GEOPHYSICS, VOL. 66, NO. 5 (SEPTEMBER-OCTOBER 2001); P. 1488–1496, 11 FIGS.

Automatic event picking in prestack migrated gathers using a probabilistic neural network

Michael E. Glinsky*, Grace A. Clark‡, Peter K. Z. Cheng**, K. R. Sandhya Devi§, James H. Robinson§, and Gary E. Ford**

ABSTRACT

We describe algorithms for automating the process of picking seismic events in prestack migrated common depth image gathers. The approach uses supervised learning and statistical classification algorithms along with advanced signal/image processing algorithms. No model assumption is made, such as hyperbolic moveout. We train a probabilistic neural network for voxel classification using event times, subsurface points, and offsets (ground truth information) picked manually by expert interpreters. The key to success is using effective features that capture the important behavior of the measured signals. We test a variety of features calculated in a local neighborhood about the voxel under analysis. Selection algorithms ensure that we use only the features that maximize class separability. This event-picking algorithm has the potential to reduce significantly the cycle time and cost of 3-D prestack depth migration while making the velocity model inversion more robust.

INTRODUCTION

There is an increasing need for 3-D prestack depth migration (PSDM). It possibly gives better resolution and placement of events than conventional time migration, especially in areas of complex structure such as near salt bodies. Unfortunately, the iterative process of finding the correct velocity model for the PSDM is a bottleneck in determining cycle time, cost, and quality. Traveltime tomography relies on the automatic or manual picking of events which are inverted to give a correct velocity model. In this tomographic velocity model updating process, a primary bottleneck is the manual picking of prestack events. The velocity model inversion method that we use in conjunction with PSDM needs prestack picks of migrated, depth-imaged gathers. The automatic picking method described here is applied to these depth-imaged gathers that have been migrated with a trial velocity model. If the velocity model is incorrect, over- or undermigration of the imaged events will be observed in the migrated, depth-imaged gathers, sorted in the offset-depth domain. These events are picked using the probabilistic neural network method described in this paper. In practice, the prestack migrated depth imaged gathers are converted back to time-offset gathers for signal processing purposes. This is done using the local trial velocity model to minimize the effects of depth stretching usually seen on depth-imaged gathers. The term common image point (CIP), used frequently in this paper, refers to these prestack migrated depth-imaged gathers that have been converted to time-domain gathers. The picks are used to iteratively update the velocity model, which forms the next set of prestack migrated gathers. The number of picked events and iterations are determined by economic and business factors. Increasing either the number of picked events or iterations could lead to a more robust and accurate inversion. By automating a significant portion of this picking process, we hope to enable and improve PSDM.

The conventional automatic picking of events on prestack migrated gathers is complicated because of the low S/N ratio. This leads to loop skipping, since most conventional picking algorithms follow the noise from one local maximum to another and skip to another phase of the wavelet (2π from the original). This is very undesirable for the velocity updating algorithms and must be corrected manually.

The automatic event-picking technique we describe uses advanced algorithms from the areas of automatic target recognition, computer vision, and signal/image processing. Whenever possible, prior knowledge of the geophysics is incorporated into the processing algorithms to ensure physical relevance

Presented at the 66th Annual Meeting, Society of Exploration Geophysicists. Manuscript received by the Editor December 18, 1998; revised manuscript received January 25, 2001.
*BHP Petroleum, 1360 Post Oak Blvd., Suite 150, Houston, Texas 77056. E-mail: glinsky@mailaps.org.
‡Lawrence Livermore National Laboratory, P.O. Box 808, Livermore, California 94550. E-mail: clark9@llnl.gov.
**University of California, Davis, One Shields Ave., Davis, California 95616.
§Shell Exploration and Production Technology Co., Bellaire Research Center, P.O. Box 481, Houston, Texas 77001-0481.

and to enhance the ability to obtain meaningful results. Supervised learning methodology is used to train a probabilistic neural network (PNN) (Specht, 1990b) for voxel classification using manually picked event times. The key to success is in using effective features that capture the important behavior of the measured signals. The algorithm uses a variety of 2-D features calculated in a neighborhood about the voxel under analysis. These features are designed to capture the character of the event for which expert pickers look.

An interesting aspect of the proposed algorithm is the use of proximity features to limit the event search space to only those events specified as important by the analyst. The proposed algorithm generally picks all possible events in a panel of common prestack migrated-depth image point gathers (CIPs). It is possible to search only for those primary reflections of interest by exploiting knowledge provided by the expert analyst. The algorithm uses special proximity features to measure the distance of the voxel to the nearest event picked by the analyst. This creates a proximity mask that constrains the search to only those important events picked by the analyst. This process significantly reduces the confusion involved with interpreting the final picks and ensures that the final picked CIP panels are useful. Note that the use of the proximity features is optional.

This work is applied research, with novelty over past work (Lu, 1982; Taner, 1988; Geerlings and Berkhout, 1989; Lu and Cheng, 1990; Veezhinathan and Wagner, 1990; Aminzadeh and Simaan, 1991; Kemp et al., 1992; McCormack et al., 1993; Chu and Mendel, 1994; Woodham et al., 1995) primarily in the creative combination of algorithms from a variety of disciplines with some new algorithms to solve a difficult applied problem. A review of past work is beyond the scope of this paper but can be found in the work of Cheng (1999). The significant contributions of the work are the following:

1) the use of prestack migrated gathers rather than stacked data with a better S/N ratio,
2) the use of a 2-D image-processing approach, including 2-D statistical and wavelet features,
3) the use of the PNN for voxel classification,
4) the use of proximity features to limit the event search spce to only those designated as important by the analyst, and
5) the excellent performance in picking events and avoiding loop skips.

We present this method in four parts: feature definition, feature selection, voxel classification, and application of valley finding with constraints. The results are compared to those from a commonly used correlation picker. Four major conclusions are drawn:

1) 2-D Gabor wavelet features are very effective in capturing the character of events,
2) a PNN combines many different features into a best estimate that eliminates loop skips,
3) proximity features when combined with a PNN quantify where to look for events, and
4) this algorithm has the potential of significant time and financial savings when doing PSDM.

METHOD

We illustrate the method by applying it to a 2-D marine data set with 468 subpoints (spaced every 69 m) and 45 offsets from 260 to 5636 m. There are 1600 time samples with a sampling interval of 4 ms. The flowchart of the numerical algorithm is shown in Figure 1.

