Seismic Time Section Analysis Using Machine Vision

Peter Tu, Andrew Zisserman, Iain Mason
Oxford University, Department of Engineering Science
Parks Road, Oxford OXI 3PJ
eMail peter.tu@eng.ox.ac.uk

Abstract

In this paper we introduce a three stage approach for extracting events from a seismic time section. The approach incorporates recent techniques from computer vision such as deformable templates and multiple hypothesis tracking, and also takes advantage of constraints arising from the physics of seismic reflectors. First, a 2D local matched filtering scheme is used to reduce the time section to a collection of event tokens suitable for tracking with a Kalman filter. Second, a multiple tracking system is used to analyse regions with crossing events. By using a dynamic model of event shape parameters, Kalman filters are able to track events that deviate from an ideal form. Finally, based on events found through Kalman filtering, flexible templates are used to exploit similarity between events. Due to the global nature of the flexible template search process, even events with sections of poor visibility can be identified. Based on this research, a semi-automatic system for extracting seismic events has been developed.

Introduction

Geoexploration is the attempt to identify and locate significant features of the earth's sub-surface. Seismic experiments are used to illuminate the area of interest by sending pulses of energy into the earth's sub-surface. When these pulses hit a discontinuity they are reflected back to a bank of receivers. A seismic trace is a log of the reflected energy recorded by a single receiver as a function of time (see figure 1). By placing the traces of each receiver side by side, a seismic time section is created. Figure 2 is a sample time section taken from an experiment in the North Sea - see Mason [14]. All of the reflected energy due to a particular discontinuity is known as a seismic *event*. Once an event has been accurately identified, this information can help to deduce the nature and location of the discontinuity. The seismic time section is similar to a medical image in that there are no hard edges. Authors such as [1, 4, 7] have demonstrated successful applications of machine vision techniques to medical imaging, we wish to do the same for seismic imaging.

There are several issues that must be considered when developing a pattern recognition system for event extraction from a time section. The time of arrival of a seismic pulse depends on the position and shape of the discontinuity and the average velocity of the pulse along the path traveled. If the average velocity of each

pulse remains constant from trace to trace and if targets are small, events on the time section will have a hyperbolic shape. The hyperbola can be parameterised by the position, (x_p, y_p), of the discontinuity along with the average velocity v of the pulse. Using the Pythagorous theorem, the event shape based on the experimental geometry in figure 1 can be characterised in the following hyperbolic form:

$$(yv - \sqrt{x_p^2 + y_p^2})^2 = (x - x_p)^2 + y_p^2 \qquad (1)$$

Where x is the receiver position and y is the time of arrival of the pulse. However, if the propagation medium is inhomogeneous, local deviations from hyperbolic form will occur. Also there are often multiple reflectors, so that time sections are usually composed of a collection of seismic events that in general cross one another. Due to various noise factors, certain areas of the time section will be very difficult to analyse. A robust event recognition system should take these issues into account.

Huang [8] and others have used Hough and Radon transforms to search for events on time sections. However, these methods for the most part rely on rigid templates which perform poorly in the face of deviations of the event from pure hyperbolic form. Geerlings [6] used the A* search algorithm, a heuristic tree search (see Lee [12]), to track events using a cost function based on pulse similarity. Since the A* algorithm is a form of dynamic programing, it is difficult to take advantage of local event shape using this type of cost function. This makes it susceptible to crossing events. A number of researchers such as Le [11] and Lu [13] have proposed systems that extract seismic events by maximising the correlation between trace pulses. These systems operate under the assumption that there are no crossing events. This is inappropriate for the general case.

In this paper we propose a three stage approach for extracting events. First, matched filters are used to exploit local trace coherency so as to reduce the time section to a collection of tokens that represent the time of arrival of each pulse and the local orientation of the event. Second, Kalman filters use these tokens to track the events by incorporating a dynamic model for the shape parameters. When ambiguities occur due to crossing events, the system will follow multiple tracks until the proper segmentation decision becomes evident. Thirdly, once a track has been found, it is used to construct a flexible template to search for similar events elsewhere in the time section.

