

INTERPRETER-DRIVEN AUTOMATIC IMAGE
SEGMENTATION AND MODEL EVALUATION

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF GEOPHYSICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Adam D. Halpert

July 2014

© Copyright by Adam D. Halpert 2014
All Rights Reserved

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Biondo L. Biondi) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Jon F. Claerbout)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Gerald Mavko)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Robert G. Clapp)

Approved for the University Committee on Graduate Studies

Abstract

This dissertation addresses a major bottleneck in iterative seismic imaging projects: interpretation and modeling of large, subsurface salt bodies. These salt structures are ubiquitous in many of the world’s most active areas for seismic exploration, and are extremely time-consuming and difficult to interpret – a process that is conventionally undertaken manually, with little automation. In this thesis, I propose two tools that can help interpreters increase the degree of automation in the salt interpretation workflow, while allowing them to maintain control of the process through expert guidance and scenario testing. These tools are an interpreter-guided image segmentation algorithm, and an efficient, wavefield-based scheme to evaluate several potential velocity models.

The image segmentation method presented here is modified from a graph-partitioning algorithm designed to be both highly efficient and globally accurate when segmenting an image. To account for the unique nature of seismic images, changes to the algorithm’s input data, procedure for building graph edges, and subsequently for weighting those edges, are required. The modified algorithm is capable of performing accurate, fully automatic segmentations of salt bodies in 2D and 3D seismic images; however, a fully automatic approach can fail when the salt boundary is poorly imaged. In these cases, expert interpreter guidance can be supplied, either directly on the 2D image, or on one or more 2D slices of a 3D volume. In the 3D case, the interpreter’s picks are projected into the third dimension prior to the automatic 3D segmentation. By incorporating interpreter guidance, highly accurate segmentations of a 3D field image are obtained with limited manual intervention.

If multiple salt scenarios lead to several possible velocity models being created, these models can be quickly tested by synthesizing new source and receiver wavefields from an initial image. Both wavefields are generated using prestack velocity information from the initial image, via a generalized form of Born modeling. This allows the velocity inaccuracies present in the initial model to be identified and, ideally, corrected in future model iterations. Because the synthesized receiver wavefield is imaged with an areal source function (both obtained with the initial velocity model), a single shot can be migrated using any other velocity model to provide an image of targeted locations within the model. Crosstalk issues arising from interfering events in the subsurface offset domain can be mitigated by imaging only sparsely-spaced model locations; alternatively, several separate experiments can be performed at different model locations, and the resulting images summed to provide a more detailed final image. If qualitative inspection is insufficient to determine the most accurate model among those being tested, a measure of image focusing can provide a quantitative comparison based on the proportion of an image's energy focused at or near zero-subsurface offset. This strategy is shown to be effected for synthetic and field datasets, in 2D and 3D.

The image segmentation and model evaluation tools are designed to work together to alleviate the salt interpretation bottleneck in model building. Using a wide-azimuth survey from the Gulf of Mexico, image segmentation is used to isolate a sedimentary inclusion within a salt body, and to define two alternate interpretations of the base of salt. The efficient model evaluation scheme is then used to test these two models, along with one provided with the dataset, in a fraction of the time required for full migrations. Qualitative and quantitative analysis of the results indicates that one of the alternate models is most desirable, and a subsequent remigration of the full dataset with this model provides an image with improved clarity and continuity of subsalt reflectors.

Preface

The electronic version of this report¹ makes the included programs and applications available to the reader. The markings **ER**, **CR**, and **NR** are promises by the author about the reproducibility of each figure result. Reproducibility is a way of organizing computational research that allows both the author and the reader of a publication to verify the reported results. Reproducibility facilitates the transfer of knowledge within SEP and between SEP and its sponsors.

ER denotes Easily Reproducible and are the results of processing described in the paper. The author claims that you can reproduce such a figure from the programs, parameters, and makefiles included in the electronic document. The data must either be included in the electronic distribution, be easily available to all researchers (e.g., SEG-EAGE data sets), or be available in the SEP data library. We assume you have a UNIX workstation with Fortran, Fortran90, C, X-Windows system and the software downloadable from our website (SEP makerules, SEPScons, SEPlib, and the SEP latex package), or other free software such as SU. Before the publication of the electronic document, someone other than the author tests the author's claim by destroying and rebuilding all ER figures. Some ER figures may not be reproducible by outsiders because they depend on data sets that are too large to distribute, or data that we do not have permission to redistribute but are in the SEP data library, or that the rules depend on commercial packages such as Matlab or Mathematica.

¹<http://sepwww.stanford.edu/doku.php?id=sep:research:theses:sep153>

CR denotes Conditional Reproducibility. The author certifies that the commands are in place to reproduce the figure if certain resources are available. The primary reasons for the CR designation is that the processing requires 20 minutes or more.

NR denotes Non-Reproducible figures. SEP discourages authors from flagging their figures as NR except for figures that are used solely for motivation, comparison, or illustration of the theory, such as: artist drawings, scannings, or figures taken from SEP reports not by the authors or from non-SEP publications.

Our testing is currently limited to LINUX 2.6 (using the Intel Fortran90 compiler) and the SEPlib-6.4.6 distribution, but the code should be portable to other architectures. Reader's suggestions are welcome. For more information on reproducing SEP's electronic documents, please visit <http://sepwww.stanford.edu/>.

Acknowledgments

My time at Stanford and at the Stanford Exploration Project has been very rewarding, and I would like to thank Biondo Biondi, not only for welcoming me as part of the group, but for fostering an environment of research freedom and initiative that makes SEP unique. At the same time, his advice and guidance were invaluable to me on my journey. SEP's founder, Jon Claerbout, provided numerous ideas and insights that improved my research and my understanding of a wide range of geophysical topics. Bob Clapp's expertise in both scientific computing and the geophysical issues it can help address makes him irreplaceable at SEP. I would have been lost without his help, especially my first couple years. Finally, thanks to Diane Lau, who works behind the scenes to make everything at SEP run smoothly.

I would like to thank the SEP sponsors for their support over the years, and in particular those who have donated datasets I have used to test and improve my research: Unocal/Chevron and Schlumberger Multiclient. Field datasets are essential to SEP's mission, and I am grateful for the opportunity to test my ideas in real-world situations.

I will miss all my colleagues at SEP, including those who have graduated since my arrival: Sjoerd de Ridder, Xukai Shen, Mandy Wong, Yunyue Li, Chris Leader, Ohad Barak (who performs selflessly as our systems administrator), Yang Zhang, Ali Almomin, Yi Shen, Jason Chang, Musa Maharramov, Taylor Dahlke, Huy Le, Eileen Martin, Guillaume Barnier, Daniel Blatter, Gustavo Alves, Gboyega Ayeni, Yaxun Tang, Qiang Fu, Kittinat Taweasantanon, Claudio Cardoso, Nader Moussa, Abdullah al Theyab, Mohammad Maysami, Bill Curry, Alejandro Valenciano, Guojian

Shan, Roland Gunther, and Jeff Shragge. I have learned a great deal from all of you, about geophysics and beyond. I am especially grateful to those who provided babysitting services my last few months here.

