INT. J. REMOTE SENSING, 1998, VOL. 19, NO. 15, 2987–3002

SODAPRETER: a novel approach towards automatic SODAR data interpretation

A. K. DE, D. P. MUKHERJEE, P. PAL and J. DAS

Electronics and Communication Sciences Unit, Indian Statistical Institute, 203 BT Road, Calcutta-35, India

(*Received 26 September 1994; in final form 7 January 1998*)

Abstract. Instead of piecemeal approaches for detecting specific patterns in the SODAR echograms, an integrated modular approach 'towards' automatic interpretation of the ABL structure patterns, as depicted in the SODAR facsimile records, is presented. Here we propose a unified approach where the user is at liberty to select a wide range of image processing and the pattern recognition techniques required to extract remotely-probed meteorological information from the closest geometric representation of the SODAR pattern boundary. The ultimate goal, part of which is already implemented and reported here, is to generate 'expert-like' interpretation of SODAR echograms.

1. Introduction

SODAR (abbreviated form of SOnic Detection And Ranging) is an instrument that remotely monitors dynamic processes in the atmospheric boundary layer (ABL) within 1 km height vertically upwards from the Earth's surface. Operating on monostatic mode it radiates short bursts of high frequency sound energy which is scattered back due to fluctuations in the sonic refractive index within the boundary layer. The back-scattered signals are received and displayed in a facsimile recorder as the height–time–intensity information in a three-dimensional plot. Since the sonic refractive index is a function of temperature, wind speed, and humidity (to some extent) of the atmosphere, the SODAR echo pattern provides valuable meteorological information on the structure of the ABL. Some of the common ABL structures are *Inversion*, *Flat Top*, *Spike*, *Depression*, *Bulge*, *Gradual decrement/increment*, *Plumes*, *No structure*, *Multi-elevated layer*, *Dot Echo*, etc. The meteorologists of this field by observing the characteristic shapes on the facsimile records (analog) can qualitatively predict some form of weather conditions.

Promising reports on computer analysis of SODAR facsimile record to data do exist. Foken *et al.* (1987) have done some pioneering work of classifying various convective and inversion type of sodar echopatterns into 2-digit code compatible with computer pattern recognition. Chaudhuri *et al.* (1992) have given some basic concepts on automatic recognition and classification of SODAR structure patterns. Tripathi *et al.* (1993) have separated the plume structures. De *et al.* (1994) have identified dot-echo structures and studied their nature. Tripathi *et al.* (1992) have also developed some algorithms for noise removal from the digitized SODAR patterns.

One of the major hurdles not widely addressed by many researchers is the huge amount of data generated through the round-the-clock operation of the SODAR system which is continuously monitoring the ABL. The aim of this research is to

detect efficiently the boundary of the ABL structure patterns, as we have observed from our experience that decisions taken regarding the shape of the patterns are considered primarily with their boundaries and not with the overall structures. The methodology developed in this paper has a two-fold advantage—firstly, the detection and geometric interpretation of the SODAR echogram boundary solves data storage problems to a large extent. In fact, the boundary as a *signature* of ABL dynamics stores the meteorological information in a more compact form. Secondly, the trends of the boundary transformed to a parameter space, can be used to generate inferences similar to an expert meteorologist. We also note that the detected boundary may appear because of the limitation in dynamic range and sensitivity of the sodar signal recording instrument†. However, given this limitation and since a potential boundary point represents abrupt variations in signal intensity, our aim is to detect these points which carry *significant* information. It can be shown that the trends of SODAR boundary may effectively be inferred from the geometric parameters of the best fitted curve, for example from the gradient of straight boundary segment or curvature of the circular boundary segment.

This paper presents a novel modular approach in developing the SODAR Data Inter-PRETER, SODAPRETER. The overall modular design is shown in figure 1. Each module is composed of a set of image processing and pattern recognition algorithm, giving user a host of alternatives. Keeping in mind the minimum user interaction, generality and robustness of the system to be developed, mostly conventional and time-tested image processing and pattern recognition algorithms are used. It has to be noted that quite a few of the algorithms described are image dependent and the extent of noise cripped in during data collection and digitization process influences the result.

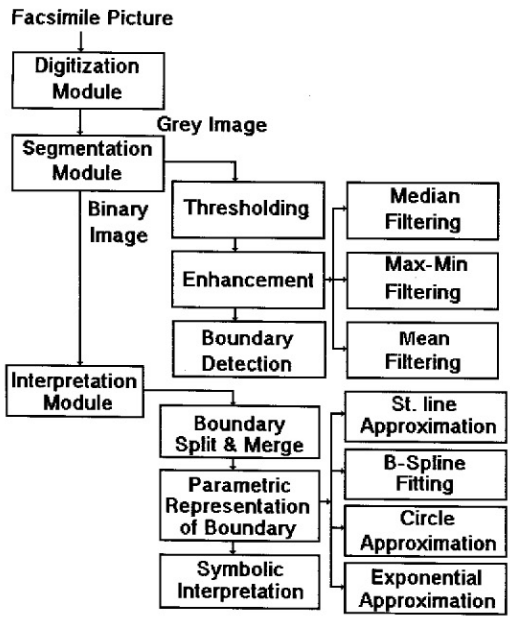

Figure 1. Overall block diagram for SODAPRETER system.

2. SODAR echogram segmentation

The echogram patterns used in this paper are collected from the SODAR instrument, indigenously developed by the scientists of the Indian Statistical Institute, Calcutta in collaboration with National Physical Laboratory, New Delhi. The SODAR system installed in this Institute has the following specifications as shown in table 1.

A typical SODAR echogram pattern is shown in figure 2. Grey value histogram of figure 2, representing the frequency of occurrence of individual grey value of the entire image, is shown in figure 3. Note that the grey scale ranges between 0 to 255.