Feature definition

The first and most important step we make is to define a set of features to consider. This set needs to capture all of the character of an event used by the expert. Care should be taken to be inclusive; redundant or unimportant features will be eliminated during the feature selection. Since the discrimination of coherent noise, such as multiples, is best in the CIP domain and since we would like to avoid the need to resort the data, we only consider 2-D image features of this gather.

The features are normalized by subtracting from each feature the mean of the feature values calculated over the ensemble of training voxels (defined in the next section) and dividing this result by the ensembel standard deviation. This normalization makes the classifier insensitive to absolute units that can vary from feature to feature.

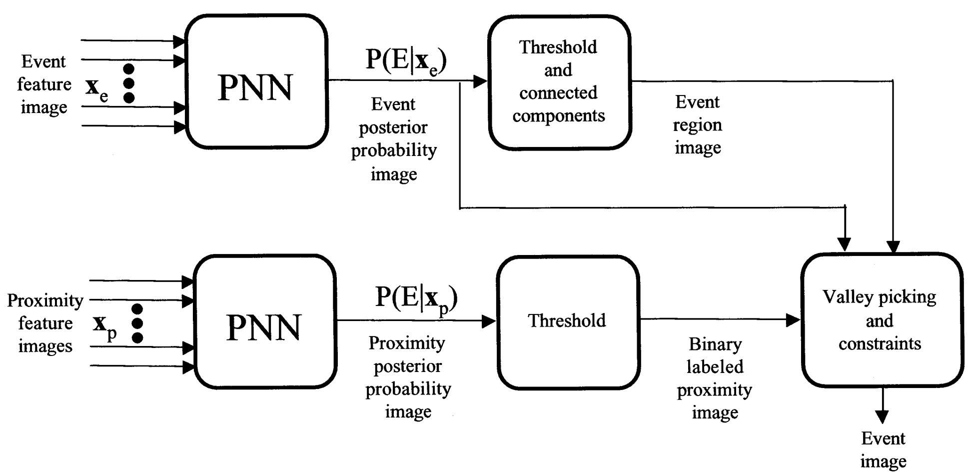

FIG. 1. Flowchart of PNN-based event-picking algorithm.

Gabor.—These features (Gabor, 1946; Morlet et al., 1982a,b; Daugman and Kammen, 1987; Daugman, 1988; Clark et al., 1989, 1991) are derived from hierarchical multiresolution 2-D Gabor wavelet transforms of the CIP panels. These provide magnitude and phase information about the events at a variety of resolutions (scales), orientations (rotational angles), and frequencies. A variety of elliptical Gabor kernels were designed to have several different scales (with corresponding frequencies) and a variety of orientations characteristic of the CIP panel image and seismic wavelet (see Figure 2). Two scales were chosen to match the mean frequency at early time (25 Hz) and at later times (15 Hz). The time width of the Gaussian envelope encompasses three loops of the wavelet (16 and 27 ms). The offset width matches the lateral resolution of the migration (610 and 1039 m). Four different orientations were used, spanning the slope seen in both the events and the coherent noise (0, 16, 39, and 131 ms/km). While this set of eight Gabor kernels is not orthogonal, it does span the information content of the data.

Figure 3 displays the magnitude of two of these Gabor transforms of the image overlaid on the raw seismic traces. Note that the event is highlighted by the one Gabor kernel and the coherent noise by the other. Figure 4 displays the Gabor phase for the same data. The events are picked by the expert at a well-defined phase, that is, at a negative peak. The Gabor magnitude specifies where to look (with a resolution of order of the width of the seismic wavelet envelope), and the Gabor phase specifies exactly where to pick the event with a resolution of the time digitization level.

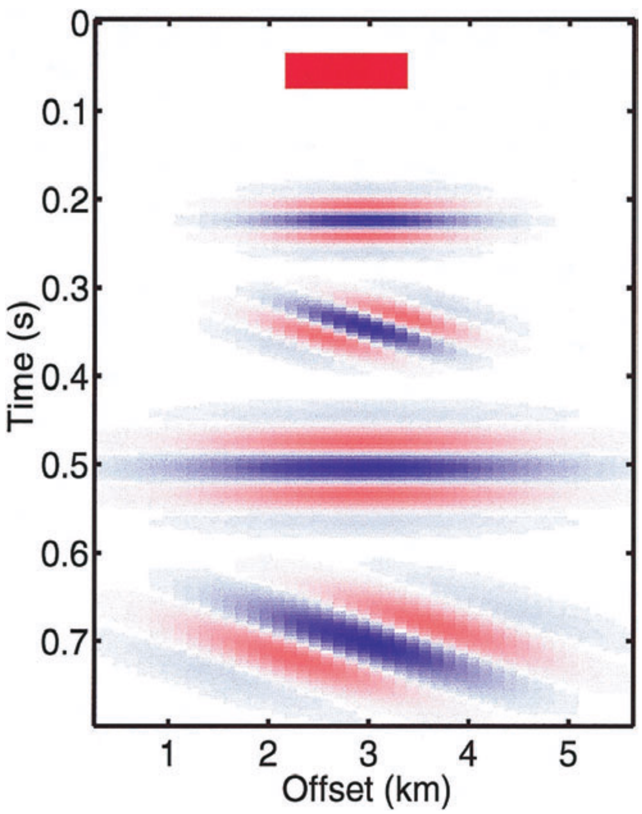

FIG. 2. Gabor kernels and tile used to form event features. Shown (top, in red) is the smaller of the two tiles used to calculate the histogram features (40 ms × 1222 m). Beneath are real parts of the Gabor kernels. (top) Small scale and (bottom) large scale, both with 0 ms/km and 39 ms/km slope. Blue is positive; red is negative.

To be more specific, the formula for the Gabor kernel is

$$K(t, x; t', x') = \exp\left(-T^2/2\sigma_T^2 - S^2/2\sigma_S^2\right)[\cos 2\pi f T + i \sin 2\pi f T], \tag{1}$$

where

$$T \equiv (t - t')\cos\theta + \frac{s - s'}{\Delta s/\Delta t}\sin\theta, \tag{2}$$

$$S \equiv (s - s')\cos\theta - \frac{t - t'}{\Delta t/\Delta s}\sin\theta, \tag{3}$$

$$\tan\theta \equiv \frac{dt/ds}{\Delta t/\Delta s}, \tag{4}$$

f is the frequency of the Gabor kernel, σ_T is the time width, σ_S is the offset width, ds/dt is the orientation, Δt is the time sampling interval, Δs is the offset spacing, s is the offset, t is time, and x is the subpoint.