Last but not least, I need to thank my family. My parents and brother have been very supportive over the years, ever since I first tried to explain to them what geophysics is. My wife Claire has been amazingly patient and wonderfully supportive – thank you. And finally, I would like to dedicate this thesis to my son Evan, whose birth provided just the inspiration I needed to finish.

Contents

Abstract	vi
Preface	viii
Acknowledgments	xi
1 Introduction	1
2 Interpreter-guided seismic image segmentation	13
3 Efficient velocity model evaluation	51
4 Integrated model building workflow	91
Bibliography	107

List of Tables

3.1	Calculations from equation 3.5 for each migration velocity model, after the initial image and synthesized wavefields were created using the true velocity model.	72
3.2	Calculations from equation 3.5 for each migration velocity model, after the initial image and synthesized wavefields were created using the “slow salt” velocity model.	75
3.3	Calculations from equation 3.5 for each 2D migration velocity model, after the initial image and synthesized wavefields were created using the provided velocity model.	79
3.4	Calculations from equation 3.5 for each 2D migration velocity model, after the initial image and synthesized wavefields were created using a model 5% slower than the one provided.	79
3.5	Calculations from equation 3.5 for each 3D migration velocity model, after the initial image and synthesized wavefields were created using the velocity model provided with the data.	82
3.6	Calculations from equation 3.5 for each 3D migration velocity model, after the initial image and synthesized wavefields were created using a velocity model 5% faster than the one provided with the data.	82

4.1	Calculations from equation 3.5 for each migration velocity model in Figure 4.3, after the initial image and synthesized wavefields were created using the initial velocity model.	100
-----	---	-----

List of Figures

1.1	Slices through a 3D seismic image cube featuring a salt body in the Gulf of Mexico. Even though the salt body is relatively well-imaged, portions of the boundary and subsalt reflectors are missing or discontinuous. [ER]	3
1.2	An automatic horizon-tracker attempting to follow the salt boundary on a seismic image from the Gulf of Mexico, using (a) two, (b) three, and (c) four manually-placed seed points indicated in red on each figure. Even using four seed points on this relatively prominent boundary, the algorithm struggles to accurately track the boundary. [ER]	5
1.3	The same image used in Figure 1.2, with the salt body automatically delineated using a modified version of the Pairwise Region Comparison (PRC) scheme from Felzenszwalb and Huttenlocher (2004). This result was obtained with negligible computational expense and with less human interaction than was required for even the inaccurate results in Figure 1.2. [ER]	7
1.4	Slices through the same image as in Figure 1.1, with two different velocity models overlaid: (a) the original model provided with the data and image; and (b) an alternate model obtained via image segmentation techniques. The alternate model features a modeled inclusion within the salt body, and a different base salt interpretation. [ER]	9

2.1	Modified from Zahn (1971). A graph with weighted edges (a); a spanning tree of that graph (b); and the minimum spanning tree of the graph (c). [NR]	17
2.2	Illustrations of three crucial definitions for PRC segmentation, on two groups of pixels that may be separated by a boundary. (a) The <i>Internal difference</i> of a group of pixels is the largest edge weight between two pixels in that group; (b) the <i>Minimum internal difference</i> is the smaller of the internal differences from each group; (c) the <i>Difference</i> is the smallest edge weight between two pixels in different groups. [NR]	18
2.3	(a) A 2D field seismic image, and (b) its corresponding segmentation using the original algorithm from Felzenszwalb and Huttenlocher (2004). Prior to modification, the algorithm performs poorly because of the unique characteristics of seismic data. [NR]	19
2.4	Amplitude of the envelope calculation for the raw image in Figure 2.3(a). This becomes the input for the PRC segmentation algorithm. [ER]	21
2.5	(a) A 3D image from the Gulf of Mexico; (b) The result of smoothing the image in panel (a) with a standard box-filter. [ER]	23
2.6	Results of differencing the original image in Figure 2.5(a) with a smoothed version obtained using (a) standard box-filtering, and (b) directional maximum homogeneity filtering. Both examples use the same size operator (box side length or bar mask length, respectively). [ER]	24
2.7	Results of applying the hybrid-MH filter on the image in Figure 2.5(a), with α set at (a) 0.5, (b) 0.2, and (c) 0.1 and (d) 0.01. While the reflector amplitudes are affected, a great deal of speckle noise is removed at low α values. [ER]	25

2.8	Stencils used for comparing pixel values and assigning edge weights for the graph corresponding to a 2D image. At left, the five-point stencil (8 edges per pixel) used in the original implementation from Felzenszwalb and Huttenlocher (2004); at right, a modified 21-point stencil (40 edges per pixel) used for the seismic images. For 3D images, additional stencil arms extend into the third dimension; the length of the stencil arms is a user-defined parameter. [NR]	27
2.9	Diagram illustrating the logic behind deciding which pixel intensity value to use when calculating edge weights. Pixel intensities along one “arm” of the stencil in Figure 2.8 are shown on the left; darker colors represent higher intensities. The right column indicates which intensity value will be used when calculating the edge weight between the “active” (red) pixel and the adjacent pixel. [NR]	28
2.10	Final segmentation result of the image in Figure 2.3(a), after modifications to the algorithm. [ER]	30
2.11	Automatic segmentation results for the image in Figure 2.3(a), using three different lengths for the stencil arms depicted in Figure 2.8: (a) 10 pixels; (b) 5 pixels; (c) 3 pixels. [ER]	33
2.12	A 2D section from a Gulf of Mexico dataset. The prominent salt body provides an ideal test case for segmentation algorithms. [ER]	35
2.13	(a) A segmentation of the image in Figure 2.12 according to the algorithm presented in section 2. The salt body is divided into several segments, so an interpreter can select which segments (denoted by the red dot) belong with the salt body. (b) Result after merging the segments marked in panel (a). It is now much easier to properly evaluate the segmentation results. [ER]	37
2.14	Manual salt picks supplied to guide the automatic segmentation. These picks are designed to correct the leakages seen on the segmentation result in Figure 2.13(b). [ER]	38

2.15	Segmentation of Figure 2.12 after incorporation of interpreter guidance. [ER]	39
2.16	Depiction of the pyramid used to “project” an interpreter’s picks from a single 2D slice into the third dimension. The influence of the pick (at the apex of the pyramid) decays with distance. [NR]	40
2.17	(a) Slices through a 3D image cube from the Gulf of Mexico; (b) Segmentation result prior to interpreter guidance. [ER]	42
2.18	Manually-interpreted salt picks used to guide the automatic 3D segmentation. [ER]	43
2.19	Amplitude of the envelope volumes (a) before and (b) after modification according to the interpreter guidance scheme. [ER]	44
2.20	Segmentation result incorporating interpreter guidance. While not perfect, this result is much more accurate than the initial result in Figure 2.17(b). [ER]	45
2.21	Amplitude volumes at a location far from any manual picks both (a) before and (b) after modification via 3D interpreter guidance. The effects of the interpreter guidance scheme can be seen along the salt body boundaries, even though the interpreter’s picks are farther away. [ER]	46
2.22	Segmentation results for another set of slices through the 3D cube, (a) before and (b) after interpreter guidance. The result is greatly improved, even far from where the manual salt picks were supplied. [ER]	47
2.23	Salt boundary identified on a 2D image using two automatic segmentation techniques: NCIS (green/light color) and PRC (red/dark). [NR]	49