Token Extraction

Figure 4 shows a portion of the time section in figure 2. The first stage of this system is to reduce the discretized time section to a collection of tokens which are suitable for tracking with a Kalman filter. If it is decided that an event passes through pixel (i, j), a token showing the local orientation of the event is associated with the pixel. Since there are crossing events, an individual pixel may have multiple tokens associated with it. Figure 5 shows the tokens extracted from figure 4.

In order to take advantage of pulse characteristics and local correlation between traces, a bank of 2D matched filters is used to assign tokens to the pixels. All of the filters are localised both in space and frequency. Each filter is tuned to a pulse with a specific scale and a specific local event orientation. By convolving the image with the matched filters, a local spectrum can be assigned to each pixel. When an event is centered on pixel (i, j), there will be a local spectral maxima. This spectral maxima will also be a local spatial maxima with respect to the equivalent
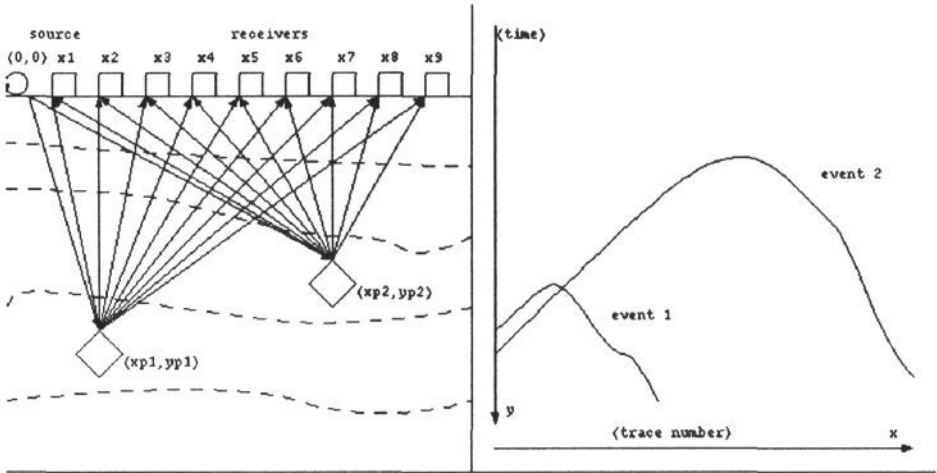
Figure 1: The left hand figure depicts a source at position (0,0) which radiates a pulse through an inhomogeneous medium and encounters two reflectors at (xp1,yp1) and (xp2,yp2). The pulse is then reflected back to a bank of receivers at (xi,0). The right hand figure is the time section created by placing the energy received by the receivers as a function of time, side by side. There are two roughly hyperbolic events that cross one another. Perturbations in the events are due to an inhomogeneous velocity field.
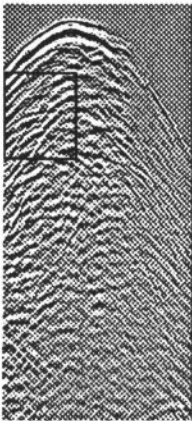

Figure 2: A time section created from an experiment performed in the North Sea. The vertical axis is the time of arrival (time going from the top to the bottom). The horizontal axis is the receiver position. There are 286 by 1024 data points
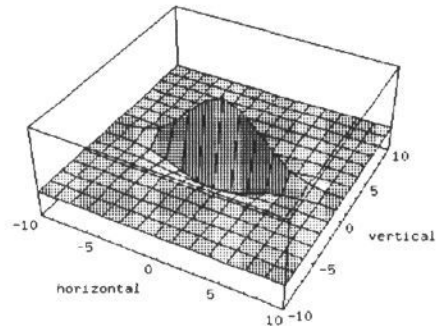

