize the ECAPMO method using the straight line parameters obtained after applying the straight line retrieval method;
- (4) estimate the fluctuations of the position of the pixels around the initialization straight line by using the gradient algorithm;
- (5) solve the phase indetermination problem, by using the hypothesis of continuity of the curve.

3.6. Numerical complexity of the method

We previously defined N_p as the number of nonzero pixels, and d as the number of straight lines. With given values of these parameters and of the image size parameter N , the order of magnitude of the complexity of the angle estimation method is $N_p + N \cdot (\sqrt{N} + d)$ [9, 14].

Concerning the algorithm of offset estimation, let us recall that L is a parameter chosen close to N , and d_k is the number of parallel lines with a given orientation index k . In practice L is the integer part of $(N - dk/2)$.

For signal generation, $7 \cdot N_p$ operations are needed to obtain the signal \mathbf{z} of (13). For each of the d orientations found through constant parameter propagation, the signal \mathbf{w} of (15) is obtained from the signal \mathbf{z} with $4 + 3 \cdot N$ operations. For the MFBLP method, we consider the case when one offset is expected for each orientation value. We chose for the parameter L the value $N - 1$. The procedure ‘‘roots’’ employed at step (5) in order to find the zeros of the polynomial function H is based on an eigen-decomposition of an $L \times L$ matrix. This eigen-decomposition dominates the

other operations realized by the MFBLP algorithm in terms of complexity. Thus the complexity of this dominant step is L^3 or equivalently $(N - 1)^3$. Therefore, the order of magnitude of the computational complexity of the offset determination algorithm is $7 \cdot N_p + d \cdot (4 + 3 \cdot N + (N - 1)^3)$.

The complexity dominating part of our algorithm for curve distortion estimation is the iterative algorithm. Let ‘‘Niter’’ be the number of iterations necessary for the convergence of the algorithm. We count the number of operations, neglecting the time required by the additions, and including one storage operation for the current value of the estimated vector \mathbf{x}_k of a current iteration k . Computing and storing the vector \mathbf{x}_0 from the parameters angle and offset given by the initialisation step requires $N + 1$ operations. For each iteration k , including one storage operation for each computed value of $\nabla J(\mathbf{x}_k)$, we obtain the following results.

(1) The numerical derivative $\nabla J(\mathbf{x}_k)$ requires the computation of two values of the function J , computed for vector \mathbf{x}_k and an incremented version of \mathbf{x}_k , and one division by the incremental vector. Then, $10 \cdot N$ operations are needed for the computation of the function J . The subtraction of two successive values of the function J and a division by the size N incremental vector needs N operations. So, $2 \cdot (10 \cdot N) + N + 1$ operations are needed for the computation and storage of the derivative $\nabla J(\mathbf{x}_k)$.

(2) N operations are needed for the multiplication of $\nabla J(\mathbf{x}_k)$ by λ , so the computation of \mathbf{x}_{k+1} from (28) and the storage of \mathbf{x}_{k+1} needs $N + 1$ operations.

So $N + 1$ operations are needed for the computation of \mathbf{x}_0 , and $22 \cdot N + 2$ operations for the computations of all iterations. In total, $N + 1 + \text{Niter} \cdot (22 \cdot N + 2)$ operations are needed. Some experimental results about the computational time required for the distorted curve retrieval method will be presented in the Section 5.

4. GENERALIZATION OF ECAPMO FOR THE ESTIMATION OF SEVERAL CURVES

In this section we consider the case where the estimated number of curves d is larger than one. We will suppose that each curve is composed of a single pixel per line in the image. Therefore the model for the input signal $\mathbf{z}_{\text{input}} = [z(1), z(2), \dots, z(N)]^T$ is

$$\begin{aligned} \mathbf{z} &= \begin{bmatrix} z(1) \\ z(2) \\ \vdots \\ z(N) \end{bmatrix} = \sum_{k=1}^d \begin{bmatrix} \exp(j\varphi_{1k}) \\ \exp(j\varphi_{2k}) \\ \vdots \\ \exp(j\varphi_{Nk}) \end{bmatrix} + \begin{bmatrix} n(1) \\ n(2) \\ \vdots \\ n(N) \end{bmatrix} \\ &= \begin{bmatrix} \exp(j\varphi_{11}) & \cdots & \exp(j\varphi_{1d}) \\ \exp(j\varphi_{21}) & \cdots & \exp(j\varphi_{2d}) \\ \vdots & \ddots & \vdots \\ \exp(j\varphi_{N1}) & \cdots & \exp(j\varphi_{Nd}) \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + \mathbf{n} \\ &= \mathbf{A}(\varphi)\mathbf{s} + \mathbf{n}, \end{aligned} \quad (39)$$

where $\mathbf{A}(\varphi)$ is a matrix of size $N \times d$ taken as a model for the matrix of the directional vectors of the sources and \mathbf{s} is the vector of sources amplitudes, all equal to 1.