The extent of noise in the SODAR facsimile image along with the *fuzziness* of the demarcation between SODAR information to the background is the critical issue and proper segmentation is needed. Before explaining the different types of segmentation approach, we mention that for the current implementation, the digitization of SODAR echo patterns on chart paper is done with a TRUVEL scanner under constant magnification and uniform lighting condition. We are in the process of building a software interface which can accept the SODAR echo signals directly into the PC thus eliminating digitization noise to a considerable extent. Following is the details of segmentation and various noise removal algorithms developed to provide in the software package. It is noted here that for the set of SODAR patterns we have worked with, segmentation followed by noise removal provide better result though the reverse methodology could be adopted depending upon the image quality. One such example of image filtering followed by segmentation is provided in the next section.

2.1. *Segmentation*

Segmentation is based on one of the two basic properties of grey level values—similarity and discontinuity. The principal approach we have adopted here depends on thresholding technique following Otsu's method (Otsu 1979). Based on inter and intra-class variances, the optimum threshold value is evaluated and used to divide the entire grey levels of image into two classes: information and background. The *goodness* of the threshold at level k is determined based on the following discriminant criterion:

$$\lambda = \sigma_B^2/\sigma_W^2 \quad \kappa = \sigma_T^2/\sigma_W^2 \quad \text{and} \quad \eta = \sigma_B^2/\sigma_T^2 \tag{1}$$

where σ_W^2, σ_B^2 and σ_T^2 are intra-class, inter-class and the total variances of grey levels respectively. For detailed calculation the interested reader may refer to Otsu (1979).

Table 1. Specification of ISI SODAR.

Transmitting frequencies	1–3 KHz (tunable, operating at 2·34 KHz)
Pulse width	10–100 ms
Resolution	1·7–17·0 m height
Transmitted power	100° W (electrical), 15° W (acoustical)
	6 ft parabolic fibre glass dish with transducer and horn fitted within an iron shield pasted with foam
Preamplifier sensitivity	169° dbw
Preamplifier gain	110 db
Receiver sensitivity	110 dbm
Filter bandwidth	$Q = 100$
Display	Facsimile recorder

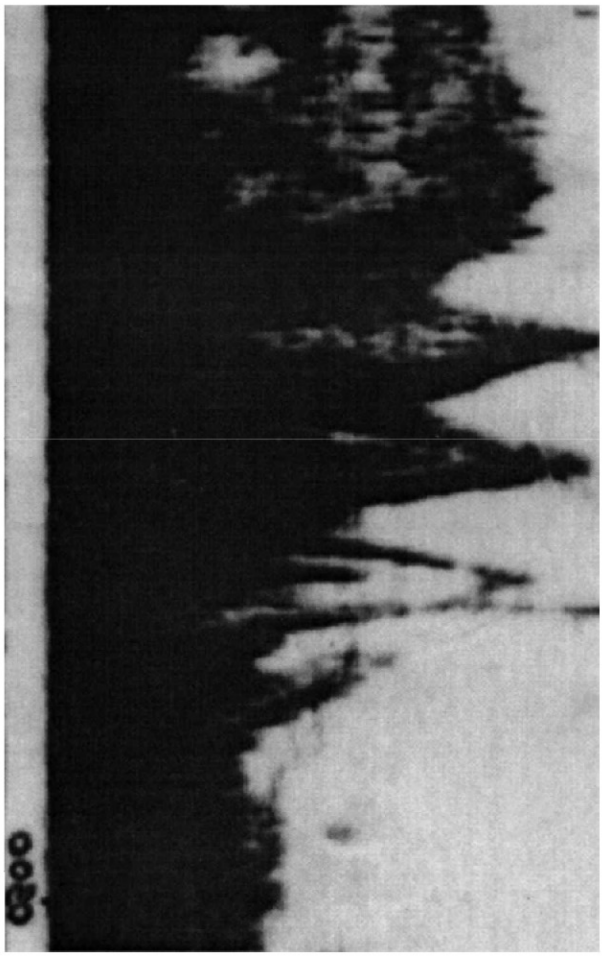

Figure 2. Example of typical SODAR pattern.

The problem of searching the optimum threshold is reduced to an optimization problem to find *best* k which maximizes these discriminant functions. In fact, it can be shown that η is the simplest and appropriate measure to evaluate the *goodness* or separability of the threshold at level k. Segmentation of the SODAR pattern of figure 2 after median filtering is shown in figure 4. Note that the threshold value calculated using Otsu's method is 114.

Therefore, segmentation at level k generates a bi-level picture from a grey level image. The information pixel is represented as 1 (say, grey value $\geqslant k$ in grey level image) whereas the remaining pixels (grey value $<k$) are 0 denoting background. This generates a simple intuitive notion for boundary detection: a potential boundary point is the pixel where the previous pixel value is different from the current one. This is implemented in the next section.

2.2. *Noise removal*

The major aim at this stage is to remove the image noise as much as possible. A set of time tested and conventional filtering algorithms give desired results as claimed

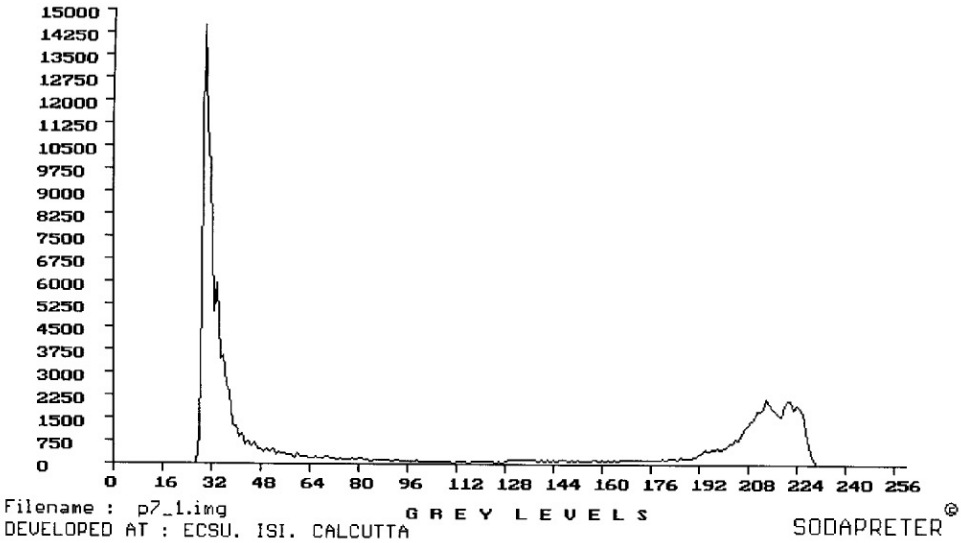

Figure 3. Grey value histogram of figure 2, representing the frequency of occurrence of individual grey value, is shown.