3.1	Recorded source wavefields that have been reverse-propagated to zero-time; the result in (a) does not include information from the nonzero subsurface offsets of the initial image, while (b) does include this information. Both the initial migration and the modeling used a velocity model that was 15% too slow. [ER]	55
3.2	Prestack depth migration images of (a) a flat reflector and (b) a reflector dipping at a 20° angle. The images were migrated with a constant velocity 15% too slow compared to the true velocity, causing the noticeable artifacts and lack of focusing in the subsurface offset dimension. [ER]	59
3.3	Isolated image points from Figures 3.2(a) and 3.2(b) used for the modeling procedure. The points are separated by twice the maximum subsurface offset value in order to avoid crosstalk artifacts in the modeling. [ER]	60
3.4	Migrated images after Born modeling using the images in Figure 3.3 as the reflectivity model. Although the amplitudes differ, the kinematics of the events in both figures match. [ER]	61
3.5	Migration result using Born-modeled data from the model in Figure 3.3(b). In this case, the synthesized data was recorded in the subsurface instead of on the surface, effectively re-datuming the wavefields. [ER]	62
3.6	Result of migrating the Born-modeled data with (a) 5% too slow velocity; (b) correct velocity; and (c) 5% too fast velocity. Each migration was nearly instantaneous, and the effects of the different velocity models are readily apparent. [ER]	64
3.7	Migration result if the image points sampled from Figure 3.2(a) are spaced at less than twice the maximum subsurface offset. Crosstalk artifacts dominate the image, making interpretation extremely difficult. [ER]	65

3.8	Migration results using (a) four and (b) three image locations sampled from the flat reflector in Figure 3.2(a), and (c) the sum of these two images. Taken together, the sample points in panels (a) and (b) are identical to those used to produce the result in Figure 3.7; however, the image in panel (c) is free of the crosstalk artifacts seen in that result. [ER]	66
3.9	(a) Initial image of a 40° dipping reflector; (b) crosstalk-contaminated result of using a single source function derived from closely spaced locations along the reflector; (c) result of summing several images generated using wavefields synthesized from properly-spaced locations along the reflector; (d) full reconstruction of the reflector after using all locations along the reflector to synthesize wavefields and summing the individual results. [CR]	68
3.10	Results of imaging wavefields synthesized using every location along the reflector in Figure 3.9(a), but restricting the use of subsurface offset information to three offsets only ($h = 0, \pm 200$). In (a), the slow velocity was used to migrate the wavefields, while the true velocity was used in (b). [CR]	69
3.11	(a) A true velocity image using data from a section of the Sigsbee synthetic model; and (b) a base-of-salt reflector selected for further analysis because of its sensitivity to changes in the salt interpretation. [CR]	71
3.12	Isolated image locations selected from the reflector picked in Figure 3.11(b). [CR]	72
3.13	Two different velocity models to be tested. The model in (a) is the true Sigsbee model, while (b) is an alternate model created by one possible automatic segmentation of the initial image. [ER]	73

3.14	Three images of the Born-modeled data using three different migration velocity models: (a) the true model; (b) the extra-salt model; and (c) the slow-salt model. In this example, the initial image was created with the true model. [CR]	74
3.15	Three images of the Born-modeled data using three different migration velocity models: (a) the true model; (b) the extra-salt model; and (c) the slow-salt model. In this example, the initial image was created with the slow-salt model. [CR]	76
3.16	(a) A 2D field image produced using the provided velocity model, and (b) a manually-picked horizon of interest used to test three different velocity models. [CR]	77
3.17	A single location from the reflector indicated in Figure 3.16(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% faster than the one provided; (b) the provided velocity model; and (c) a model 5% slower than the one provided. [CR] . . .	78
3.18	(a) A 2D field image produced using a velocity model 5% slower than the one provided, and (b) a manually-picked horizon of interest used to test three different velocity models. [CR]	80
3.19	A single location from the reflector indicated in Figure 3.18(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% slower than the one provided; (b) the provided velocity model; and (c) a model 5% faster than the one provided. [CR]	81
3.20	(a) A 3D field image produced using the provided velocity model, and (b) a manually-picked horizon of interest used to test three different velocity models. [CR]	83

3.21	A single location from the reflector indicated in Figure 3.16(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% faster than the one provided; (b) the provided velocity model; and (c) a model 5% slower than the one provided. [CR] . . .	84
3.22	Subsurface offset images extracted from the indicated locations in Figure 3.21. The images were created using (a) the slower velocity; (b) the provided velocity; and (c) the faster velocity. [CR]	85
3.23	(a) A 3D field image produced using a velocity model 5% faster than the one provided, and (b) a manually-picked horizon of interest used to test three different velocity models. [CR]	86
3.24	A single location from the reflector indicated in Figure 3.23(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% faster than the one provided; (b) the provided velocity model; and (c) a model 5% slower than the one provided. [CR] . . .	87
3.25	Subsurface offset images extracted from the indicated locations in Figure 3.24. The images were created using (a) the faster velocity; (b) the provided velocity; and (c) the slower velocity. [CR]	88
4.1	(a) A 3D image from the Gulf of Mexico (data courtesy of Schlumberger Multiclient) obtained via one-way migration with the velocity model shown in (b). A prominent sediment inclusion within the salt body, and/or a misinterpreted base of salt, may contribute to the subsalt reflectors' loss of continuity. [CR]	93
4.2	(a) Close-up view of the sediment inclusion first seen in Figure 4.1(a); (b) Interpreter-guided 3D segmentation of the image in (a). [CR] . .	95
4.3	(a) The velocity model provided with the data; (b) An updated model based on the segmentation result in Figure 4.2(b) (and another defining an alternative base-salt interpretation). [CR]	96