So the term $\varphi_{ik} \in] - \pi, \pi]$ represents the phase of the transfer function of the system source k and sensor i having an amplitude equal to 1. The source vector related to source k is

$$\mathbf{a}(\varphi_k) = [\exp(j\varphi_{1k}), \exp(j\varphi_{2k}), \dots, \exp(j\varphi_{Nk})]^T. \quad (40)$$

Referring to matrix notations, we obtain thus

$$\mathbf{A}(\varphi) = [\mathbf{a}(\varphi_1), \mathbf{a}(\varphi_2), \dots, \mathbf{a}(\varphi_d)] \quad (41)$$

so that we can define an application written as \mathbf{z} , such that to every matrix variable

$$\varphi = [\varphi_1, \varphi_2, \dots, \varphi_d] = \begin{bmatrix} \varphi_{11} & \dots & \varphi_{1d} \\ \varphi_{21} & \dots & \varphi_{2d} \\ \vdots & \ddots & \vdots \\ \varphi_{N1} & \dots & \varphi_{Nd} \end{bmatrix} \quad (42)$$

we associate the vector $\mathbf{z}(\varphi)$ such that

$$\mathbf{z}(\varphi) = \sum_{k=1}^d \begin{bmatrix} \exp(j\varphi_{1k}) \\ \exp(j\varphi_{2k}) \\ \vdots \\ \exp(j\varphi_{Nk}) \end{bmatrix} = \mathbf{A}(\varphi)\mathbf{s}. \quad (43)$$

To all $\mathbf{z}(\varphi)$, we associate a real value written as $J \circ \mathbf{z}(\varphi)$. If we consider both J and \mathbf{z} as functions of a vector or matrix variable, \circ denotes composition between functions J and \mathbf{z} . The function which is obtained is applied to variable φ such that

$$J \circ \mathbf{z}(\varphi) = \|\mathbf{z}_{\text{input}} - \mathbf{z}(\varphi)\|^2. \quad (44)$$

From a numerical point of view we stack successively the columns φ_i of the matrix φ of size $N \times d$ in a vector ϕ of size $N \cdot d$ such that

$$\phi = \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \vdots \\ \varphi_d \end{bmatrix}. \quad (45)$$

We set $\tilde{\mathbf{z}}(\phi) = \mathbf{z}(\varphi)$ and we replace the previous problem by

$$\text{Minimize } \tilde{J}(\phi) = \|\mathbf{z}_{\text{input}} - \tilde{\mathbf{z}}(\phi)\|^2. \quad (46)$$

We initialize ϕ taking as column-vectors φ_i the vectors of the d straight lines obtained by the method for the case of the rectilinear contours \mathbf{x}_{0k} , $k = 1, \dots, d$. We use afterwards the gradient methods in order to estimate a vector $\hat{\phi}$ minimizing \tilde{J} . In the case when $d = 1$, we find anew the work presented in Section 3.

5. SIMULATIONS

This section falls into three parts dedicated to the efficiency of the use of high-resolution methods that we presented in this paper. The first part concerns the estimation of rectilinear curves. The second one concerns the estimation of distorted curves. In each part several examples are given in order to emphasize the potential of the high resolution methods for image processing.

5.1. Application in the case of rectilinear curves in binary images

As a first example, we propose an application of our method in the case of robotic vision. Figure 3(a) is a photograph taken by a camera and transmitted to the automatic command of a vehicle moving on the railway. This vehicle is used in particular for servicing of railways, that is, for the replacement of the parallel crosspieces. The vehicle, when moving along the railway, determines first the position of the rails from the obtained picture. Then, the position of the nearest crosspiece is detected. It places itself over the detected crosspiece and the replacement of this one is performed by an auxiliary engine. The iterative replacement of the crosspieces is realized step by step. First, the position of the rails is determined. The array processing methods of "SLIDE" and variable propagation scheme associated to MFBLP are employed. The result of this determination is presented in Figure 3(b). Referring to the retrieved position of the rails, the vehicle decides about the correction to give to its progression. Once the rails are retrieved, the image is processed once again. The localization of the first crosspiece is performed and presented in Figure 3(c). The crosspiece can be detected by changing the position of the antenna (this technique is described in [10]). The process is repeated and the crosspieces are retrieved iteratively. For this grey-level image, the computational time which is required to retrieve the two rails by means of SLIDE algorithm, when MFBLP method is associated to the variable speed generation scheme, is the following: the estimation of the angles needs 0.063 second, and the estimation of the offsets needs 1.1 seconds. As a comparison, the Extension of the Hough Transform, employed with the a priori knowledge of the angles, needs 47 seconds to find the offsets of the two lines that fit the rails.