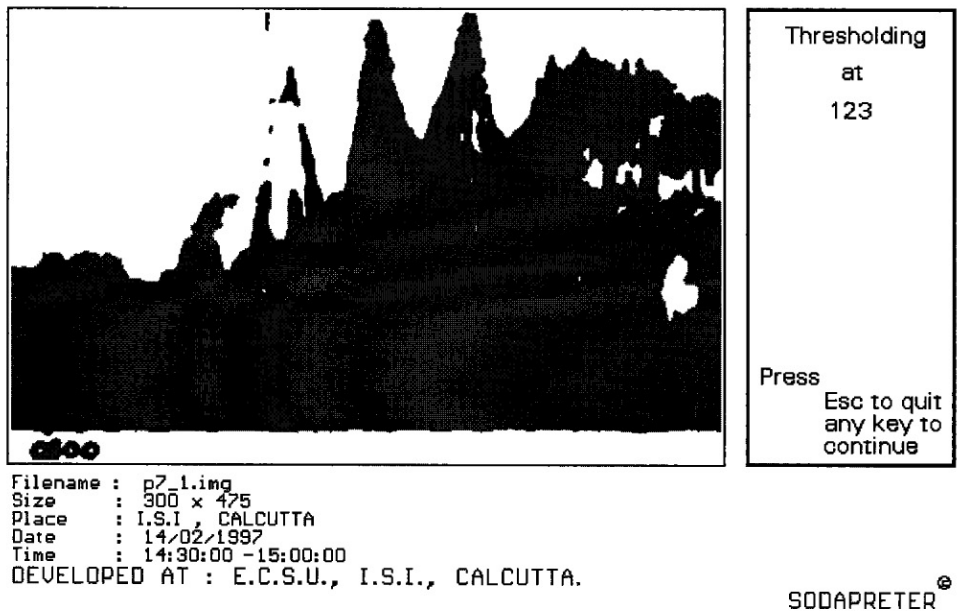

Figure 4. Segmented SODAR pattern of figure 2 after thresholding (Otsu 1979) at grey value 114.

in the introduction. Following are the details of filtering techniques provided in the software package:

2.2.1. *Max–min filtering*

The image is smoothed in two steps. Given an image matrix $r \times c$ with r rows and c columns, $\mathbf{f}(x, y)$ is smoothed to an image $\mathbf{g}_{\max}(x, y)$, whose grey level at every

point (x, y) is the maximum grey level values of the pixels of $\mathbf{f}$ contained in a pre-defined neighbourhood of (x, y). In the second step, $\mathbf{g}_{\max}(x, y)$ is smoothed to an image $\mathbf{g}_{\min}(x, y)$ whose grey level at every point (x, y) is the minimum grey level values of the pixels of $\mathbf{g}_{\max}$ contained in a pre-defined neighbourhood of (x, y). Obviously, $\mathbf{g}_{\min}(x, y)$ gives the final smoothed image of $\mathbf{f}(x, y)$.

2.2.2. *Neighbourhood filtering*

Given a $r \times c$ image of $\mathbf{f}(x, y)$ the procedure is to generate a smoothed image $\mathbf{g}(x, y)$ whose grey level at every point (x, y) is obtained by averaging the grey level values of the pixels of $\mathbf{f}$ contained in a pre-defined neighbourhood of (x, y).

2.2.3. *Median filtering*

For a $r \times c$ image of $\mathbf{f}(x, y)$ is smoothed to image $\mathbf{g}(x, y)$ whose grey level at every point (x, y) is obtained by the median value of the grey level values of the pixels of $\mathbf{f}$ contained in a pre-defined neighbourhood of (x, y).

2.2.4. *Homo-morphic filtering*

For a given image $\mathbf{f}(x, y)$ the algorithmic steps are:

1. calculate $\mathbf{f}'$ where $\mathbf{f}' = \log(\mathbf{f})$,
2. scale $\mathbf{f}'$ appropriately,
3. compute $\mathbf{I} = \mathrm{FFT}(\mathbf{f}')$, $\mathbf{I}$ is a 2D matrix to store the transformed image,
4. filter frequency domain image $\mathbf{I}$ with user selected cut-off frequency,
5. compute $\mathbf{f}'' = \mathrm{IFFT}(\mathbf{I})$,
6. take exponent of individual pixel values of $\mathbf{f}''$ and scale the image within 0–255 grey scale range.

Interested readers may refer to Das *et al.* (1994) for sodar image processing results after median, max–min, neighbourhood and homomorphic filtering. A comparative study of results of different filtering approaches also with varying filtering mask sizes are also given in Das *et al.* (1994).

3. Boundary interpretation

Having segmented the SODAR imageries, we now present methodologies to interpret the SODAR pattern boundary. As the name of the section suggests, it involves the detection of boundary and its best representation to a geometric form. In this section, we are dealing with binary images (with only two different pixel values representing information and background) vis-à-vis the grey level images which are segmented in the last section.