4.4	A manually-selected base-salt reflector that will be used to quickly evaluate the velocity models in Figure 4.3. [CR]	98
4.5	Results of performing multiple rounds of the model evaluation procedure on different locations along the reflector indicated in Figure 4.4, and summing the results. The velocity models used for the final imaging step correspond to those in Figure 4.3: (a) the original velocity model; (b) an alternate model with aggressive removal of salt; and (c) an alternate model with more conservative salt removal. The arrows indicate locations at which the subsurface offset panels in Figure 4.6 were extracted. [CR]	99
4.6	Subsurface offset panels at a single x, y, z location indicated by the arrows in Figure 4.5 for the image corresponding to (a) the original velocity model; and alternate models with (b) aggressive and (c) more conservative removal of salt. The greater degree of focusing near zero subsurface offset in (c) suggests a more accurate velocity model. [CR]	100
4.7	Full, one-way migrations with identical parameters using (a) the original velocity model, and (b) the updated model. [CR]	102
4.8	Same as Figure 4.7, but with areas of interest indicated. The deep subsalt reflectors and the base-salt reflector show particular improvement on the image generated with the new velocity model (b). [NR]	103
4.9	Images at a second location, obtained using (a) the original velocity model, and (b) the updated model. [CR]	104
4.10	Same as Figure 4.9, but with areas of interest indicated. At this location, the subsalt reflectors show improved continuity and the salt inclusion is more accurately depicted on the image generated with the new velocity model (b). [NR]	105

Chapter 1

Introduction

An accurate velocity model of the subsurface is a crucial component of successful seismic imaging projects. Unfortunately, producing such a model is also one of the most time-consuming and difficult aspects of the imaging workflow. For several reasons, this is especially true in areas characterized by large subsurface salt bodies. First, salt is relatively fluid or ductile compared to the sedimentary formations within which salt is normally encountered. This creates a wide variety of shapes and formations as the salt moves and deforms throughout the subsurface. For example, salt structures often feature extremely steep dips along their flanks, making them difficult to image (and subsequently model) using standard seismic migration techniques. A second reason salt modeling can be so time-consuming is the extreme care necessary to ensure that salt interpretation is as accurate as possible. Seismic waves travel at a much greater velocity (often over twice as fast) through salt than they do through the sedimentary rocks that typically surround salt bodies. The sharp velocity contrast leads to a large reflection coefficient, and the resulting scattering of seismic energy, along the interface between salt and sediments. Even a slight error in the interpreted location of this interface can have a disproportionate impact on calculations of wave propagation beneath the interface, and lead to significant degradation of the resulting image. To combat this, salt modeling is often carried out in an iterative fashion, with interpretations of the top and base of salt bodies being performed on one or more

images created using sediment- or salt-flood velocity models, respectively (Mosher et al., 2007). The iterative nature of the process, combined with the challenges interpreters face when salt bodies are not clearly imaged, leads to salt modeling workflows that can stretch for weeks or even months for the enormous 3D seismic surveys that are standard in today’s exploration environment. Thus, salt modeling represents a tremendous bottleneck in the backbone workflow of a multi-billion dollar industry.

At the same time, exploration for oil and gas is becoming increasingly focused in areas of the world dominated by salt bodies, and in reservoirs existing below those salt bodies. Notable examples include offshore Brazil and West Africa, the North Sea, and the Gulf of Mexico, which is the origin of all the field data used in this thesis. Increased exploration in these salt-dominated areas is not a coincidence; in much of the world, the “easy oil” in reservoirs above salt has been located and produced, leaving the more challenging reservoirs below salt as prime targets for imaging and exploration. Furthermore, the presence of salt itself can sometimes be an indicator of possible resource deposits. As it moves through and deforms the subsurface, salt helps create structural traps for oil and gas, and in addition acts as an impermeable seal for those reservoirs (Hudec and Jackson, 2007). An indication of the extent to which salt tectonics can dominate the geology of a region can be found in the bathymetry of the transition zone between the continental shelf and deepwater in the Gulf of Mexico. As Pilcher et al. (2011) demonstrate, the extreme bathymetric variations found there are a result of subsurface salt movements – as salt bodies flow and retreat, they deform the overlying sediments to such an extent that their surface expression is clearly visible. These irregularly shaped salt bodies, like the one shown in Figure 1.1, are difficult to interpret even when well-imaged, and contribute to the model-building bottleneck described above.

The goal of my thesis is to alleviate this bottleneck by introducing two model building tools. The first of these tools is interpreter-guided seismic image segmentation, which is a semi-automatic method of delineating salt bodies like the one in Figure 1.1. The method is designed to incorporate valuable interpreter insight from 2D sections into an automatic 3D segmentation, thus combining humans’ pattern