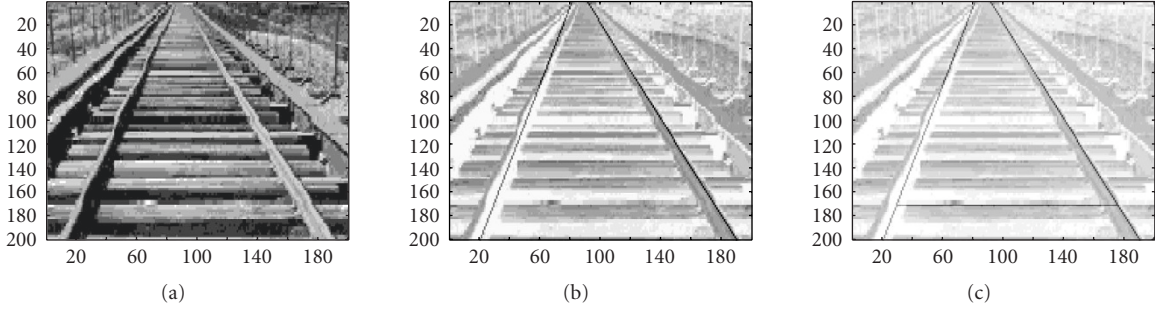


FIGURE 3: (a) Image transmitted to the automatic command of a vehicle that is moving on a railway for the servicing of the railways. (b) Detection of the rails for the progress of the vehicle. (c) Localization of the first crosspiece that the vehicle has to replace. The process is iterated crosspiece after crosspiece: photography, detection of the rails, and detection of the next crosspiece.

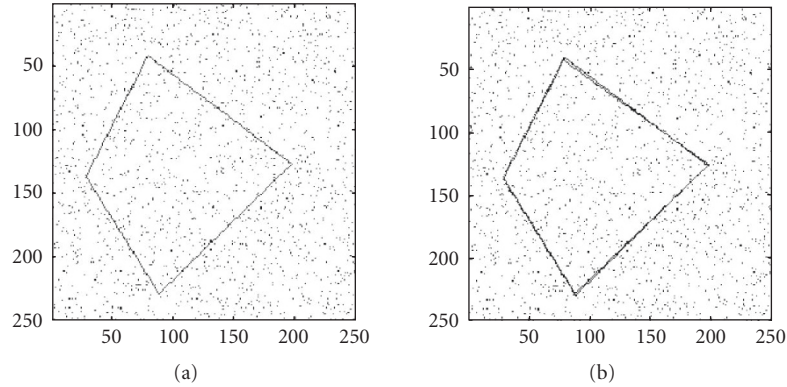


FIGURE 4: (a) Noisy image containing a convex polygon the summits of which we aim at detecting. (b) Superposition of the segments that fit the sides of the polygon, estimated by our method, and the original image.

We exemplify now the estimation of convex objects contours with polygonal geometry, as a second application case. Our method is employed to retrieve the characteristics of a polygon, namely, the number of sides and the coordinates of the summits. For this purpose, the straight lines that fit the sides of the polygon are determined by the method. The number and the parameters (angles, offsets) of the straight lines allow respectively to estimate the number of sides and the summits. The summits are fitted by considering the polygon as the smallest convex of the image, corresponding to the common intersection of the half-planes associated respectively to the support straight lines. Figure 4(a) presents the case of a polygon included in a noisy image. This image contains 15% of randomly distributed noisy pixels. The straight lines that fit the sides are given in Figure 4(b) and are determined in spite of the presence of noise in the image. For an image with more noisy pixels, a bias on the values of angle and offset can appear. It is difficult to obtain a valuable result from images with more than 20% of randomly distributed noisy pixels.

Figure 5 presents the result obtained on an image containing a set of roughly aligned points. Like images contain-

ing dashed lines, this kind of images leads to generated signals that are not continuous. Nevertheless the employed method manages to retrieve the main direction of the points of the image [21]. The image in Figure 5(a) contains a set of points. Figure 5(b) shows the result given by our line detection algorithm; in Figure 5(c) the superposition of the initial image and the result obtained shows that the overall orientation of these points is efficiently retrieved by the proposed method.

5.2. Simulations on nonrectilinear contours

This part is dedicated to the method employed in order to retrieve distorted curves. Several examples of use of ECAPMO are presented. Figure 6 presents a curve that we wish to determine. This distorted contour containing an almost straight section is a typical example of curve retrieved by the method ECAPMO. The different steps of the method are presented in Figure 7.