Figure 3: Example of a 2D matched filter kernel. This filter is localised both in space and in frequency. The axes are in pixels and the standard deviation in the vertical direction is 1.
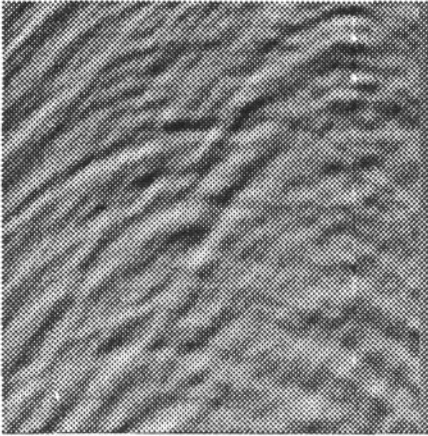
Figure 4: Detail of figure 2. The shape of these events deviate from hyperbolic form. There are a large number of crossing events and areas with poor visibility. The size of this time section is 128 by 128 pixels. The vertical axis is time and the horizontal axis trace number.
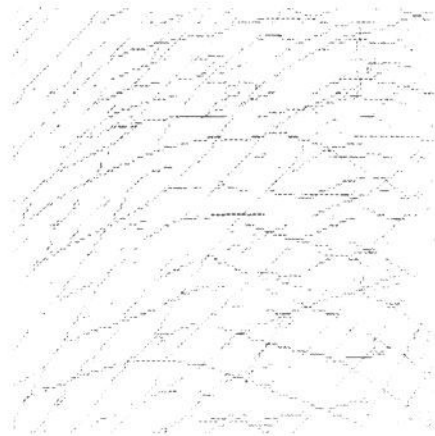
Figure 5: Using a bank of 2D matched filters, the time section shown in figure 4 is reduced to a set of event tokens depicted by line segments. The orientation of the line segments indicate the local orientation of the event. By searching for secondary local spectral maxima, good token extraction is achieved even in regions with crossing events.

spectral components for pixels that are on a line perpendicular to the event. If there are multiple events passing through the pixel there will be multiple spectral maxima. A search algorithm similar to that used by Perona [15] is used to locate these event conditions.

In this application the matched filter kernels are created by the product of a vertical and horizontal filter. The vertical component is the second derivative of a Gaussian with standard deviation σ. The horizontal component is a regular Gaussian with standard deviation 3σ (see figure 3). The kernel is either sheared or rotated by an angle θ to match a particular orientation and σ is varied to match a particular scale. The power of the kernel is normalised. In this application shear angles ranged from $-\pi/3$ to $\pi/3$ radians and the choice of σ was from 1.25 to 2.5 pixels.

Multiple Track Kalman Tracking

Here we outline a strategy for tracking an event through a time section containing multiple crossing events. Using the event tokens as input observations, a tracker attempts to trace an event across the time section. Due to the physics of the experiment, the seismic event is locally hyperbolic and can be modeled using the three shifting shape parameters x_p, y_p and v - see equations 2 and 3. The tracker is based on the extended Kalman filter. Validation gating is used so that much of the data can be ignored. When multiple valid observations are encountered, the tracker performs a branching operation creating a track tree that represents all possible interpretations. Pruning of the track tree is carried out in order to

maintain a manageable tree size.

Extended Kalman Filter

The extended Kalman filter approach, as it is used by Porrill [16] in the field of computer vision to extract ellipses from edge data, provides a natural framework for event tracking. The Kalman filter is a sequential estimator of a set of dynamic model parameters based on a series of related observations. During each iteration of the filter process, an estimate of the model parameters is computed, a confidence measure of this estimate calculated and a prediction of the next observation made. The Kalman filter takes into account both model and observation noise. In this application, the observations are the token variables (x, y, θ) where x is the receiver location, y is the time of arrival and θ the local orientation of the event. The dynamic/shape parameters are (x_p, y_p, v). Based on the geometry of the experiment in figure 1, the observations and the shape parameters can be related through the following model (from equation 1):

$$y = (\sqrt{x_p^2 + y_p^2} + \sqrt{(x - x_p)^2 + y_p^2})/v \tag{2}$$

$$\theta = \partial y/\partial x \tag{3}$$

Where the shifting parameters are modeled as follows:

$$\mathbf{X_i} = (v, x_p, y_p)^T \tag{4}$$

$$\mathbf{X_i} = \mathbf{X_{i-1}} + \xi \tag{5}$$

with ξ an independent Gaussian noise function with zero mean and standard deviation $\mathbf{Q}$. The innovation $\mathbf{z}$ is:

$$\mathbf{z} = (y_o - y, \theta_o - \theta)^T \tag{6}$$

where y_o is the observed time of arrival, θ_o is the observed local orientation, y and θ are calculated from equations 2 and 3 using the current parameter estimate $\mathbf{X_i}$. The Kalman Update equations are as follows:

$$\mathbf{X_i} = \mathbf{X_{i-1}} + \mathbf{K}\mathbf{z} \tag{7}$$

Where $\mathbf{K}$ is the Kalman gain function:

$$\mathbf{K} = (\mathbf{P_{i-1}}\nabla\mathbf{x}^T)(\nabla\mathbf{x}\mathbf{P_{i-1}}\nabla\mathbf{x}^T + \Lambda)^{-1} \tag{8}$$

and $\mathbf{P}$ is the confidence matrix and Λ the covariance matrix of the observation noise.

$$\mathbf{P_i} = \mathbf{P_{i-1}} - \mathbf{K}\nabla\mathbf{x}\mathbf{P_{i-1}} + \mathbf{Q} \tag{9}$$

By incorporating model noise into the Kalman filter, the shape parameters are able to shift. This allows events that deviate from hyperbolic form to be tracked. For a given x position in the data set and a set of shape parameter estimates, the local y position and the local orientation θ can be predicted. The Kalman filter limits the choice of observations for a given x position by analyzing the difference between the predicted observation values and the true observation values. When this difference is normalised, it may be modeled as a χ^2 distribution with 2 degrees of freedom. By thresholding based on a confidence limit, a highly selective filter is created, for both y and θ, that is used to reject tokens as part of the current track. This operation is known as validation gating (see Bar-Shalom [2]). In this application a confidence limit of 0.95 was chosen.
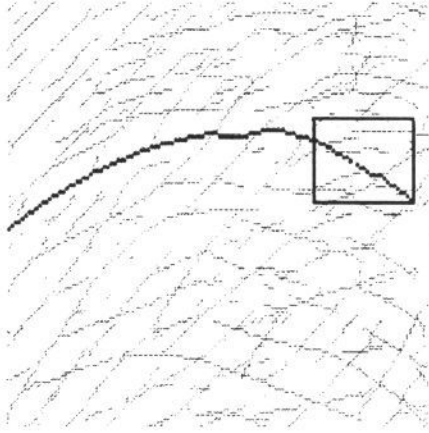
Figure 6: The results of multiple tracking with Kalman filter trackers using the tokens of figure 5 as input. The solid circles indicate tokens that were connected to form the extracted event. Even though the event shape is not a perfect hyperbola and there are multiple crossing events, the event was successfully extracted.
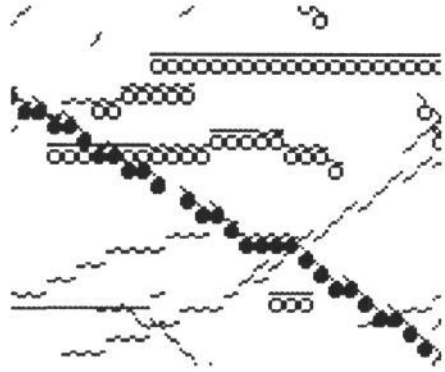
Figure 7: Details of box in figure 6. Tokens with solid circles were connected to form the extracted event. Tokens with hollow circles were considered by the multiple system and simultaneously tracked. Line segments with no circles at all were not considered due to validation gating with a confidence limit of 0.95.

Pruning

Cox [5] uses a multiple hypothesis algorithm in which all possible combinations of tracks are considered, and pruning is carried out based on a probabilistic measure. In a seismic time section, crossing events quickly diverge from one another so that a more simplistic pruning strategy can be implemented. In our strategy, each branch at trace position x is given a probabilistic measure. The branches at trace $x - N$ are re-evaluated based on the performance of their children at trace x. For each observation at trace $x - N$, the best branch is kept and suboptimal branches along with their children are terminated. This form of looking ahead delays the pruning decision until there is sufficient information to disambiguate the situation. In our application we use a value of $N = 4$. In the tree growing process, branches may be extended even though no observations are encountered so that events with observation gaps can be tracked. Using the track tree and a simplified pruning strategy, the problem of tracking through crossing events can be handled efficiently. Figure 6 shows an example of this process. Figure 7 details the tracking operation.