Amplitude histogram.—These features (Jain, 1989) can also be called statistical moments of the data in an $M \times N$ neighborhood (tile) centered about the image voxel. We started with mean, standard deviation, skewness, and kurtosis using two different-sized tiles (40 ms × 1222 m and 68 ms × 2077 m). The raw amplitude data was also considered. After feature selection (described in the next section), we chose to use only the raw data.

Semblance.—These features (Robinson and Treitel, 1980) are calculated over the local neighborhood and provide a useful indication of the coherence of the seismic traces in the offset direction. This is also the mean square stack amplitude divided by the mean square amplitude. We calculated this over the same two neighborhoods used to calculate the amplitude histogram features. Selection indicated the Gabor features captured the same information as the semblance but with more class separability. The semblance features were therefore not used for classification.

Proximity.—The proximity features are defined as follows. Let Δt represent the temporal sampling interval, which for our data is 4 ms. Specify the location of the voxel currently under analysis by (t, s, x). Specify the location of the jth analyst pick for the ith event by (t_{ij}, s_{ij}, x_{ij}). The first proximity feature, T, is defined as follows:

$$T = \ln\left(\frac{|\Delta T|}{\Delta t} + 1\right), \tag{5}$$

where ΔT is defined as the time difference between the voxel currently under analysis to the nearest event:

$$\Delta T = t - t_{ij}. \tag{6}$$

The natural log in equation (5) was used because we found that when we used equation (6) for the proximity feature, the histogram of the values of T for events had a small dynamic range but the histogram of T for backgrounds had a very large dynamic range. This is undesirable because it leads to poor classifier performance. However, the problem is avoided by scaling the feature using equation (5) because the histograms of the event and background features using that equation have comparable dynamic ranges.

The second proximity feature, d, is defined as the spatial distance between the voxel currently under analysis and the nearest analyst pick:

$$d = \sqrt{(s - s_{ij})^2 + (x - x_{ij})^2}. \tag{7}$$

The third and fourth proximity features are similar to the first two proximity features. The difference is that the next closest analyst pick on the other side of the nearest analyst pick is used. What is meant by the other side is the other side of a line through the voxel under analysis (in the x–s plane) and perpendicular to a line from the voxel under analysis to the nearest analyst pick. These two features are added to allow the PNN to interpolate between two analyst picks that bracket the voxel under analysis.

Feature selection

Feature selection is important for several reasons. First, we want to minimize the effects of the curse of dimensionality, in the sense that classification computational complexity increases rapidly with the dimension of the feature vector. Second, we want to use only features that add significant value to the quality of the classification results. Unimportant or redundant features add negative or zero value and should be removed. It is significant that human feature analysis experts generally produce classification results based upon a very small number of the most important attributes of a signal. If too many features are used, the classifier performance can actually degrade. Statistical decision theory tells us that the probability of correct classification is an increasing function of the number of features provided if the sample size is very large. Empirical studies show that the probability of correct classification is not generally a monotonically increasing function of the number of features used. It generally increases up to a point at which it reaches a knee in the curve and begins decreasing, finally leveling off at a value less than the value at the knee (Devijver and Kittler, 1982; Fukunaga, 1990). Clearly, our goal is to find the number of features corresponding to the knee in the curve. Third, an important byproduct of feature selection can sometimes be increased knowledge of the physical processes that create the data. By understanding which features are statistically most important, we can often draw important conclusions about the physical reasons why they are important, and this can lead to productive insights that aid in the system design.

To select the features and train the PNN classifier, we must create a set of training voxels. First, an expert picks several event times for every offset and subpoint combination in the data set. Twenty equally spaced CIP panels are chosen out of the 468 CIP panels in the full data set. Several of the expert event picks are chosen at random from each of the 20 panels.