In the example in Figure 8, we chose a curve presenting some shift of the useful pixels of the curve at the beginning and at the end of its shape. The ECAPMO manages to return

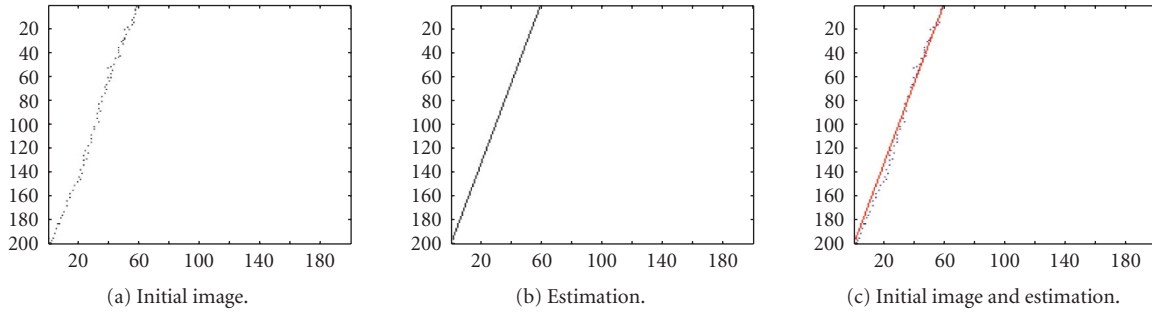


FIGURE 5: The main direction of a set of points.

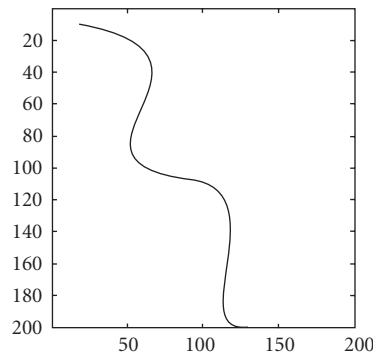


FIGURE 6: An image the contour of which exemplifies the distorted curves our method can cope with.

the shift values. In the example in Figure 9 are presented the results obtained with an image containing a single curve. For this 200×200 image, with a 3.0 Ghz pentium processor, the initialization needs the following computational times: the estimation of the angle takes 0.047 second, the estimation of the offset takes 0.66 second. As a comparison, when the extension of the Hough transform [2] is employed for the estimation of both angle and offset of the initialization straight line, the computational time is 8.54 second. The computational time required to run the iterative algorithm of the ECAPMO method is 0.80 second; 1800 iterations were necessary while solving the inverse problem in order to obtain this result. Figure 10 shows the results obtained in the case of a noisy image. This image contains 10% of randomly distributed noisy pixels. The curve is still efficiently retrieved. In Figure 11 appears a figure with two distorted curves. The specific method described in Section 4 is employed in this case. It manages to retrieve the two curves. Figure 12 shows that the method for distorted contour estimation copes with straight lines as well. The specific algorithm was applied in common to both distorted curve and straight line of the image. The slight bias on the offset value is canceled by the algorithm for distorted curves estimation.

Some practical situation was examined in Figure 13. This image symbolizes a vehicle and two road borders. The

algorithm for the estimation of all contours is the following.

- (i) The borders of the road are obtained through an initialization step: they are the two dominant directions in the image.
- (ii) Referring to the information obtained in step (1), the vehicle in the center is isolated. Then its contours are estimated.
- (iii) The algorithm dedicated to multiple-curve images is applied in order to estimate finely the borders of the road.

6. CONCLUSION

This paper handles the case of the contour retrieval in images. The formulation and resolution of rectilinear contour estimation can be transposed to a classical array processing technique. The rectilinear contours parameters appeared as real parameters of a source localization problem in array processing. In particular, we proposed the association of an array processing method and a frequency estimation method called MFBLP for the estimation of the offsets.

For the main point of the article, that is, estimation of distorted contours, we adopted the same conventions for signal generation. The work dedicated to rectilinear contours