The SODAR facsimile record provides valuable meteorological information on the structure of the boundary layer. Our intention is to find the best geometric representation of the boundary, the parameters of which will provide the necessary information towards automatic boundary interpretation. The steps involved in automatic boundary interpretation are:

1. Boundary detection
2. Boundary approximation
3. Symbolic interpretation

3.1. *Boundary detection*

It is primarily based on the abrupt changes in grey values. Starting from the top-left corner, the image is scanned row-wise and a potential boundary point is detected

if the current pixel value is not the same as the immediately preceding one. If p_i gives the current (ith) pixel value of a $r \times c$ segmented image and $\{B\}$ gives the set containing potential boundary points, the pseudo-code for boundary detection is as follows:

```
for (row = 0 to row < r}
    for (column = 0 to column < c)
        if p_i ≠ p_i+1
            {B} ← p_i
                next column
                    next row
```

Again, the same image is scanned column-wise starting from top-left corner of the image matrix repeating the earlier steps. This gives the entire set of boundary points $\{B\}$ which we use in the next step for geometric approximation. The flowchart for boundary detection methodology is provided in figure 5. An additional boundary filtering steps are incorporated to clean stray pixels. If, in a pre-defined neighbourhood, the number of pixels is less than a certain threshold, those pixels in the neighbourhood are deleted. The result of using this boundary detection algorithm on the segmented and noise-filtered SODAR pattern of figure 4 is shown in figure 6.

The problem of boundary detection could just as well be posed as an edge detection problem between two regions with relatively distinct grey level properties. The basic idea underlying most edge detection techniques is the computation of a

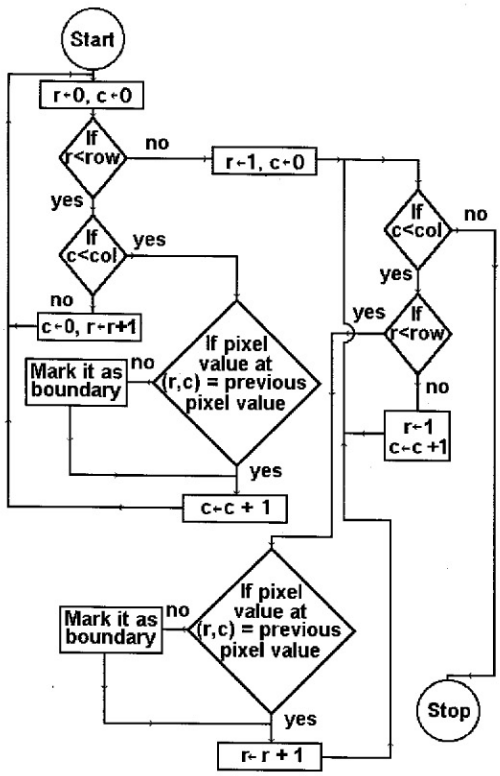

Figure 5. Flowchart for boundary detection methodology.

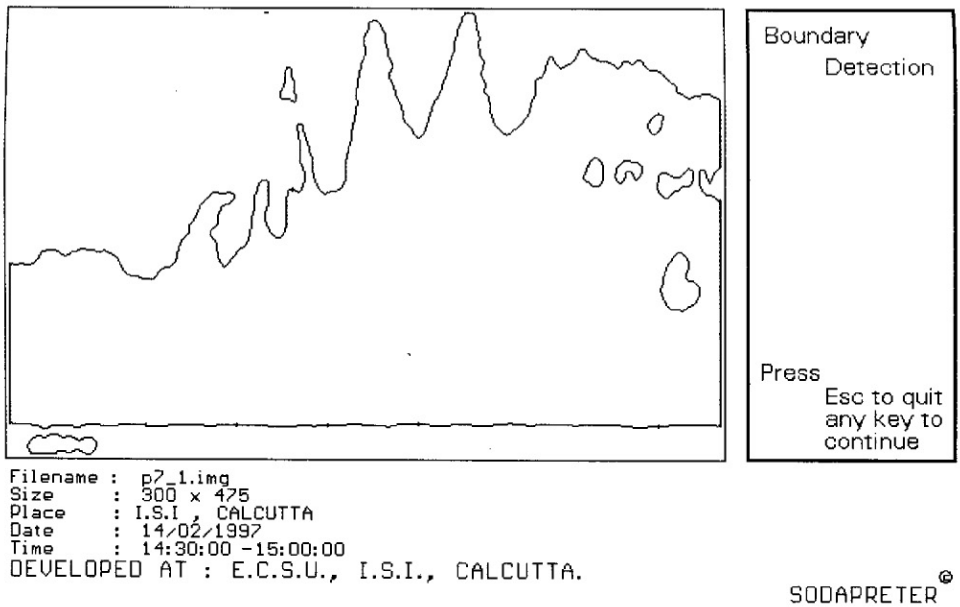

Figure 6. Result of boundary detection methodology on figure 4 is shown.

local derivative operator. Also this gradient vector operator points towards maximum rate of change of the function in a given location. First derivatives are implemented using gradient operators where the gradient of an image $\mathbf{f}(x, y)$ at location (x, y) is the vector

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\delta f}{\delta x} \\ \frac{\delta f}{\delta y} \end{bmatrix} \tag{2}$$

G_x and G_y are the image intensity gradients or differentials along x and y directions respectively. In digital image space, this is implemented by convolving following 3×3 Sobel masks as in Gonzalez and Wintz (1987) with the image,

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad G_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{3}$$

In case of edge detection, magnitude of Δf which is given by

$$\nabla f = \text{mag}(\nabla f) = [G_x^2 + G_y^2]^{1/2} \tag{4}$$

is calculated along with the direction angle with the vector ∇f at (x, y) as

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_x}{G_y}\right) \tag{5}$$

Gradient operation is followed by non-maximal suppression and edge linking (based on edge response $\mathbf{G}$). By non-maximal suppression, we mean suppression of *weak*

edge points generated due to noise pixels. These are checked from the edge intensity value of an edge after comparing it with its local 8-neighbourhood. In case of broken edges, the *gap* between two potential edge points is linked by interpolation depending on the edge responses of the connecting pixels. Results of Sobel edge detection followed by non-maximal suppression and edge linking is given in Das *et al.* (1994).