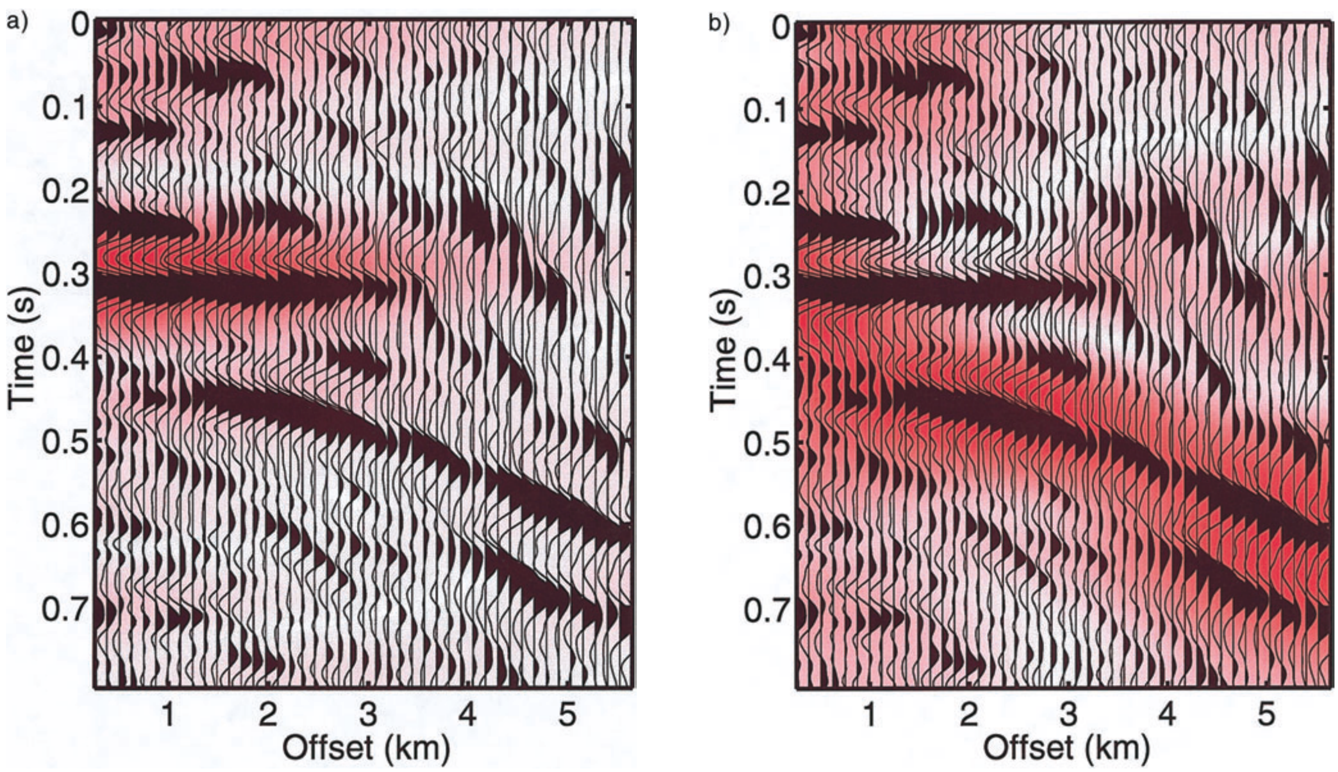

FIG. 3. Gabor magnitude feature images. Seismic amplitude data are shown as the vertical traces. The magnitude of the Gabor transform is shown as an image behind the seismic data. White is zero, and the maximum value is red. There is an arbitrary time origin. (a) Large-scale Gabor kernel with 0 ms/km slope. (b) Large-scale Gabor kernel with 39 ms/km slope.

Care is taken to ensure that the chosen training events are independent by demanding a minimum separation in time and offset among the voxels. Second, several background voxels are manually picked by an expert in each of the 20 panels. Care is taken that these picks represent a variety of background types and are independent.

We first use a formal feature selection algorithm (sequential forward selection) to rank the features according to the Bhattacharyya measure of class separability (Young and Fu, 1986; Fukunaga, 1990). We then choose an appropriate subset of features for actual use by the classifier. This saves computation time and allows us to use only the most effective features. We use the well-known rule-of-thumb (Devijver and Kittler, 1982) for the lower bound on the number of training samples to use: The number of independent training samples needed per class is at least five times the number of features used in the feature vector. This rule also implies an upper bound on the number of features that can be used, given the number of independent training samples. For our problem, we trained the PNN classifier with 107 event voxels and 100 backgroud voxels. This limits us to using about 20 features. In the classifier results presented, we actually used seven features (the raw amplitude data and the magnitude and phase of the Gabor transform using the large-scale kernel with 0 slope and the small-scale kernel with 0 and 16 ms/km slope). By reducing the number of features from 25 to 7, we saved 72% of the computation (CPU) time. Although we sacrificed 52% of the class separability as measured by the Bhattacharrya distance, the performance of the overall algorithm was not significantly degraded.

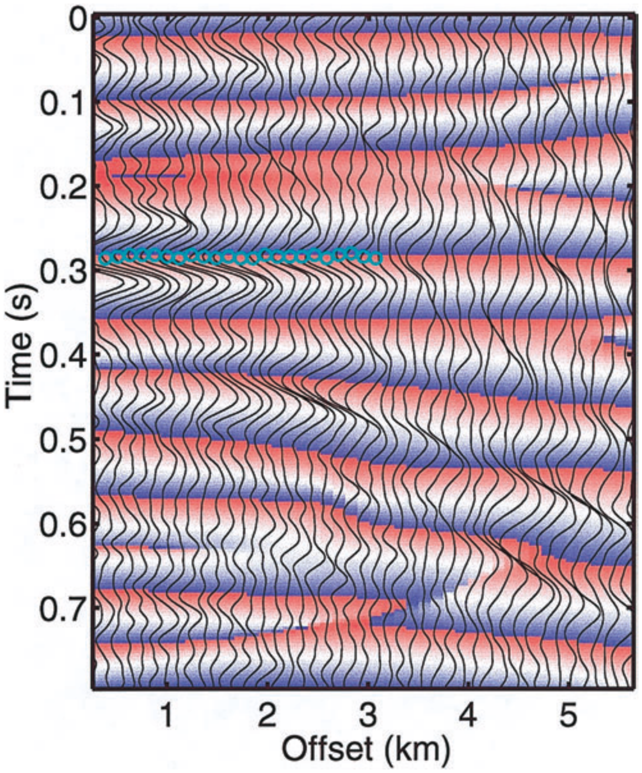

FIG. 4. Gabor phase feature image. Seismic amplitude data are shown as the vertical traces. The phase of the Gabor transform is shown as an image behind the seismic data. Red is $-\pi$ phase, white is 0 phase, and blue is π phase. Large-scale Gabor kernel with 0 ms/km slope is used. Human-edited correlation picks are shown as blue circles.

Voxel classification

PNN.—Linear classifiers such as the Fisher linear discriminant (Devijver and Kittler, 1982; Young and Fu, 1986) create a linear decision surface in feature space. In general, optimal performance requires that the classifier have the ability to create a decision surface of arbitrary shape (i.e., nonlinear). We use a special neural network known as the PNN that has this property.

The PNN is a Bayesian statistical classifier based upon the Parzen estimator of conditional probability density functions (pdf's) (Parzen, 1962; Specht, 1990a,b). The PNN has the desirable property that it provides the Bayes optimal pdf estimate in the limit as the number of training samples approaches infinity. For the two-class problem ($E =$ event and $B =$ background or nonevent), given input data feature vector $\mathbf{x}$, it estimates the conditional probability density function values $f(\mathbf{x}|E)$ and $f(\mathbf{x}|B)$. These pdf values can be used to calculate the posterior probability of E given $\mathbf{x}$, $P(E|\mathbf{x})$, and the posterior probability of B given $\mathbf{x}$, $P(B|\mathbf{x})$. Examples of a posterior probability image are shown in Figures 5 and 6. Figure 5 shows the posterior probability using the seven event features, $P(E|\mathbf{x}_e)$. Notice the loop skipping of a traditional correlation picker in the low S/N ratio area at 3.6 s. Figure 6 shows the quantification of proximity given by the posterior probability using the four proximity features, $P(E|\mathbf{x}_p)$. Only 0.1% of the total number of expert event picks chosen at random were used as analyst picks. This corresponds to one pick every fourth CIP panel. Notice the large area highlighted by this posterior probability. One will need to rely on the event posterior probability to more precisely locate the event, but the proximity posterior probability does indicate where to look.