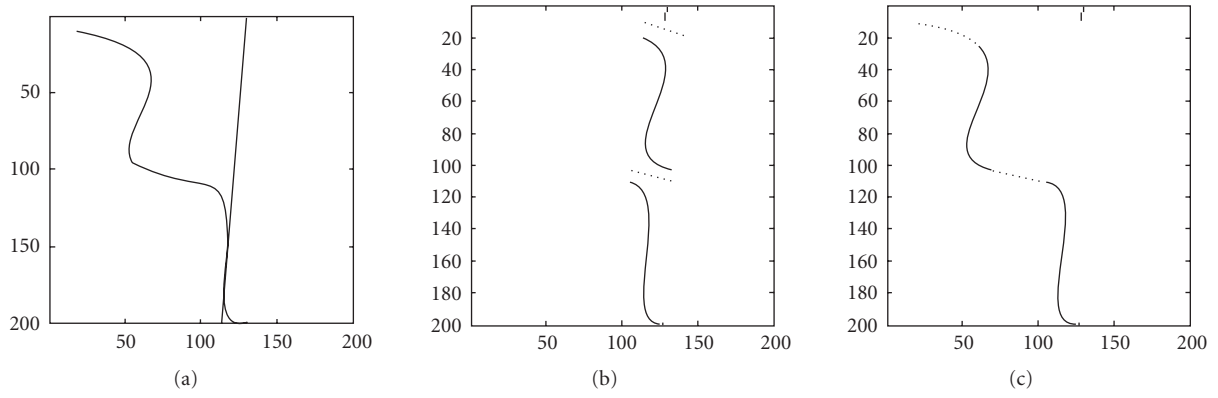


FIGURE 7: (a) Initialization of the method. (b) Determination of the vector $\hat{\mathbf{x}}$. (c) Junction of the different parts of the curve by determining the coefficients p_i .

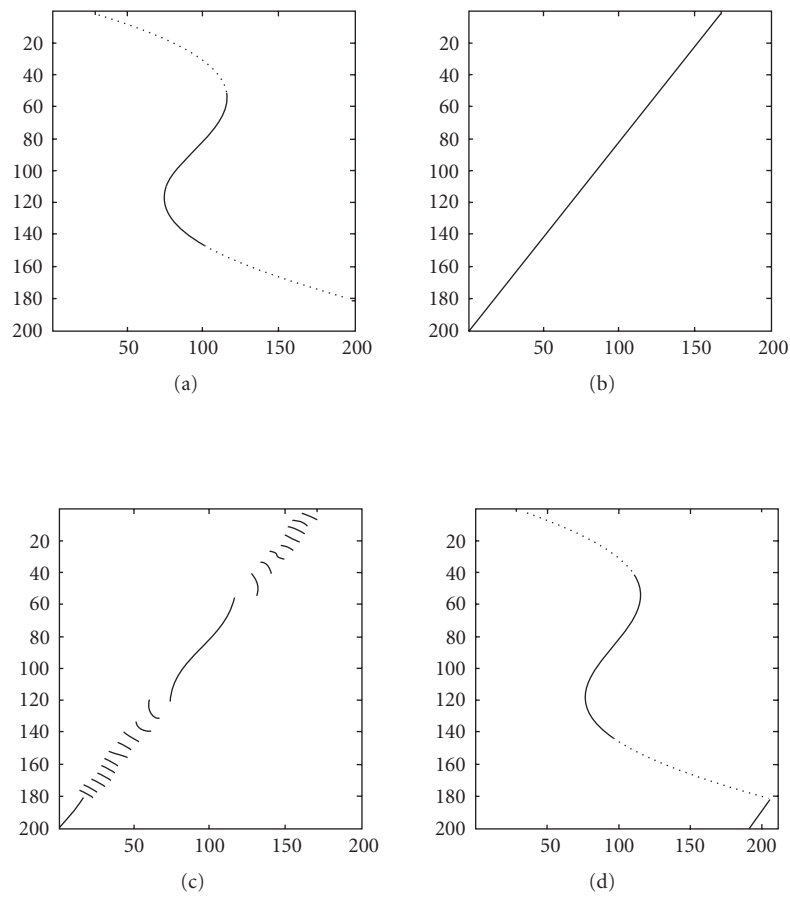


FIGURE 8: An image and the different steps of the method: original image, initialization, estimation obtained before sticking sections together, and final estimation. The method manages to return the shifts.