The important and crucial step is the tracing of boundary points and recording the co-ordinates of each point. Note that, the tracing should be done in a particular order—clockwise or anti-clockwise and also segment by segment. Segment-wise scanning is necessary to approximate each segment to the closest geometric form. The tracing is implemented using a recursive routine taking the co-ordinate of the top-left corner of the image (0,0). The recursive routine TRACE searches for a boundary point starting from the position G of a 3 by 3 mask (shown below) and moving clockwise, the position of the current pixel being E, the centre of the mask. The pixel positions are shown below. The mask is immediately shifted from E to the next boundary point if the routine finds a boundary point in the 8-neighbourhood, followed by another recursive call of the routine TRACE. This iterative call to routine TRACE continues till all the boundary points are traced. The co-ordinates of each boundary point (with respect to the origin at top-left corner of the image) is recorded in a separate file.

A B C
D E F
G H I

If no pixel is found in the 8-neighbourhood, the mask size is increased to 5 by 5 and the clockwise scanning within the mask is done in the same fashion. If there is no potential boundary pixel inside the 5 by 5 mask (except the central pixel), a break is considered in the segment and tracing for a new segment starts. A threshold for number of boundary points in a segment is also provided in order to eliminate *insignificant* segments.

As claimed in the introduction, the representation of SODAR data through its boundary envelope gives the unique advantage of data compression. The following analysis in table 2 shows a comparative study of bit-wise storage space saving for SODAR pictures.

3.2. *Boundary approximation*

Linear, circular, exponential and B-spline representations are implemented. Before explaining the representation scheme, we detail the detection of *control points* of the SODAR pattern boundary. The segment between two consecutive control points are approximated by the above mentioned geometric forms. Following Haralick and Shapiro (1992), an iterative endpoint fit-and-split procedure is implemented.

Table 2. Bit-wise storage space saving after boundary detection.

Image	Size	kByte required for storing SODAR data (approx.) as normal image file	kByte required for storing SODAR data boundary (approx.) only	Bit-wise saving
Figure 2	300 × 475	96	15	6:1

3.2.1. *Linear approximation*

Let a boundary segment be given by

$$L = \left[\left(\frac{x}{y} \right) \alpha x + \beta y + \gamma = 0 \right] \quad \text{where, } \alpha^2 + \beta^2 = 1 \tag{6}$$

with endpoints at (x_1, y_1) and (x_n, y_n). For any point (x_r, y_r), the distance between L and (x_r, y_r) is given by,

$$d_r = |\alpha x_r + \beta y_r + \gamma| \tag{7}$$

If d_r is greater than a user defined distance threshold $d_{\max}$, we take (x_r, y_r) to be a break point or *control point*. Else, the point set within the boundary (x_1, y_1) and (x_n, y_n) is approximated to a straight line. Taking the most general representation $\hat{\alpha}\hat{x} + \beta\hat{y} + \hat{\gamma} = 0$ under the constraint that $\hat{\alpha}^2 + \beta^2 = 1$, and minimizing the least squared error ε^2 (Haralick and Shapiro 1992) for N points using the Lagrange multiplier form,

$$\varepsilon^2 = \sum_{n=1}^{N} (\hat{\alpha}\hat{x}_n + \beta\hat{y}_n + \hat{\gamma})^2 - \lambda(\hat{\alpha}^2 + \beta^2 - 1)N \tag{8}$$

Note that, we assume

$$\hat{x}_n = x_n + \xi_n, \quad \text{and} \quad \hat{y}_n = y_n + \eta_n$$

where $(\hat{x}_n, \hat{y}_n)$ is the noisy observed value for a point (x_n, y_n) added with random noise (ξ_n, η_n), distributed independently and identically. Evaluating partial derivatives and solving

$$\frac{\partial \varepsilon^2}{\partial \hat{\gamma}} = 0, \quad \frac{\partial \varepsilon^2}{\partial \hat{\alpha}} = 0, \quad \text{and} \quad \frac{\partial \varepsilon^2}{\partial \beta} = 0$$

$\hat{\alpha}$, β and $\hat{\gamma}$ are evaluated.

The boundary segment denoted by two consecutive *control points* are approximated by different geometric forms. The best fit, both visually and with minimum *error of fit*, are chosen for the particular segment. The other geometrical form approximations are detailed below.

3.2.2. *Circular approximation*

In the case of a circular arc, an exact function to be minimized can be worked out. For a circular arc centered at (a, b) with radius R, following same notations as in the last paragraph, the error to be minimized are

$$\varepsilon^2 = \sum_{n=1}^{N} (\sqrt{(\hat{x}_n - a)^2 + (\hat{y}_n - b)^2} - R)^2 \tag{9}$$

The parameters (a, b, R) are evaluated setting $\nabla\varepsilon = 0$ where

$$\nabla\varepsilon = \left\{ \frac{\partial \varepsilon^2}{\partial \alpha}, \frac{\partial \varepsilon^2}{\partial \beta}, \frac{\partial \varepsilon^2}{\partial R} \right\}$$

These equations are minimized iteratively using gradient descent technique. The detail calculation and psuedocode is given in Haralick and Shapiro (1992).

3.2.3. *Exponential approximation*

A curve of exponential nature is modelled with the equation:

$$y = ae^{bx} + c \tag{10}$$

The equation (10) is linearized to $Y = A + bx$ taking logarithm on both the sides, where $Y = \log_e(y - c)$, $A = \log_e^a$. The methodology similar to linear fit is then used for the straight line derived from exponential relation in equation (10). The parameters derived after line fitting are transformed back to image space.