Classification of the vector $\mathbf{x}$ is obtained by applying appropriate thresholds to the posterior probabilities given above. As depicted in Figure 1, the next step is to form a binary labeled image for each of the posterior probability images $P(E|\mathbf{x}_e)$ and $P(E|\mathbf{x}_p)$ by applying thresholds to them. By thresholding the posterior probability, we classify each voxel in the image to belong to either the class event or the class background. We call the result a binary labeled image. The Bayesian threshold on the posterior probability is a function of the prior probabilities and losses assumed for the analysis. For our application, we cannot reasonably define the losses for the problem, so we assume they are equal. The threshold is therefore not affected by the losses. We can show that the decision threshold for the posterior probability is just the prior probability of the background, that is, $P(E|\mathbf{x}) > P(B)$ to be classified as an event.

For our large data set of 468 CIP panels, we can estimate the prior probability of background $P(B)$ to be the number of background voxels divided by the total number of voxels. Using this method and visually inspecting the images, we estimate $P(B) = 0.7$. Interestingly, after classification with this threshold, the fraction of voxels classified as background is 0.85.

As part of the PNN training, a smoothing parameter σ is chosen. This parameter determines the neighborhood of influence of a training sample to the estimate of the pdf. The value of σ should therefore be larger than the average spacing between training samples but less than the scale on which the pdf varies. This behavior is shown in Figure 7. Displayed is the probability of correct classification using the hold-one-out method (Hogg and Craig, 1978; Devijver and Kittler, 1982; Young and Fu, 1986) as a function of σ. There is a broad plateau between 0.05 (the spacing between training samples) and 1 (the scale on which the pdf varies). If this plateau did not exist, it would indicate there were not enough training samples to sample the pdf. To calculate the posterior probability in Figure 5, we use a value of $\sigma = 1$. This allows for the maximum smoothness in the estimate of the pdf without significantly sacrificing performance.

Connected components.—We create the labeled regions from the binary labeled image using the method of connected components (Jain, 1989; Haralick and Shapiro, 1992). The connected components algorithm is a region-based segmentation technique designed for use with binary images. The algorithm maps the binary labeled image to an image showing regions that are similar according to connectedness measures. All

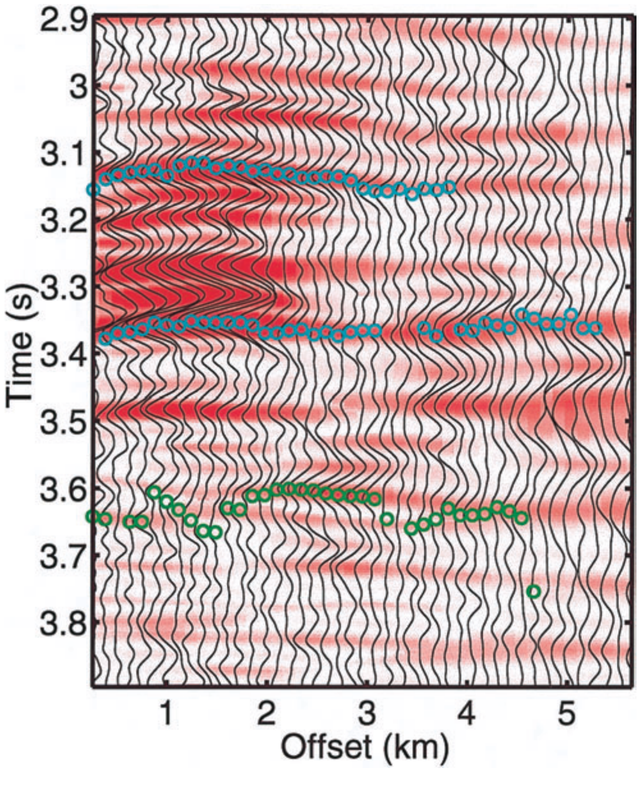

FIG. 5. Event posterior probability image, $P(E|\mathbf{x}_e)$, of a CIP panel, shown as an image behind the seismic data. Seismic amplitude data are shown as the vertical traces. White is 0 probability, and red is a probability of 1. Large-scale Gabor kernel with 0 ms/km slope is used. Human-edited correlation picks are shown as blue circles. Unedited correlation picks in a low S/N area, shown as green circles, demonstrate loop skipping. Time origin is arbitrary.

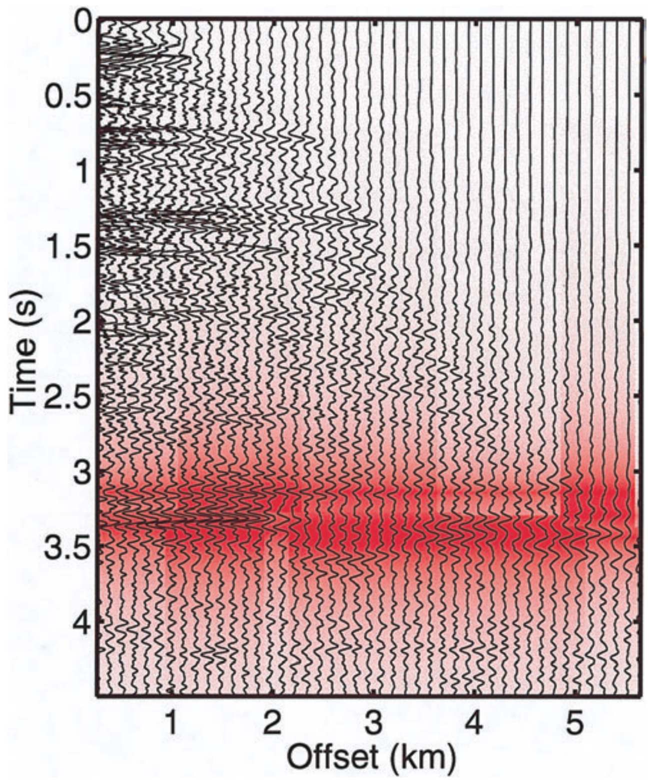

FIG. 6. Proximity posterior probability image, $P(E|\mathbf{x}_p)$, of a CIP panel, shown as an image behind the seismic data. Same panel and time origin as Figure 5. Seismic amplitude data are shown as the vertical traces. White is 0 probability, and red is a probability of 1.