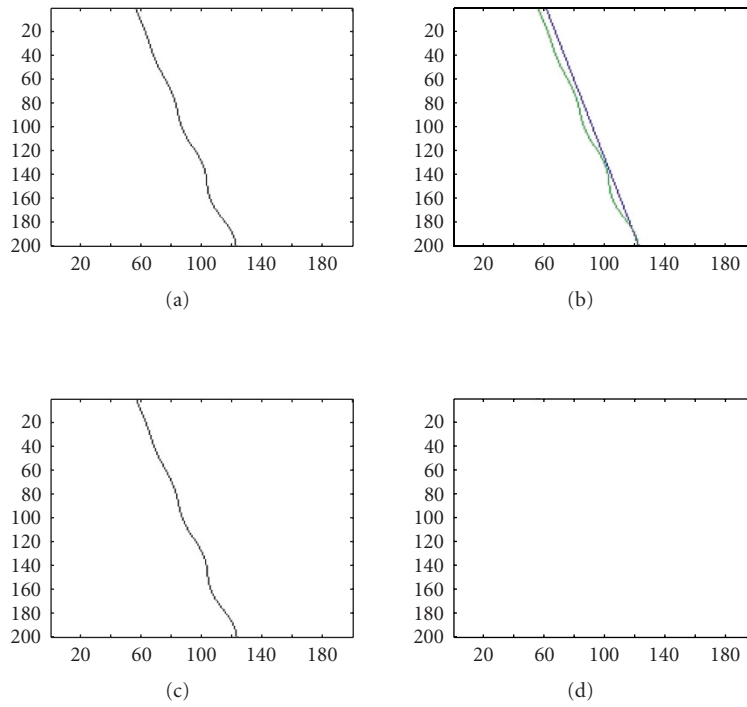


FIGURE 9: The main results obtained by distorted contours estimation: (a) image to be treated, (b) initialization, (c) estimation obtained by the proposed method, (d) difference between the initial image and the estimation.

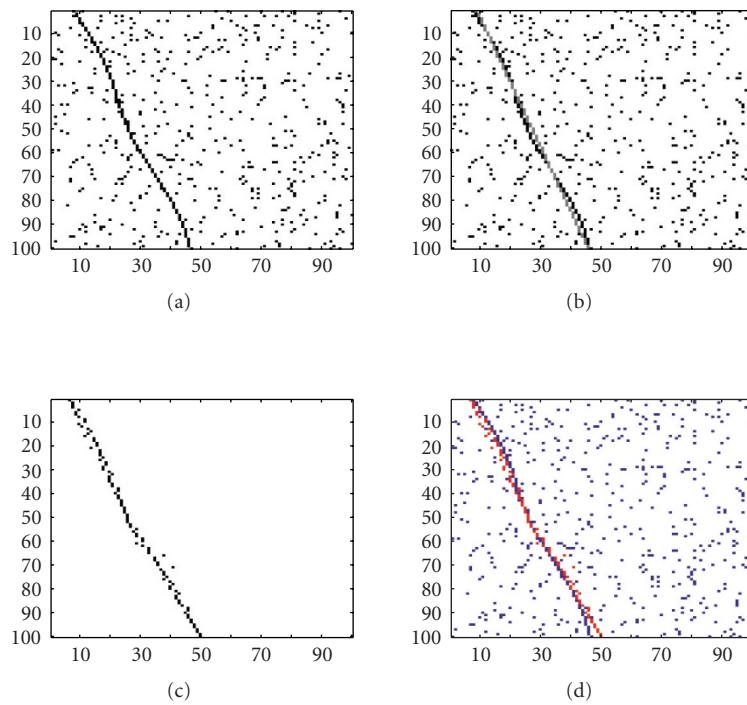


FIGURE 10: Distorted contours estimation on a noisy image: (a) image to be treated, (b) initialization, (c) estimation obtained by the proposed method, (d) superposition of the initial image and the estimation.

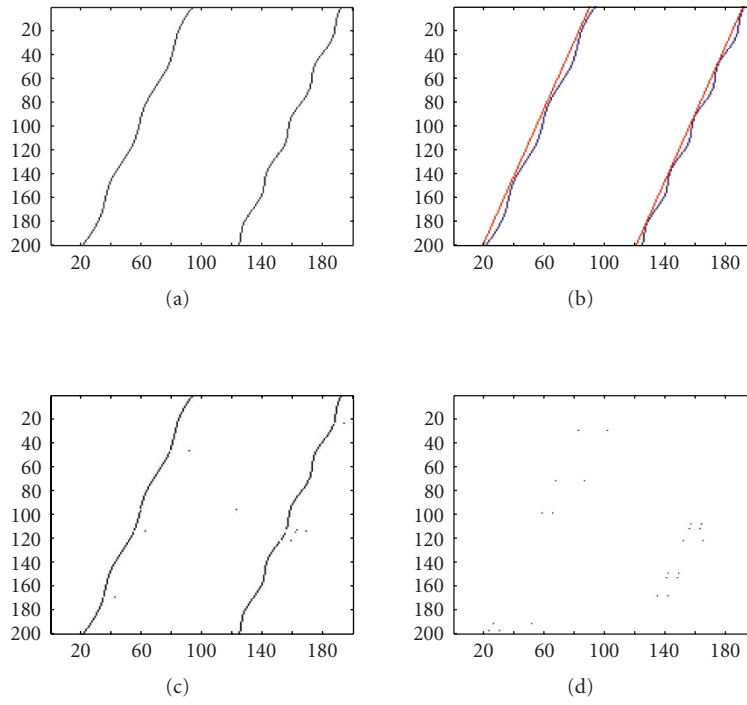


FIGURE 11: The case of an image containing two distorted curves: (a) image to be treated, (b) initialization, (c) estimation obtained by the proposed method, (d) difference between the initial image and the estimation.