Tracker Initialization

In order to track successfully, the Kalman filter requires a good initial estimate of the shape parameters. Porrill [16] takes a few well spaced points to estimate the initial parameters. However since our aim is to isolate the event, we do not have well spaced points readily available. At most we can rely on a limited region of the event with which to make our parameter estimates. Equation 2 can be re-parameterised in terms of apex position (x_a, y_a) and velocity v resulting in a

new set of state variables, where $x_a = x_p$ and $y_a = (\sqrt{x_p^2 + y_p^2} + \sqrt{y_p^2})/v$. By searching for the apex positions in the data, the trackers can be initialised with a strong estimate of x_a, y_a and a weak estimate of v which is reflected in the initial confidence matrix.

To find the initial apex positions we remove all of the tokens with local orientation greater than 12 degrees. A Hough transform is then used to locate likely apex positions. By using the tokens found around the apex, an initial estimate of the hyperbolic parameters (x_a, y_a, v) can be calculated by an adaptation of the Bookstein [3] conic fitting algorithm (see Appendix A).

In our automatic system, all of the apex positions found using the Hough transform are used to start a Kalman filter tracker. All successful events are then reported. Figure 8 shows the result of this system. There were 74 apex candidates found, 10 of these 74 tracks were successful requiring 15 minutes of processing time on a Sun Sparc 2 work station. The time required to track a particular event, ranged from 8 to 25 seconds. The initial Hough transform required 7 seconds.

An alternative method for event initialization is to use saliency mapping (a form of local relaxation) as developed by Ullman [17] in order to extract salient portions of the events. However, if we use a non apex segment of the event, we can not constrain (x_p, y_p).

Flexible Templates

Once an event has been extracted using Kalman filtering, a flexible template (see Kass [9] and Lipson [10]) can be created from this event for the purpose of exploiting the structural similarity between related events. When the template is translated to a different position in the *time section* its shape is modified to be consistent with the geometry of the experiment. This is achieved by fitting equation 1 to the translated data points using the adapted Bookstein algorithm. This determines the hyperbola that best fits the translated position while maintaining physical plausibility.

The flexible template locks onto an event by searching the time section for local event structure predicted by the template within the search window. Like the token extraction algorithm (see Token Extraction), the image is convolved with a set of 2D locally matched filters. For every x position along the template a search window in the y direction is specified by analysing the behavior of the original event. Each pixel in the window is assigned a value. This value is determined by choosing the maximum output of the convolution kernels centered on the pixel that corresponds to the orientation specified by the template. By repeating this process for each x position, a 2D kernel response is created. If the template is relatively close to a similar event, the event will cause local maxima to occur all the way across the 2D kernel response (see figure 9). Due to the similarity of the template and the event, the local maxima will be placed in a roughly linear fashion along the 2D kernel response. By integrating over all possible lines, a maximum line integral is found and used as a base for picking the new data points. The algorithm then determines the new data points by choosing for each x the closest local maxima to the base line. The template is then reformed using the Bookstein algorithm with the new data points. This procedure is performed iteratively until the template has converged. Due to the nature of this type of search, the template can extract events that have regions of high noise corruption but have sufficient support scattered throughout the time section.
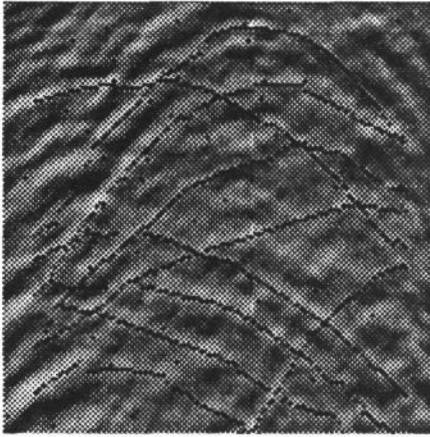
Figure 8: A Hough transform is used to locate apex positions in order to automatically initialize the Kalman trackers. The solid dots show the events that where successfully tracked. There where 74 possible initial apex positions (squares) found resulting in 10 successful events (circles).
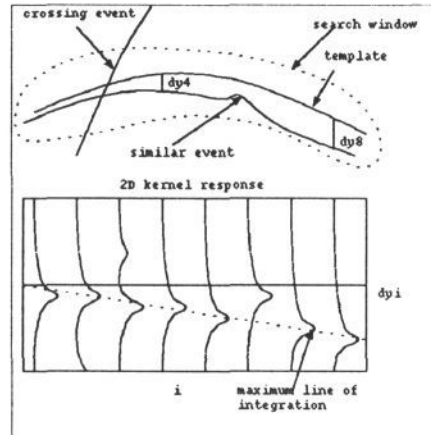
Figure 9: The top half shows a template near a similar event with interference from a crossing event. The bottom half shows the resulting 2D kernel response with a maximum line of integration. The new points are chosen by searching for the closest local maxima to the maximum line of integration.