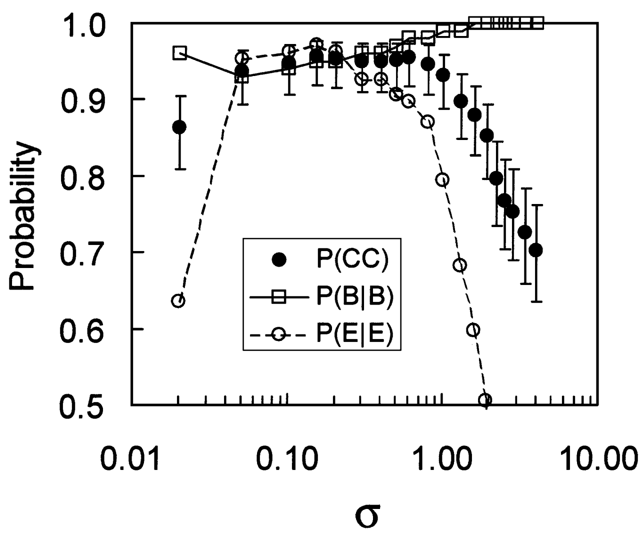

FIG. 7. Tuning curve for event PNN. Shown is the probability of correct classification $P(CC)$ as solid circles. The error bars indicate 95% confidence limits. The probability of detection, $P(E|E)$, or that an event will be classified as an event, is shown as the dashed line with open circles. The probability that a background voxel will be classified as background, $P(B|B)$, is shown as the solid line with open squares. The abscissa is the dimensionless smoothing parameter.

voxels that have the value binary one and are connected to each other by a path of voxels all with the value binary one are given the same identifying label. The label identifies a potential object (event) region. Our definition of connectedness is that two voxels are connected if they share a face, an edge, or a vertex (i.e., a voxel has 26 nearest neighbors). Each region has a rich set of properties which potentially can be exploited, including shape, position, and statistical properties of values of the image voxels corresponding to the region. An example of the results of connected components analysis applied to only one subpoint is shown in Figure 8. Note that the regions or connected components for the seismic images have a variety of shapes. We dub them clouds and refer to the multiple long, approximately horizontal regions within a given cloud as tentacles. The tentacles are likely to represent different loops of the seismic wavelet. We show in the next section that the final event picks can be found using a valley-finding algorithm operating on the clouds.

Valley finding and constraints

We wrote a rule-based valley-finding algorithm to determine the final event picks from $P(E|\mathbf{x}_e)$, $P(E|\mathbf{x}_p)$, and the event region image. We want to find event picks similar to those specified by a human expert. Analysts inspect seismic images on a workstation screen by eye and use a computer mouse to draw lines on the image, showing their judgment of where seismic horizons are located. The picks are continuous (unless there are faults), correspond to peaks in the posterior probability of event, and form single-valued surfaces (in time) in the (x, s, t) space. We are also only interested in events that are nominated by the analyst and can be tracked over a significant range of x and s. The following steps constrain the event picks to ones that satisfy these conditions.

First, find event clouds that have greater than a minimum number of voxels. [we used (10 voxels in t) by (10 voxels in s) by (26 voxels in x) = 2600 voxels.]

Second, find the voxel with the maximum $P(E|\mathbf{x}_e)$ in each event cloud. Use those as first picks.

Third, follow the event in both offset directions, at a constant subpoint, until the limit of the data is reached or the edge of the event cloud is reached. Do this by finding all local maximums of $P(E|\mathbf{x}_e)$ in time within the same tentacle of event cloud at the next offset. The new pick is the one with the minimum change in time from the previous pick.

Fourth, follow the event to the adjacent subpoints unless the limit of the data is reached or the edge of the event cloud is reached. Do this by finding all local maximums of $P(E|\mathbf{x}_e)$ in time within the same tentacle of event cloud at the next subpoint. The new pick is the one with the minimum change in time from the previous pick. Go to step 3. Note: If an event does not contain voxels that are within the binary labeled proximity image, reject it.

The result of applying this rule-based algorithm to only one subpoint is shown in Figure 9 without the proximity constraints). When the proximity constraints are applied, only

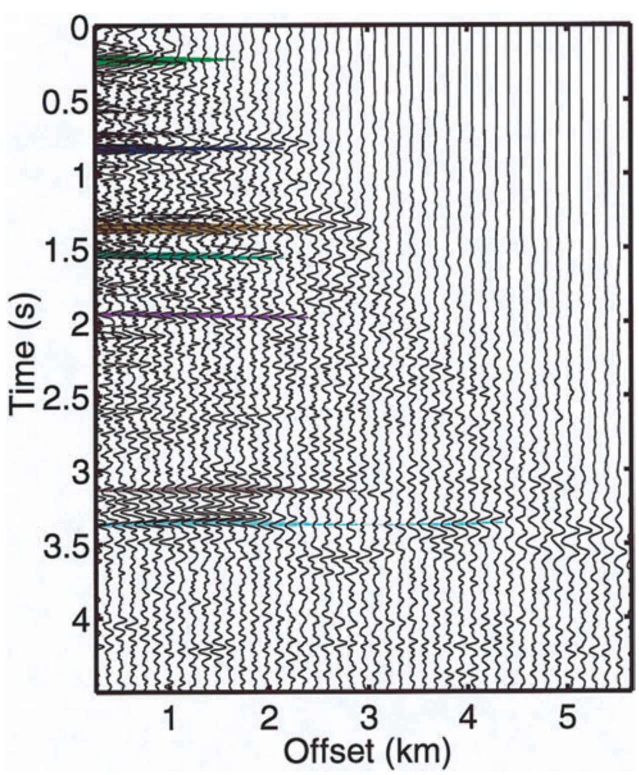

FIG. 8. Event region image of a CIP panel. Same panel and time origin as Figure 5. Seismic amplitude data are shown as the vertical traces. Background is white; each event region is a different color.