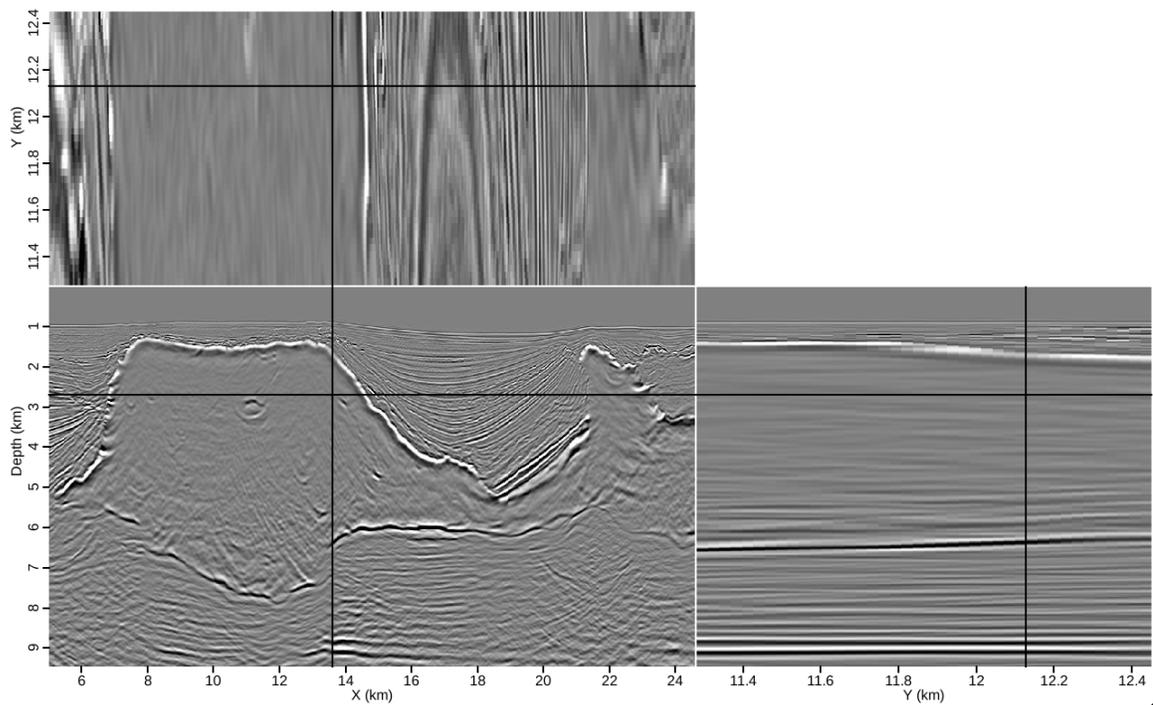


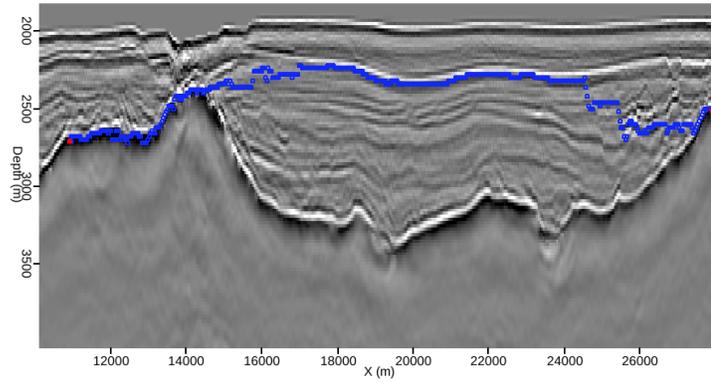
Figure 1.1: Slices through a 3D seismic image cube featuring a salt body in the Gulf of Mexico. Even though the salt body is relatively well-imaged, portions of the boundary and subsalt reflectors are missing or discontinuous. [ER] chap1/. oct-3di

recognition strengths in 2D with ever-increasing computational capabilities in 3D. The second tool is a method for quickly and efficiently evaluating the relative accuracy of two or more potential velocity models. Such a method is useful if, for example, an ambiguous base salt interpretation leads to several possible salt “scenarios”. The efficient model evaluation scheme can evaluate these models for a tiny fraction of the expense required for a full migration of the data for each model. In the following sections and chapters, I will describe the background and theory behind both of these tools, and provide examples using both synthetic and field seismic data, in 2D and 3D. Finally, in Chapter 4, I will demonstrate how these tools can work together to create and test alternate velocity models for a 3D dataset from the Gulf of Mexico, and identify a model that leads to an improved subsalt image.

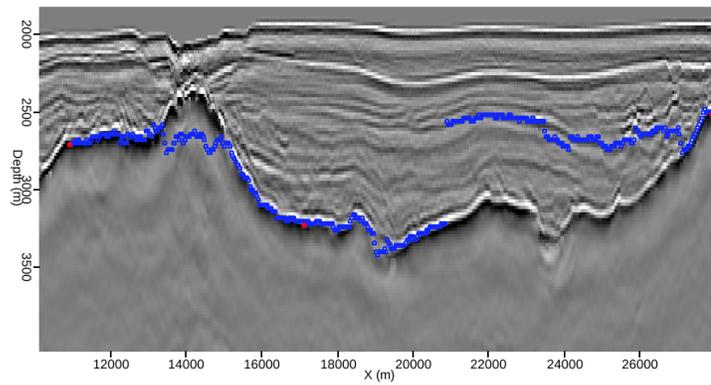
IMAGE SEGMENTATION

One way to streamline salt interpretation is to automate some of its processes. Automating some aspects of seismic interpretation is not a new idea; automatic horizon trackers are ubiquitous in seismic processing and interpretation software packages, and are often effective for relatively flat and/or highly continuous reflectors. However, “flat” and “continuous” are words rarely used to describe the imaged boundaries of salt bodies, and automatic pickers can struggle if they encounter local discontinuities along a reflector. Figure 1.2 provides one such example. Even though the salt boundary in this image is relatively prominent, the automatic picker fails to accurately track the entire boundary even when multiple seed points (in red) are supplied by an interpreter. It follows that salt interpretation would benefit from algorithms that take a more global approach to the problem, rather than relying solely on local attributes along a reflector. *Image segmentation* algorithms fit this description.

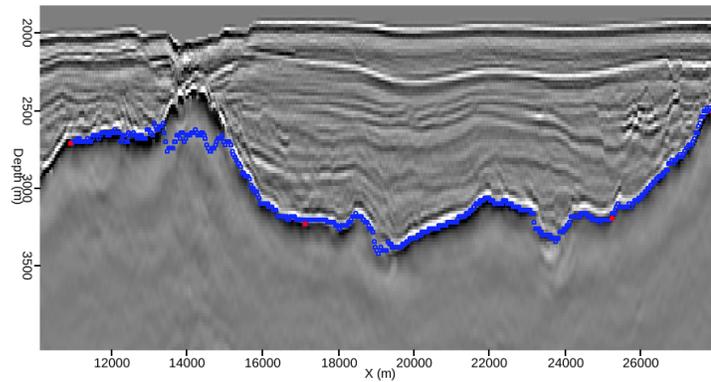
The term “image segmentation” refers to the process of automatically detecting non-overlapping regions within an image based on certain characteristics. These characteristics vary depending on the type of image in question. For example, color attributes are popular for algorithms aimed at segmenting photographs (Cheng et al.,



(a)



(b)



(c)

Figure 1.2: An automatic horizon-tracker attempting to follow the salt boundary on a seismic image from the Gulf of Mexico, using (a) two, (b) three, and (c) four manually-placed seed points indicated in red on each figure. Even using four seed points on this relatively prominent boundary, the algorithm struggles to accurately track the boundary. [ER] chap1/. seed2,seed3,seed4

2001); such algorithms are an active area of research due to their usefulness in photo processing and for machine learning purposes. Another field in which image segmentation is widely used is medical imaging, where the proliferation of diagnostic imaging in modalities such as magnetic resonance (MRI) and computed tomography (CT) have made some form of automation highly desirable (Pham et al., 2000). Within the past decade, a similar increase in demand for interpretation resources has led geophysical researchers to recognize the potential of image segmentation for seismic images.

Image segmentation algorithms can be grouped into families based on the methods employed, and several families of methods have been applied to seismic images. These include pixel-based schemes (Valet et al., 2001; Berthelot et al., 2012) and fast-marching methods based on level sets (Osher and Sethian, 1988). The latter method has attracted interest based on its utility both for standard salt segmentation (Haukas et al., 2013), and in conjunction with velocity estimation using the Full Waveform Inversion (FWI) objective function (Lewis et al., 2012; Dahlke et al., 2014). However, the class of algorithms which has proven most popular for seismic image segmentation, and to which the method presented in this thesis belongs, is called graph-based image segmentation. In these methods, pixels in the image are treated as nodes of a graph, which are connected via various edges. The graph (and thus the image) is partitioned by making one or more optimal cuts through the graph's edges. Early efforts using these methods focused on atomic meshing of seismic images (Hale and Emanuel, 2003, 2002), while later work adapted the Normalized Cuts Image Segmentation (NCIS) algorithm (Shi and Malik, 2000) for tracking 3D salt boundaries (Lomask, 2007; Lomask et al., 2007). This approach was effective, but faced computational hurdles because of the need to calculate eigenvectors from large, 3D images. In contrast, the method presented here, adapted from the Pairwise Region Comparison (PRC) scheme of Felzenszwalb and Huttenlocher (2004), is designed to be extremely computationally efficient. This allows us to obtain results like the one in Figure 1.3, in which the salt body has been automatically identified much more accurately than in the horizon-tracking results in Figure 1.2, at negligible computational expense. In the next chapter, I will describe how to modify the original PRC algorithm to account for the unique nature of seismic images, and how to incorporate interpreter guidance

into the process when necessary.