Initialisation

An interactive application has been created in which the user can choose from a set of flexible templates and then place them on similar events that are to be extracted. The initial templates are found using the Kalman filter tracking mechanism. Figure 10 shows an example of this process.

Discussion

In order to convert the *time section* into a map of the earth's sub-surface, the imaging process must be inverted. Event parameterisation is used to constrain the inversion process as well as to drastically reduce the amount of data that needs to be processed.

In this paper we have developed and integrated a number of techniques from computer vision to create a systematic approach for automated extraction of events from a general seismic time section. The novel contributions of this paper are: 1) unlike Cox [5] we do not have to track all of the events simultaneously; 2) we introduce a useful algorithm for hyperbolic fitting based on the Bookstein [3] conic fitting algorithm; 3) we use both local orientation and position instead of just position for tracking data for the Kalman filter; and 4) we use apex position as a method of automatic filter initialization and event parameterization. In this system we do not need to specify the velocity field.

Another application of Kalman tracking and flexible templates is extracting related events from a series of time sections. When using 3D seismic experiments, a rectangular array of receivers is used instead of a single line of receivers. This process generates a collection of time sections. In this 3D data volume, the energy
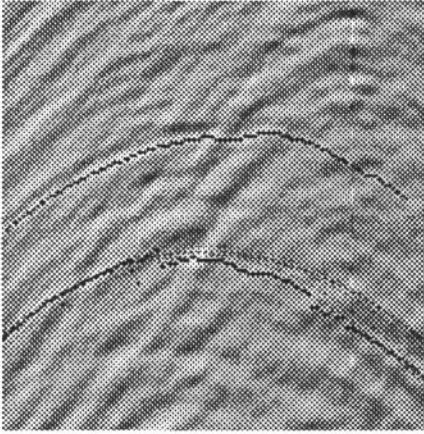
Figure 10: Using the extracted event in figure 6, a flexible template was created. The multiple dotted lines around the new event show successive approximations to the event shape. The upper set of solid dots represent the original template and the bottom set represent the new event
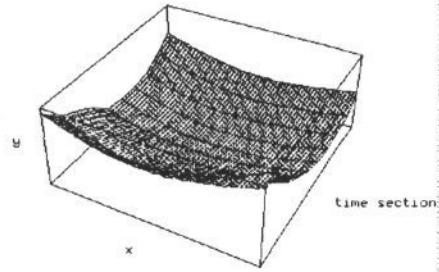
Figure 11: The flexible template system is used to extract a hyperbolic surface found in a 3D data volume created by a rectangular array of receivers. This surface corresponds to the first event in the data.

from a discontinuity creates a hyperboloid resulting in an event in each of the time sections. By using flexible templates to search for these similar events, the hyperboloid can be extracted. Figure 11 shows the hyperboloid generated by searching for the first major event with the flexible template system.