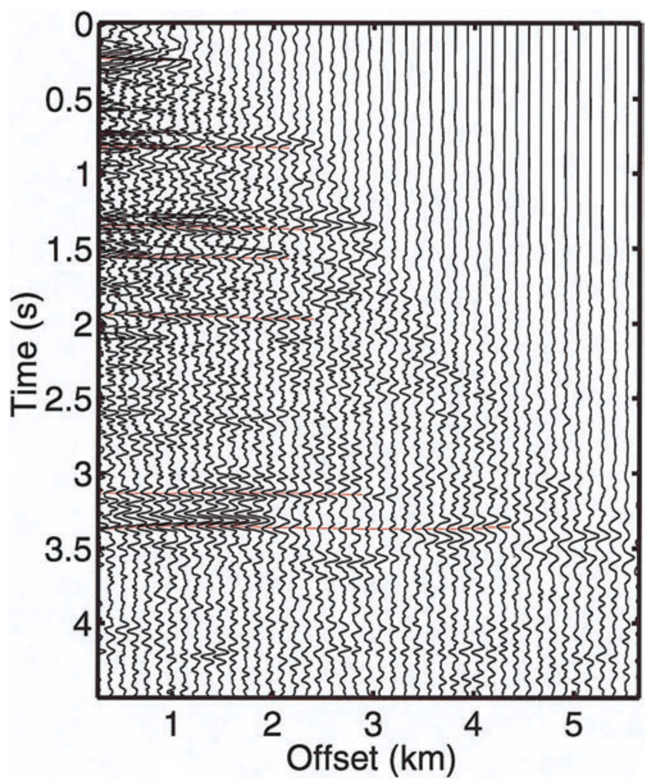

FIG. 9. Event image of a CIP panel. Same panel and time origin as Figure 5. Seismic amplitude data are shown as the vertical traces. Event picks are shown as thin red lines. Proximity constraint is not applied. When proximity constraint is applied, only the two events at 3.1 and 3.3 s remain.

the two events (at 3.1 and 3.3 s) nominated by the analyst remain.

RESULTS

The quality of the picks on the whole 2-D data set was quite encouraging. A total of 35 events were picked without the proximity constraints and 5 were picked with the proximity constraints (9 events were nominated by the analyst). No loop skips occurred on any of the events. This even included the event in the low S/N area that caused the loop skip in the correlation picker (see Figure 10). The event picks matched the expert picks to within the time sampling interval (see Figures 10 and 11). Our algorithm was not as aggressive as the expert, picking approximately 50–70% of the *x–s* area picked by the expert. The aggressiveness of the algorithm could be increased by lowering the prior probability of background $P(B)$ to a value below 0.7.

Further tests were done on selected CIP panels by both increasing and decreasing the number of features from the seven features used to process the whole data set. Adding additional features did not significantly increase the probability of correct classification, reduce the number of loop skips, increase the number of picked events, nor increase the precision of the time picks. Decreasing the number of features to only one Gabor magnitude and phase had a barely noticeable effect on the same performance measures. Not using the Gabor phase, even if the three most important Gabor magnitudes were used, caused a significant degradation in every performance measure.

This algorithm was implemented in interpreted MATLAB code using a Macintosh 5300c Powerbook; it took 15 ms/voxel/feature. We anticipate that compiling an equivalent algorithm on an Ultra Sparc workstation should reduce the time to 150 μs/voxel/feature. The time to process an OCS block (70-m subpoint spacing, 1600 time samples, and 45 offsets) would be 18 days on an Ultra Sparc.

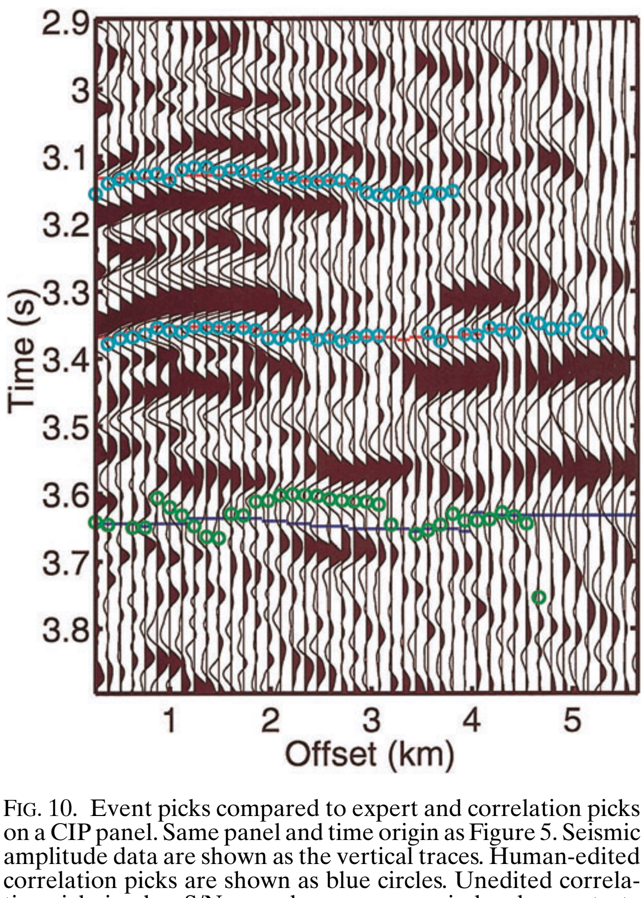

FIG. 10. Event picks compared to expert and correlation picks on a CIP panel. Same panel and time origin as Figure 5. Seismic amplitude data are shown as the vertical traces. Human-edited correlation picks are shown as blue circles. Unedited correlation picks in a low S/N area, shown as green circles, demonstrate loop skipping. The PNN event picks are shown as the red and blue lines. The threshold $P(B)$ had to be lowered to 0.3 from 0.7 to make the blue line picks.

CONCLUSIONS

Several conclusions can be drawn from this work. First, a small number of 2-D Gabor features capture the character of an event. Second, the PNN combines many different features of the data into one best estimate that has very good properties for locating and tracking an event. The probability of correct classification during training is between 89% and 96% (95% confidence limits). When the posterior probability is used in a rather crude, rule-based tracking algorithm, no loop skipping occurred. Third, the proximity posterior probability image is a good way to quantify where to look for events. In practice, this could be used to nominate events by picking the stack. The fourth and most important conclusion is that implementation of this algorithm could reduce the cost and cycle time of 3-D prestack migration while improving the robustness. Estimates indicate the cost of picking four OCS blocks would be reduced from $75,000 (manual picking) to $6,000 (with our algorithm) and the cycle time from 12 weeks to 1 week (assuming use of a multiprocessor computer such as an SP2). The robustness of the inversion would be increased since more events could be picked without additional processing. Note that we had to do work to reduce the number of picked events from 35 to 5.