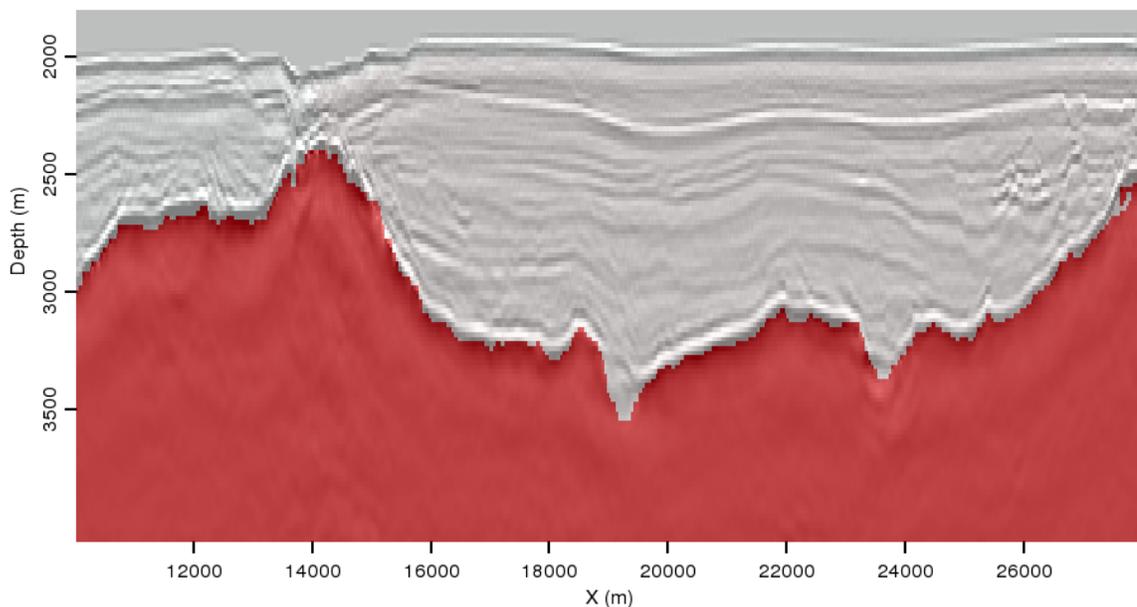


Figure 1.3: The same image used in Figure 1.2, with the salt body automatically delineated using a modified version of the Pairwise Region Comparison (PRC) scheme from Felzenszwalb and Huttenlocher (2004). This result was obtained with negligible computational expense and with less human interaction than was required for even the inaccurate results in Figure 1.2. [ER] chap1/. uno-prc

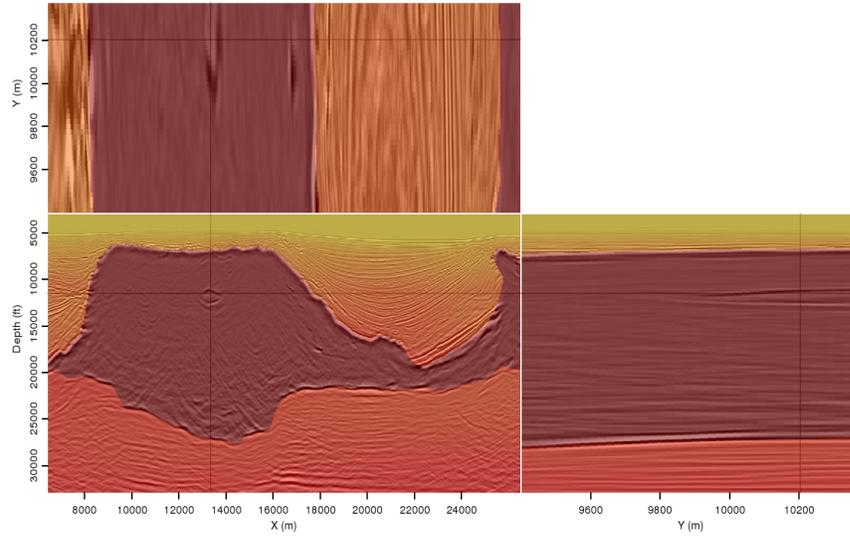
EFFICIENT MODEL EVALUATION

Velocity model building is rarely a straightforward exercise, especially when salt is involved. For example, Figure 1.4(a) shows a velocity model provided with the underlying image (from the same 3D image cube seen in Figure 1.1). Much of the salt interpretation is unambiguous, especially along the top salt reflector. However, a couple aspects of the model suggest room for improvement. First, a highly noticeable inclusion within the salt body has not been modeled with lower, sediment-like velocity. And second, the base salt reflector is much more ambiguous than the top salt in several locations, allowing for a range of possible interpretations. Either or both of these issues could be affecting the quality of the image subsalt, leading to the fading

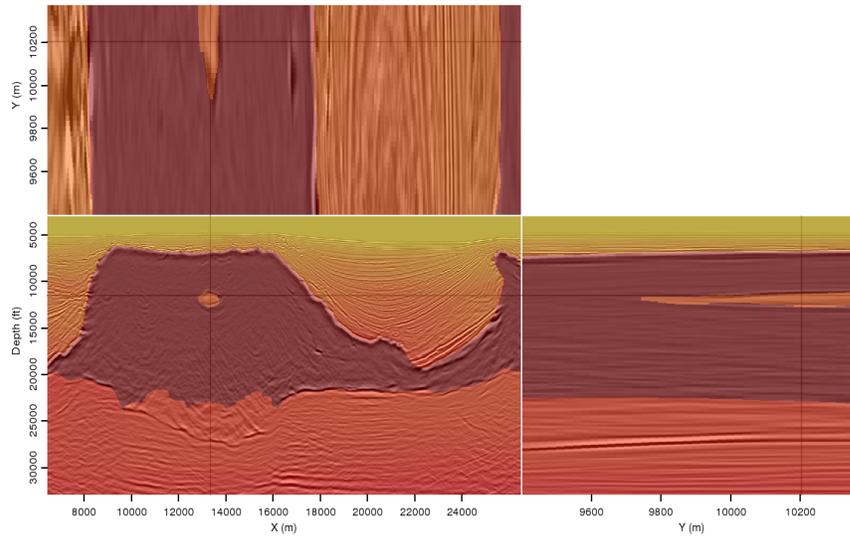
or discontinuous reflectors beneath the salt body. To investigate, an interpreter could create an alternate model like the one seen in Figure 1.4(b), either manually or, in this case, with the aid of tools such as image segmentation. The question becomes, what is the best way to determine which of these models is more accurate?