Future topics of research will include: 1) automation of the process for choosing and placing the flexible templates; 2) extending the 3D event tracking process to events that are buried deep in the 3D data volume; 3) removing tracked events from the *time section* without disturbing crossing events; and 4) identification of regions with poor visibility.

Appendix A - Hyperbola Fitting

Equation 2 can be reformed into:

$$Q(x, y) = Ax^2 + By^2 + Cx + Dy = 0 \tag{10}$$

Using the least squares criteria, we wish to minimise $\sum_i Q(x_i, y_i)^2 = V^t SV$ where:

$$V = (A, B, C, D)^t = (V_1, V_2)^t \tag{11}$$

and

$$S = E[(x^2, y^2, x, y)^t(x^2, y^2, x, y)] = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} \tag{12}$$

We constrain the minimization process by setting:

$$A^2 + B^2 = 2 \tag{13}$$

Minimising using Lagrangian multipliers, results in:

$$V_2^t = -V_1^t S_{12} S_{22}^{-1} \tag{14}$$

$$(S_{11} - S_{12} S_{22}^{-1} S_{21})V_1 - \lambda I V_1 = 0 \tag{15}$$

Where λ is a lagrangian multiplier and I is the identity matrix. Equation 15 is a standard Eigenvector problem, with V_1 equal to one of the Eigenvectors.

References

[1] C.I. Attwood, G.D. Sullivan, G.P. Robinson, K.D. Baker, A.C.F Colchester,'Model-based interpretation of anatomical structures in cranial MR images',BMVC90, pp.145-150, 1990.

[2] Y. Bar-Shalom, T.E. Fortman, 'Tracking and Data Association',Mathematics in Science and Engineering, vol 179, Academic Press, 1988.

[3] F.L. Bookstein, 'Fitting conic sections to scattered data', Computer Graphics and Image Processing 9, pp. 56-71, 1979.

[4] G.J. Brelstaff, M.C. Ibison, P.J. Elliot, 'Edge-region integration for segemention of MR images',BMVC90,pp. 139-144, 1990.

[5] I.J. Cox, J.M. Rehg, S. Hingorani,'A Bayesian multiple hypothesis approach to contour grouping and segmentation', to be published in Int. J. of Computer Vision.

[6] A. Geerlings, 'Adaptive tracking in seismic exploration' (thesis), Delft University of Technology, Delft, the Netherlands, 1990.

[7] A. Hill, T.F. Cootes, C.J. Taylor, 'A generic system for image interpretation using flexible templates', BMVC 92, pp. 276-285, 1992.

[8] K. Huang, S.W. Cheng, K.S. Fu,'Seismic image processing (1) Hough transform (2) thinning process (3) linking process', Pattern Recognition and Image Processing in Seismology, Aminzadeh.F (ed), 1987.

[9] M. Kass, A. Witkin, D. Terzopoulos, 'Snakes:Active contour Models', ICCV1, pp. 259-268, 1987.

[10] P. Lipson, A.L.Yuille, et al,'Deformable templates for feature extraction from medical images', proc ECCV, pp. 413-417, 1990.

[11] H.T. Le, E. Nyland, 'Pattern analysis of seismic records', Geophysics, vol. 55, no. 1, 1990.

[12] C.Y. Lee, 'An algorithm for path connections and its applications', IRE transactions on electronic computers, pp. 346-365, 1961.

[13] S.Y. Lu, Y.C. Cheng, 'An iterative approach to seismic skeletonization',Geophysics, vol. 55, no. 10, pp. 1312-1320, 1990.

[14] I. Mason, G. Jackson, D. Lee, 'Surface-to-borehole fault imaging in a buried horizontal waveguide - an experimental study in the southern North Sea',Geophysics, vol. 57, no.4, pp. 532-542, 1992.

[15] P. Perona, 'Steerable-scalable kernels for edge detection and junction analysis'. Proc. ECCV, 1992.

[16] J. Porrill, 'Fitting ellipses and predicting confidence envelopes using a bias corrected Kalman filter', Image and Vision Computing, vol. 8, no. 1, pp. 37-41, 1990.

[17] A. Ullman, 'Structural saliency: the detection of globally salient structures using a locally connected network', ICCV, 1990.