Although these results are quite encouraging, some issues remain to be explored. The robustness is better than standard

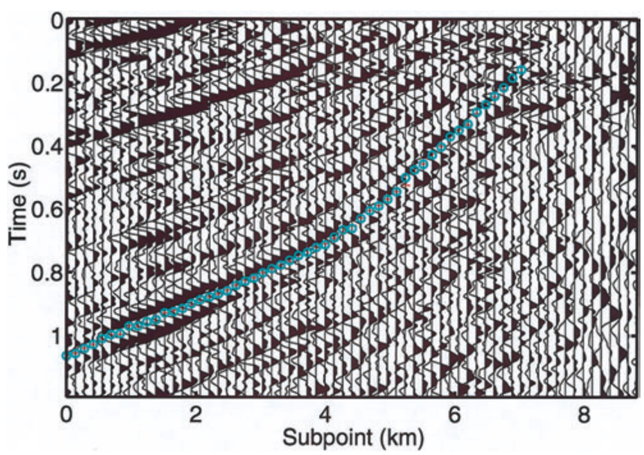

FIG. 11. Event picks compared to expert picks on a common-offset panel. Seismic amplitude data are shown as the vertical traces. Time origin is arbitrary. The PNN event picks are shown as red lines. Human-edited correlation picks are shown as blue circles.

correlation-based pickers, but improvements could still be made. The method also includes the element of the black box PNN. It would be difficult for a user to modify the algorithm if a problem occurred with a particular data set. A possible solution would be to include additional training samples from the data set where the problem occurred. A final issue is the computer execution time. Although our estimates indicate this algorithm would only take 25% of the computer time needed for a velocity update in the PSDM process, the operational implementation needs to be done to prove this. These issues lay the groundwork for future research.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the technical contributions of Cheung Tam and Ronya Yang of the University of California, Davis, and Sailes Sengupta of Lawrence Livermore National Laboratory. We especially thank our sponsor, U.S. Department of Energy through the Advanced Computing Technology Initiative (ACTI) and Shell E&P Technology Co. We thank Bill Butler, Trilochan Padhi, and Jim Roberts of Shell E&P Technology Co. for their important comments and suggestions.

M.E.G. acknowledges the support of a Department of Energy Distinguished Postdoctoral Fellowship and BHP Petroleum. Some of the work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract W-7405-ENG-48.

REFERENCES

Aminzadeh, F., and Simaan, M., 1991, Expert systems in exploration: Soc. Expl. Geophys.

Chu, C.-K. P., and Mendel, J. M., 1994, First break refraction event picking using fuzzy logic systems: Inst. Electr. Electron. Eng. Trans on Fuzzy Systems, **17**, 531–552.

Clark, M., Bovik, A. C., and Geisler, W. S., 1989, Experiments with a theory of visual texture segmentation using modulation/demodulation processes: Pattern Recognition, **6**, 143–150.

Clark, G. A., Hernandez, J. E., Lawver, B. S., and Sherwood, R. J., 1991, Gabor transforms and neural networks for automatic target recognition: Workshop on Neural Networks, Academic/Industrial/NASA/Defense (WNN-AIND 91), Lawrence Livermore National Lab. Publ. UCRL-JC-105362.

Cheng, P. Z., 1999, Seismic event picking for velocity estimation using neural networks: Ph.D. thesis, Univ. of California, Davis.

Daugman, J. G., 1988, Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression: Inst. Electr. Electron. Eng. Trans. on Acoust., Speech and Signal Proc, **16**, 1169—1179.

Daugman, J. G., and Kammen, D. M., 1987, Image statistics, gases and visual neural primitives: First Internat. Conf. on Neural Networks, **4**, 163–175.

Devijver, P. A., and Kittler, J., 1982, Pattern recognition: A statistical approach: Prentice-Hall, Inc.

Fukunaga, K., 1990, Introduction to statistical pattern recognition: Academic Press Inc.

Gabor, D., 1946, Theory of communication: J. Inst. Electr. Eng., **93**, No. 36, 251–270.

Geerlings, A. C., and Berkhout, A. J., 1989, Heuristic event tracing linked to linear discriminant analysis, advances in geophysical data processing: JAI Press.

Haralick, R. M., and Shapiro, L. G., 1992, Computer and robot vision: Addison-Wesley Publ. Co.

Hogg, R. V., and Craig, A. T., 1978, Introduction to mathematical statistics: Macmillan Publ. Co.

Jain, A. K., 1989, Fundamentals of digital image processing: Prentice-Hall, Inc.

Kemp, F., Threet, J. R., and Veezhinathan, J., 1992, A neural net branch and bound seismic horizon tracker: 62nd Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 10–13.

Lu, S.-Y., 1982, A string-to-string correlation algorithm for image skeletonization: Joint Conf. on Pattern Recognition, Inst. Electr. Electron. Eng., Proceedings, 178–180.

Lu, S.-Y., and Cheng, Y.-C., 1990, An iterative approach to seismic skeletonization: Geophysics, **55**, 1312–1320.

McCormack, M. D., Zaucha, D. E., and Dushek, D., 1993, First-break refraction event picking and seismic data trace editing using neural networks: Geophysics, **58**, 67–78.

Morlet, J., Arens E., Fourgeau, E., and Giard, D., 1982a, Wave propagation and sampling theory, part I: Complex signal and scattering in multilayer media: Geophysics, **47**, 203–221.

——— 1982b, Wave propagation and sampling theory, part II: Sampling theory and complex waves: Geophysics, **47**, 222–236.

Parzen, E., 1962, On estimation of a probability density function and mode: Annals of Math. Stat., **33**, 1065–1076.

Robinson, E. A., and Treitel, S., 1980, Geophysical signal analysis: Prentice-Hall, Inc.

Specht, D. E., 1990a, PNN and polynomial adaline as complementary techniques for classification: Inst. Electr. Electron. Eng. Trans. Neural Networks, **1**, 111–121.

——— 1990b, Probabilistic neural networks: Neural Networks, **3**, 109–118.

Taner, M. T., 1988, The use of supervised learning in first break picking: Symposium, Geophys. Soc. of Tulsa, Proceedings,

Taner, M. T., Koehler, F., and Sheriff, R. E., 1979, Complex seismic trace analysis: Geophysics, **44**, 1041–1063.

Veezhinathan, J., and Wagner, D., 1990, A neural network approach to first break picking: Internat. Joint Conf. Neural Networks, Proceedings, 235–240.

Woodham, C. A., Sandham, W. A., and Durrani, T. S., 1995, 3-D seismic tracking with probabilistic data association: Geophysics, **60**, 1088–1094.

Young, T. Y., and Fu, K. S., 1986, Handbook of pattern recognition and image processing: Academic Press Inc.