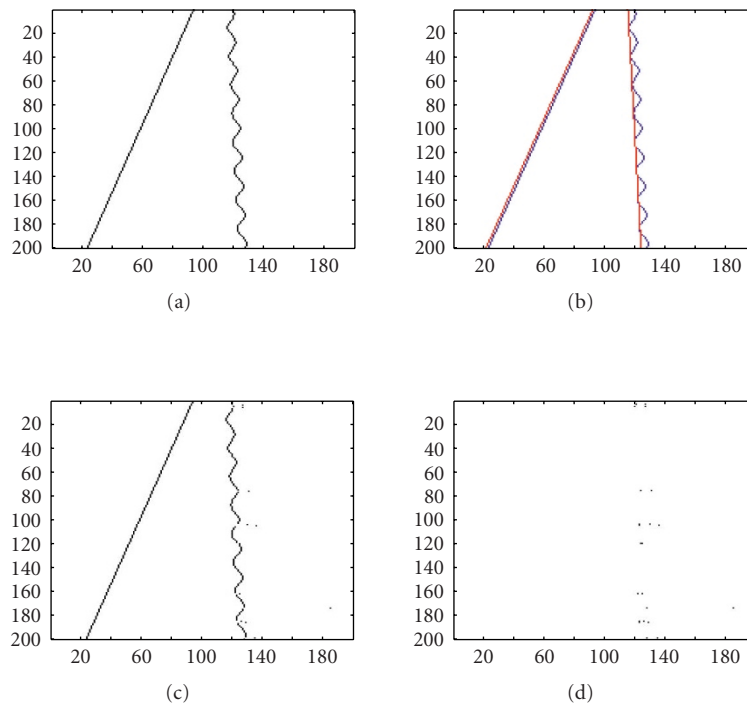


FIGURE 12: Results obtained by the method for several curves estimation One curve is a straight line: (a) image to be treated, (b) initialization, (c) estimation obtained by the proposed method, (d) difference between the initial image and the estimation.

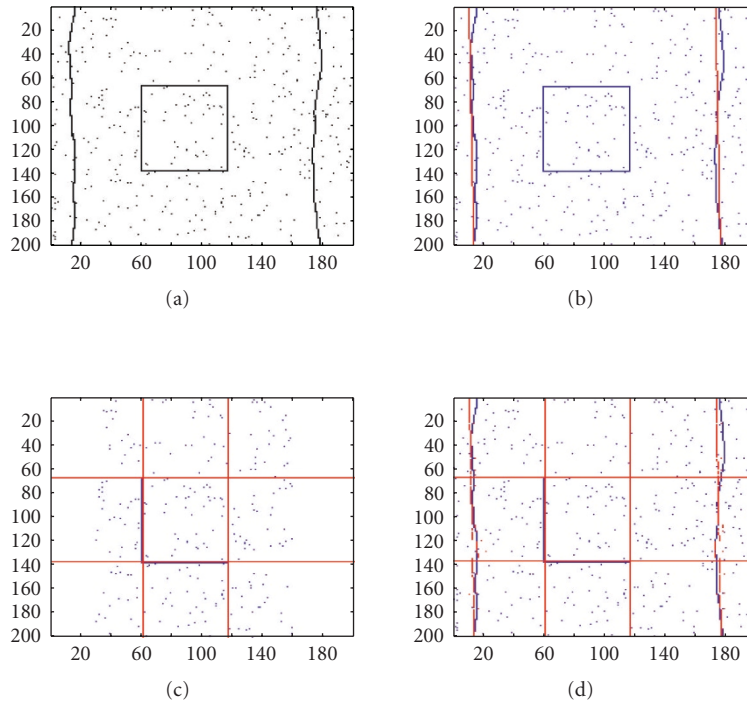


FIGURE 13: A practical situation is simulated. A vehicle and the two sides of the road are retrieved. (a) Initial image. (b) First estimation of the borders of the road. (c) Estimation of the sides of the vehicle (d). Refined estimation of the borders of the road.

was employed as an initialization step for an optimization technique. The contribution of high-resolution methods and the gradient method led to the elaboration of an efficient algorithm: ECAPMO. A recursive algorithm that is initialized by means of the straight line retrieval method estimates the fluctuations of a distorted curve around its main straight direction. Retrieval of straight and distorted curves was illustrated by numerical simulations showing the efficiency of the proposed methods. Thanks to the formalisms retained for the retrieval of general contours from image data, array processing and image processing got closer to each other.