3.2.4. *B-spline approximation*

The B-spline curve segment between two successive *control points* $P_i(x_i, y_i)$ and $P_{i+1}(x_{i+1}, y_{i+1})$ are obtained following Ammeraal (1986) by computing the parametric representation of the curve segments $x(t)$ and $y(t)$ where t grows from 0 to 1.

$$\left.\begin{aligned} x(t) &= \{(a_3t + a_2)t + a_1\}t + a_0 \\ y(t) &= \{(b_3t + b_2)t + b_1\}t + b_0 \end{aligned}\right\} \tag{11}$$

The coefficients a_i could be obtained from the following relations:

$$a_3 = (-x_{i-1} + 3x_i - 3x_{i+1} + x_{i+2})/6$$

$$a_2 = (x_{i-1} - 2x_i + x_{i+1})/2$$

$$a_1 = (-x_{i-1} + x_{i+1})/2$$

$$a_0 = (x_{i-1} + 4x_i + x_{i+1})/6$$

Similarly, coefficients b_i can be evaluated from y_{i-1}, y_i, y_{i+1} and y_{i+2}. From the properties of B-splines and taking first and second order derivatives of $x(t)$ and $y(t)$, as in equation (11), it can be shown that this representation is continuous at the *control points* and smooth as well. We note that, approximating SODAR echogram pattern boundary and the various patterns associated with it, the smoothness of this parametric representation may not always be a desirable feature. We discuss this issue in the next section.

The control points on the SODAR data boundary following equations (6) and (7) are shown in figure 7. Note that the boundary segments between two successive control points are approximated to appropriate geometric segments. Results of B-spline and linear approximation to SODAR data boundary are shown in figures 8 and 9.

3.3. *Symbolic interpretation*

As mentioned in the introduction, we attempt to interpret the boundary from its best geometric interpretation. For current implementation, symbolic interpretation of linear curve segments are implemented. Based on the gradient of the straight lines we can interpret the nature of the boundary (i.e., about the type of rise and fall of the boundary). The interpretation is derived from the experiences and heuristics of the domain knowledge of an expert human meteorologist. The gradients of the straight lines are classified into certain pre-defined ranges and their corresponding 'human-like' interpretation are provided in table 3.

Examples of the interpretation of the SODAR boundary from their closest geometric approximations are shown in figure 9. The equivalent semantics of all the segments are also shown, the numeric values in the bracket being the gradient angles for a particular segment.

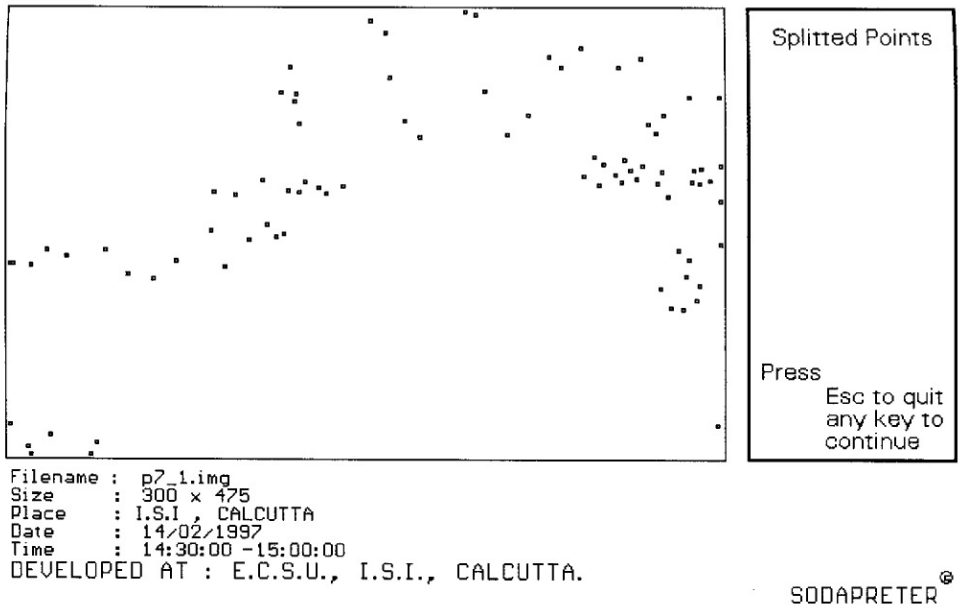

Figure 7. Control points (split points) detected in the SODAR data boundary following equations (6) and (7) are shown.

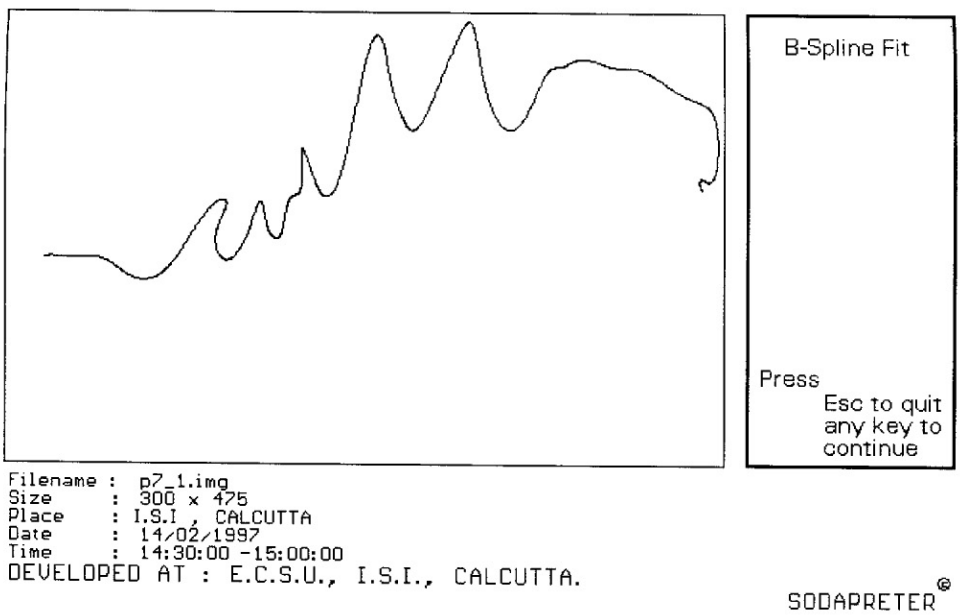

Figure 8. SODAR data boundary segments after B-spline fitting. Clearly, some boundary segments are not approximated as the error of fitting B-spline exceeds the threshold.