The simplest way to gauge the accuracy of a velocity model is to obtain an image by migrating the data using that model. However, this approach quickly becomes infeasible for very large datasets and/or multiple models. For example, the images shown here are from a wide-azimuth survey from the Gulf of Mexico, provided by Schlumberger Multicient. Wide-azimuth surveys are beneficial because the increased azimuthal coverage allows for better illumination and (ideally) higher-quality images beneath salt bodies. These advantages come at the expense of more manageable dataset sizes, however. This particular dataset is almost 8 terabytes in size, with over 700 million data traces. The goal of this thesis is to alleviate model building bottlenecks, but the ability to quickly create velocity models using image segmentation does little to achieve this goal if testing these models requires repeated migrations of such large data volumes.

In Chapter 3 of this thesis, I propose the idea of using wavefields synthesized from an initial image to more efficiently test velocity models. The goal of this procedure is similar to others (Gao et al., 2006; Wang et al., 2008) which rely on a fast variant of Gaussian beam migration (Hill, 1990, 2001). However, using wavefield propagation methods allows for a more physically realistic solution, while avoiding frequency assumptions inherent to beam imaging. Wavefield synthesis via Born modeling (Stolt and Benson, 1986) has been used previously to successfully address imaging problems in a target-oriented fashion (Tang, 2011; Tang and Biondi, 2011), which is similar to the strategy presented here. In addition, this method takes advantage of the prestack exploding reflector concept (Biondi, 2006; Guerra, 2010; Guerra and Biondi, 2011) to incorporate prestack velocity information from an initial image into an areal source function used to synthesize new receiver wavefields. This creates the opportunity to identify and correct errors in the initial velocity model. Most importantly, this method is highly efficient; by targeting specific locations along a single reflector, it



(a)



(b)

Figure 1.4: Slices through the same image as in Figure 1.1, with two different velocity models overlaid: (a) the original model provided with the data and image; and (b) an alternate model obtained via image segmentation techniques. The alternate model features a modeled inclusion within the salt body, and a different base salt interpretation. [ER] chap1/. vmod0,vmod1

can produce both qualitative and quantitative indications of a model's accuracy in a fraction of the time needed for a full migration of the original dataset.

THESIS OVERVIEW AND CONTRIBUTIONS

Interpreter-guided seismic image segmentation: In Chapter 2, I introduce the concept of image segmentation and review some of its past applications to geophysical images. I then describe the Pairwise Region Comparison (PRC) graph-partitioning algorithm of Felzenszwalb and Huttenlocher (2004), and explain its potential for use in automatic salt body segmentation. Because seismic images are quite distinct from the more conventional images for which the PRC algorithm was originally designed, I detail the adaptations and changes necessary to make the algorithm more suitable for seismic image segmentation. These changes include pre-processing of images via edge-preserving smoothing, transformation of the input data, and modifications to the way graph edges are constructed and weighted prior to partitioning. Finally, I explain the motivation and strategy for incorporating interpreter input into the segmentation process. This involves having interpreters provide limited 2D salt boundary picks, and using those picks to guide an automatic 3D segmentation. Throughout the chapter, I use 2D synthetic data from the Sigsbee 2a model, as well as 2D and 3D field data images from the Gulf of Mexico, to demonstrate these concepts.

Efficient velocity model evaluation: In Chapter 3, I introduce a target-oriented wavefield synthesis method for quickly and efficiently evaluating velocity models. After providing the motivation and background for such a procedure, I detail each of the required steps. These include generating an areal source function from an initial image with prestack velocity information; using the new source function to synthesize a receiver wavefield via Born modeling, again using the initial image as a reflectivity model; and imaging the synthesized wavefields to produce a new image of locations along a target reflector. As long as the first two steps are carried out using an identical, initial velocity model, the final step

may use any other model to produce the final image. This is what allows for the fair and efficient comparison of multiple models. Next, I explain that one limitation of this approach is the potential for crosstalk artifacts to contaminate the results, and propose a strategy for mitigating these artifacts. This strategy requires imaging only sparsely-spaced locations along a reflector of interest, preventing overlapping events in the subsurface offset domain from contaminating the model evaluation experiment. Even when following this approach, a clearer picture of the reflector can be obtained by synthesizing several wavefields using appropriately-spaced locations, and summing the resulting images into a final image. Finally, I introduce an image-focusing measure that can provide a quantitative measure of the relative accuracy of velocity models being tested. After demonstrating the basic procedure using single-reflector synthetics, I show its applicability to the more complicated Sigsbee 2a model, and 2D and 3D datasets from the Gulf of Mexico. Importantly, I show that even when an incorrect velocity model is used to generate the initial image, this method can be used to identify a more accurate model.

Integrated model building tools: In Chapter 4, I show how the two tools I introduce in this thesis, interpreter-guided image segmentation and efficient velocity model evaluation, can be used in an integrated fashion to produce an improved velocity model and image. After presenting an image obtained from a wide-azimuth 3D survey in the Gulf of Mexico provided by Schlumberger Multicient, I introduce the velocity model used to obtain that image, and discuss areas of potential improvement. Specifically, the initial model does not incorporate a sedimentary inclusion within the salt body that is clearly visible on the image, and features an ambiguous base salt interpretation. I then demonstrate how automatic image segmentation can be used to isolate the salt body inclusion, and generate two additional interpretations of the base salt reflector. Alternate velocity models are easily created from the initial model by replacing salt velocities with appropriate sedimentary velocities according to the segmentation

results. Next, I use the efficient model evaluation scheme to judge the relative accuracy of the three potential models. By summing the results of multiple wavefield synthesization and imaging experiments, I am able to quickly produce partial images of the base salt reflector for each model. Qualitative examination of subsurface offset panels, as well as quantitative calculations based on the image focusing measure introduced in the previous chapter, suggest that an alternate model with a conservative removal of salt compared to the originally interpreted salt body is the most accurate model. Finally, I validate these results by showing that a full migration of the original data with this model results in an improved image, most notably characterized by improved continuity of subsalt reflectors.

Chapter 2

Interpreter-guided seismic image segmentation

Salt interpretation is a vital component of seismic imaging projects in many of the world's resource-rich basins. The sharp contrast between seismic velocities within salt structures and those in the surrounding sediments means that inaccurate interpretation of these salt-sediment boundaries can lead to severe degradation of images sub-salt; this is of particular concern since sub-salt reservoirs are often the targets for modern exploration. Unfortunately, salt interpretation is not only critical, but often time-consuming and human-intensive as well. For large 3D surveys, manual salt-picking can consume significant resources during model-building workflows that stretch for weeks or months. This can be exacerbated by iterative sediment- and salt-flooding techniques that require several rounds of salt interpretation (Mosher et al., 2007). The semi-automatic image segmentation method presented here aims to help alleviate this bottleneck, while maintaining the accuracy necessary for successful model building and imaging.