ACKNOWLEDGMENT

We would like to thank the anonymous reviewers who contributed to the quality of this paper by providing helpful suggestions.

REFERENCES

- [1] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.
- [2] N. Kiryati and A. M. Bruckstein, "What's in a set of points? [straight line fitting]," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 4, pp. 496–500, 1992.
- [3] N. Guil, J. Villalba, and E. L. Zapata, "A fast Hough transform for segment detection," *IEEE Transactions on Image Processing*, vol. 5, no. 5, pp. 787–792, 1996.
- [4] H. K. Aghajan and T. Kailath, "A subspace fitting approach to super resolution multi-line fitting and straight edge detection," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '92)*, vol. 3, pp. 121–124, San Francisco, Calif, USA, March 1992.
- [5] G. Biennu and L. Kopp, "Optimality of high resolution array processing using the eigensystem approach," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 5, pp. 1235–1248, 1983.
- [6] H. K. Aghajan and T. Kailath, "Sensor array processing techniques for super resolution multi-line-fitting and straight edge detection," *IEEE Transactions on Image Processing*, vol. 2, no. 4, pp. 454–465, 1993.
- [7] S. Van Huffel and J. Vandewalle, "The total least squares technique: computation, properties and applications," in *SVD and Signal Processing: Algorithms and Architectures*, E. F. Deprettere, Ed., pp. 189–207, Elsevier, New York, NY, USA, 1988.
- [8] H. K. Aghajan and T. Kailath, "SLIDE: subspace-based line detection," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '93)*, vol. 5, pp. 89–92, Minneapolis, Minn, USA, April 1993.
- [9] H. K. Aghajan and T. Kailath, "SLIDE: subspace-based line detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 11, pp. 1057–1073, 1994.
- [10] H. K. Aghajan, *Subspace techniques for image understanding and computer vision*, Ph.D. thesis, Stanford University, Stanford, Calif, USA, 1995.
- [11] R. O. Schmidt, "Multiple emitter location and signal parameters estimation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 4, no. 3, pp. 276–280, 1983.

- [12] A. Paulraj and T. Kailath, "Eigenstructure methods for direction of arrival estimation in the presence of unknown noise fields," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 1, pp. 13–20, 1986.
- [13] R. Roy and T. Kailath, "ESPRIT: estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.
- [14] J. Sheinvald and N. Kiryati, "On the magic of SLIDE," *Machine Vision and Applications*, vol. 9, no. 5-6, pp. 251–261, 1997.
- [15] D. W. Tufts and R. Kumaresan, "Estimation of frequencies of multiple sinusoids: making linear prediction perform like maximum likelihood," *Proceedings of the IEEE*, vol. 70, no. 9, pp. 975–989, 1982.
- [16] R. T. Williams, S. Prasad, A. K. Mahalanabis, and L. H. Sibul, "An improved spatial smoothing technique for bearing estimation in a multipath environment," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 4, pp. 425–432, 1988.
- [17] B. Halder, H. K. Aghajan, and T. Kailath, "Propagation diversity enhancement to the subspace-based line detection algorithm," in *Nonlinear Image Processing VI*, vol. 2424 of *Proceedings of SPIE*, pp. 320–328, San Jose, Calif, USA, February 1995.
- [18] S. U. Pillai and B. H. Kwon, "Forward/backward spatial smoothing techniques for coherent signal identification," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 1, pp. 8–15, 1989.
- [19] N. Yuen and B. Friedlander, "DOA estimation in multipath: an approach using fourth-order cumulants," *IEEE Transactions on Signal Processing*, vol. 45, no. 5, pp. 1253–1263, 1997.
- [20] S. Bourennane and M. Frikel, "Localization of the wideband sources with estimation of the antenna shape," in *Proceedings of the 8th IEEE Signal Processing Workshop on Statistical Signal and Array Processing*, pp. 97–100, Corfu, Greece, June 1996.
- [21] S. Bourennane and J. Marot, "Line parameters estimation by array processing methods," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, vol. 4, pp. 965–968, Philadelphia, Pa, USA, March 2005.

Salah Bourennane received his Ph.D. degree from Institut National Polytechnique de Grenoble, France, in 1990, in signal processing. Currently, he is Full Professor at the École Généraliste d'Ingénieurs de Marseille, France. His research interests are in statistical signal processing, array processing, image processing, multidimensional signal processing, and performance analysis.



Julien Marot received the M.S. degree in physics engineering from ENSP Marseille, France, in 2003 and the Image Processing DEA in 2004. Since December 2004, he has been working as Ph.D. student in the Multidimensional Signal Processing Group (GSM), Fresnel Institute (CNRS UMR-6133). His research interests include applied image processing and signal processing.