This methodology is extended to interpret multi-layer and plume signal structures in sodar images and the results are available in Das *et al.* (1994). For multilayer structure, individual signal column is scanned in the increasing order of height. In case of presence of a separate boundary layer at *significant* height apart from the

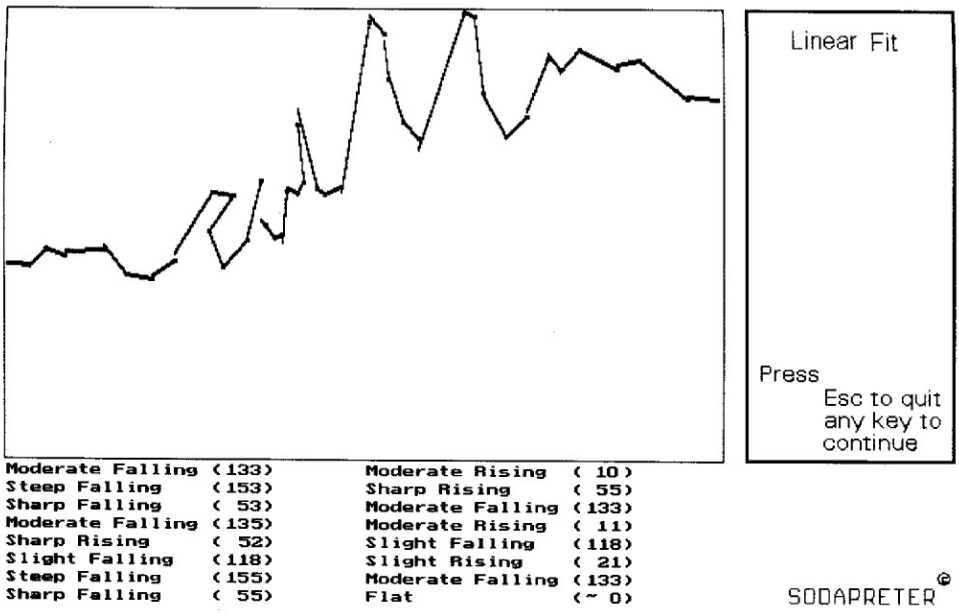

Figure 9. Boundary in figure 4, approximated using sequence of linear segments, is shown. The figures in the bracket indicate gradient angles of segments with equivalent semantics as detailed in section 3.3.

Table 3. Classified ranges of straight lines.

Interpretation	Gradient value (°)
Flat	−10 to +10
Slight rising	+10 to +30
Moderate rising	+30 to +50
Sharp rising	+50 to +70
Steep rising	+70 to +90
Slight falling	−10 to −30
Moderate falling	−30 to −50
Sharp falling	−50 to −70
Steep falling	−70 to −90

ground inversion layer, the pattern could be recognised as multilayer structure. In case of detection of plume structure, the magnitude of rise of inversion layer vis-à-vis the time of rise and fall is compared. This will indicate the steepness of the plume structure and the inversion breaking point. While elevated inversion layer is detected approximately 50 to 70 m above ground level for multilayer structure, in case of plume, its height ranges approximately between 300 to 490 meters with time span ranging 40 to 264 s approximately.

As shown in table 4 for a typical image of size 300 × 350 pixels, following is the list of representative processing time, including I/O, of individual operation of SODAPRETER. Hardware used is PC 486 DX with 66 MHz clock speed.

Table 4. Representative processing time (including I/O) of individual operations of SODAPRETER.

Operating		Processing time (s)
Display	Grey value image	Insignificant
	Threshold image	Insignificant
Filter	Neighbourhood	2
	Median	2
	Max–Min	3–5
Utility	Auto threshold calculator	Insignificant
	Boundary detection	Insignificant
	Histogram equalization	1
	Logarithmic stretching	1
	Crispening	3

3.4. *Performance evaluation*

We have considered the following approaches in the evaluation process of pattern detection/recognition in sodar images. The first and obvious technique is the subjective analysis. At least three experienced meteorologists are shown with original set of images and the corresponding processed images using SODAPRETER. Their responses indicate reasonably good performance of our proposed image processing technique. We also note that since the software is a user-oriented menu-driven package, the desired results could be achieved using any one (or combination) of many algorithms depending on the user's choice and on the type of SODAR pattern.

The second and more obvious approach is to test the output patterns against a set of test rules ('echo signatures') specific for each situation like plume, inversion, multi-layering, etc. For example, in the case of a plume, a steep rise followed by a steep fall over a time period (usually 5 to 10 minutes) is observed. The gradient during the rise is *almost* equal to the fall except the sign is reversed. Beside this local window (specific to a single plume), a larger window over a time period from half-an-hour to one hour is observed. The presence of such 'periodicity' in the larger window confirms the occurrence of plume structure. Detail rule base for other ABL patterns are dealt extensively in Chatterjee *et al.* (1996).

We are currently exploring to corroborate with balloon data recorded by the Indian Meteorological Department, Dumdum, in evaluation of SODAPRETER analysis.

4. Discussion

As a matter of fact storage of overall structure pattern requires a huge amount of memory space which can be reduced to an appreciable degree if we consider its boundary only. In contrast to several piecemeal approaches to detect and interpret the specific SODAR structure patterns, we present a unified approach towards automatic interpretation of SODAR data. We do not claim that the module, at this stage, is rich and exhaustive enough to tackle a wider variety of SODAR structure patterns. But, certainly with further upgradation, we expect to interpret *almost* all the possible ABL structure patterns recorded in the SODAR system. The images we have experimented with to date have hardly any segment close to the circular arc. For B-spline curve fitting, the approximated boundary is more smooth to be desired. But, the advantage is that every *control point* P_i depends on P_{i+1} and P_{i-1} as shown

in equation (11). This makes it ideal to interpret the continuous boundary rather than the discrete structures. We hope to find the trend of the boundary from the maxima and minima of the B-spline function (from the first and second derivatives). The exponential curve fitting seems to be the most promising rest aside the linear approximation. However, the exact form of equation (10) and the constants associated with it need to be properly investigated and thoroughly understood.

A wide range of flexibility is provided to the user in almost every module. This is particularly to tackle variations in the SODAR facsimile images so far as noise and digitization problems are concerned. Our attempt to detect the boundary solves the data storage problem to a large extent. What we think is a major contribution is an approach towards interpreting the boundary as its best approximated geometric form. It is certainly open up a new avenue, part of which is already implemented and reported here, where an expert system like the front-end may predict the different ABL patterns from the parameters of the best fitted curve of the SODAR boundary.

We are in the process of combining the geometric information, as an envelope of intensity values of the SODAR pattern boundary, with the important time and height information associated with it. For example, a sequence of 'Flat' boundary segments over an hour of observation should identify the atmospheric condition of the flat top signifying little variation of rate of change of energy on the specified height for the given time duration. Similar interpretations for other echo signatures are being developed, which will be reported in detail in a subsequent publication.

5. Future direction

Normal sodar patterns include various types of man-made/atmospheric noises and no single noise filtering technique will be able to clean all types of noises as treated in Tripathi *et al.* (1993). We have tried with various conventional filtering algorithm and found that the type of filter to be used depends upon particular class of imageries.

We are currently implementing homomorphic and the conventional filters like low, high and band pass filters which will give more flexibility to the user. Additionally, methodologies to enhance the image quality or *crispening* the grey values could be incorporated finding the Laplacian of the image (Gonzalez and Wintz 1987). The mathematics of exponential curve fitting is to be explored fully. The significance of the parameters for both the exponential and B-spline fitting in interpreting the various meteorological conditions needs to be completely understood.

It is obvious that 'visibility' of internal structure having adequate meteorological value may be possible within the SODAR echo patterns recorded in sodar subsystems with increased dynamic range and/or higher sensitivity value. Instead of finer structure our aim is to 'extract' finer details of the mixed/embedded ABL structure for a commonly-used sodar system by advanced IP techniques. Multithresholding of image will surely give more detailed information. In fact subplume (with greater energy content) within the convective plume structure may be identified just with the adjustment of threshold (Tripathi *et al.* 1992).

Even SODAPRETER can produce measurement of mixed layer height on a routine basis provided the 'definition' of it is standardised, as was mentioned by Beyrich (1993). As stated, the height of the elevated layer or a corresponding maxima secondary maxima in a digitised S-profile, proportional C_T^2, is its measure within

the sodar probing range (500–1000 m). But beyond the probing range, the maximum plume height is its representative which is obviously less than its true value.

The two specific problems to be taken care of are: firstly, the real time approach where mapping of SODAPRETER onto the parallel hardware may be a solution; secondly, statistical parameters is to be incorporated in the final phase of inference making. Those factors may be based on the previous data and past records which are available for last few years. This can also be combined with a *confidence factor* value which may be generated depending on the performances of IP/PR algorithms and the user feedback. Based on heuristics derived from expert meteorologist, a knowledge base could be created which may be used for the expert system mentioned above.

Acknowledgement

The authors gratefully acknowledge the financial help rendered by the Department of Science and Technology for carrying out the project. The authors are also grateful to the Director of the Indian Statistical Institute Calcutta, for providing the necessary infrastructure for the said investigation.

References

AMMERAAL, L., 1986, *Computer Graphics* (Chichester: John Wiley).

BEYRICH, F., 1993, On the use of sodar data to estimate mixing height. *Journal of Applied Physics*, **B57,** 27–35.

CHATTERJEE, N., PAL, P., and DAS, J., 1996, Automatic recognition of SODAR patterns: a rule-based approach (an intermediate draft).

CHAUDHURI, B. B., DE, A. K., GANGULY, A., and DAS, J., 1992, Automatic recognition and interpretation of sodar records. *Indian Journal of Radio and Space Physics*, **21,** 123–128.

DAS, J., DE, A. K., MUKHERJEE, D. P., *et al.*, 1994, SODAPRETER: Sodar Data Interpreter (Part I), Technical Report No. ECSU/DST/1.0/94, Electronics and Communication Sciences Unit, Indian Statistical Institute, India.

DE, A. K., TRIPATHI, S., and DAS, J., 1994, On fine structure of dot echoes as observed by acoustic sodar. *International Journal of Remote Sensing*, **15,** 2157–2165.

FOKEN, T., HARTMAN, K. H., KEDAR, J., KUCHTER, W., NEISSER, J., and VOGT, F., 1987, Possibility of an optimal encoding of sodar information. *Z. Meteorology*, **35**, 348–354.

GONZALEZ, R. C., and WINTZ, P., 1987, *Digital Image Processing* (New York: Addison-Wesley Publishing Company).

HARALICK, R. M., and SHAPIRO, L. G., 1982, *Computer and Robot Vision* (New York: Addison-Wesley Publishing Company).

OTSU, N., 1979, A threshold selection method from grey level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, **9,** 61–65.

TRIPATHI, S., DE, A. K., and DAS, J., 1992, Computer analysis of atmospheric plume structures. *Indian Journal of Radio and Space Physics*, **21,** 321–328.

TRIPATHI, S., DE, A. K., and DAS, J., 1993, Computer algorithm of noise removal in acoustic radar echograms. *Indian Journal of Radio and Space Physics*, **22,** 301–305.