SEGMENTATION METHODS

While image segmentation is most often associated with fields such as medical imaging and photo processing, several efforts have been made to apply automatic segmentation concepts to seismic images. A variety of approaches has been tried, including pixel-by-pixel classifier methods using fuzzy math (Valet et al., 2001) or texture attributes (Berthelot et al., 2012). These methods can incorporate interpreter input by “training” the algorithm through the use of if-then guidelines or training images. Another category of methods that has proven popular for seismic images is known as graph-based image segmentation. In this method, each pixel in a seismic image is treated as a node or vertex in a graph; then edges are constructed between specific pixels and weighted according to some property. Image segments are created by partitioning the graph (for example, a partition may represent a salt boundary). An advantage of graph-based segmentation is that it provides a globally optimum solution to the segmentation problem. This compares favorably with automatic interpretation tools such as horizon trackers that tend to get “lost” if a boundary becomes chaotic or discontinuous. Recall the example from the previous chapter (Figure 1.2), in which one such horizon tracker fails to follow a prominent salt boundary, even when several seed points are provided.

The first graph partitioning seismic image segmentation algorithms were adapted from the eigenvector-based Normalized Cuts Image Segmentation (NCIS) method (Shi and Malik, 2000). This method is based on transforming an image into a normalized eigenvector, with values ranging from -1 to 1 across a boundary. One of the first applications was for atomic meshing of seismic images (Hale and Emanuel, 2002, 2003), followed by efforts to track salt boundaries (Lomask et al., 2007; Lomask, 2007; Halpert et al., 2009). The method was effective, but faced limitations - most notably computational. The NCIS algorithm requires the calculation of eigenvectors for an edge weight matrix of size n^2 , where n is the number of pixels in the image; this matrix quickly grows very large, especially for 3D surveys. Computationally, calculation of eigenvectors for such a large matrix is an extremely demanding task. As such, this method is limited to relatively small images; alternatively, we can restrict the

computational domain to a specific region around a previously interpreted boundary. However, this means the method is of limited utility if there is no “best guess” model available, or if the accuracy of that model is in question.

PRC SEGMENTATION

Thus, a more efficient global segmentation scheme that can include the entire image in the computational domain would be a very useful tool for interpretation of seismic images. One candidate for such a scheme is the algorithm from Felzenszwalb and Huttenlocher (2004), who write:

“Our algorithm is unique, in that it is both highly efficient and yet captures non-local properties of images.”

These two features are crucial for the task of seismic image segmentation. The algorithm is designed to run in $O(n \log n)$ time, where n is the number of pixels in the graph; in contrast, other methods such as NCIS require closer to $O(n^2)$ time to run. This represents a significant cost savings, especially for very large 3D seismic datasets that are becoming increasingly common.

The algorithm proposed by Felzenszwalb and Huttenlocher (2004) relies heavily on the concept of the “Minimum Spanning Tree” [see Zahn (1971)]. To understand the minimum spanning tree, a brief overview of graph-partitioning theory is helpful. Recall that a graph is composed of vertices (an image’s pixels) connected by edges. The graph’s edges may be weighted using a measure of similarity or dissimilarity between vertex pairs; for example, a simple weighting could simply difference intensity values at the two endpoints of an edge. A *connected* graph (example (a) in Figure 2.1) is defined as one in which all such edges are assigned a weight value. A spanning tree (Figure 2.1(b)) is a connected graph which connects all its vertices without closing a circuit. Finally, the minimum spanning tree (MST) of a graph is the spanning tree with the minimum sum of edge weights (Figure 2.1(c)). In Zahn (1971), partitioning of a graph was achieved simply by cutting through edges with large weights. However,

this approach is inadequate for images with coherent regions that are nonetheless highly heterogeneous (for example, the heterogeneous nature of the intensity values within a salt body). However, the MST concept allows Felzenszwalb and Huttenlocher (2004) to develop what they term a “pairwise region comparison” predicate, based on three definitions described below and illustrated in Figure 2.2. The *internal difference* of a region (C) in the graph to be the largest edge weight of the MST of that region:

$$\text{Int}(C) = \max_{e \in \text{MST}} w(e), \quad (2.1)$$

where e is a graph edge and $w(e)$ is the edge’s weight, defined according to some simple algorithm. In Figure 2.2(a), there are two groups of pixels that the algorithm must either separate or merge. The internal difference is computed for each group. When comparing two regions (such as C_1 and C_2), the *minimum* internal difference for the two regions is

$$M_{\text{Int}}(C_1, C_2) = \min(\text{Int}(C_1) + \tau(C_1), \text{Int}(C_2) + \tau(C_2)), \quad (2.2)$$

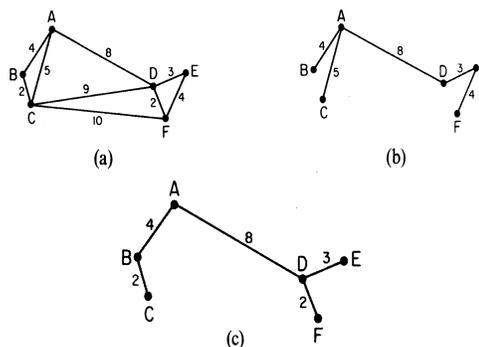
where τ is a positive thresholding function that in a sense determines the scale at which the segmentation problem is approached, and thus indirectly the size of the regions in the final segmentation. In Figure 2.2(b), the minimum internal difference is determined by comparing the internal difference values for each group of pixels. Finally, the *difference* between the two regions, shown in Figure 2.2(c), is the smallest edge weight that connects them:

$$\text{Dif}(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2} w((v_i, v_j)), \quad (2.3)$$

where v_i and v_j are vertices (or pixels) in the two different regions. When determining whether these two regions should be considered separate segments of the graph, or merged into a single region, the algorithm simply compares the values of $\text{Dif}(C_1, C_2)$ and $M_{\text{Int}}(C_1, C_2)$. If $\text{Dif}(C_1, C_2)$ is greater, the “pairwise comparison predicate” is determined to be true, and the two regions are separated. If the predicate fails, the two regions are merged.

Figure 2.1: Modified from Zahn (1971). A graph with weighted edges (a); a spanning tree of that graph (b); and the minimum spanning tree of the graph (c). [NR]

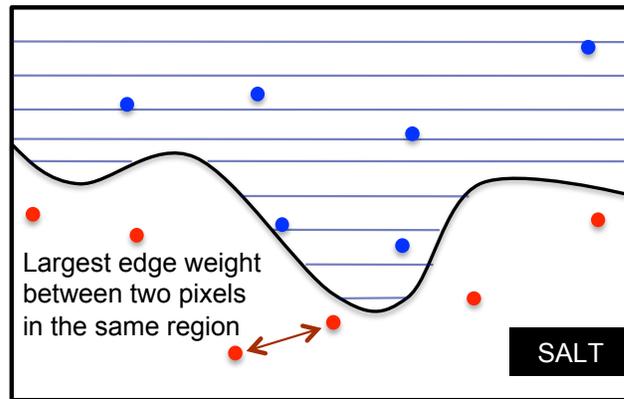
chap2/. MST



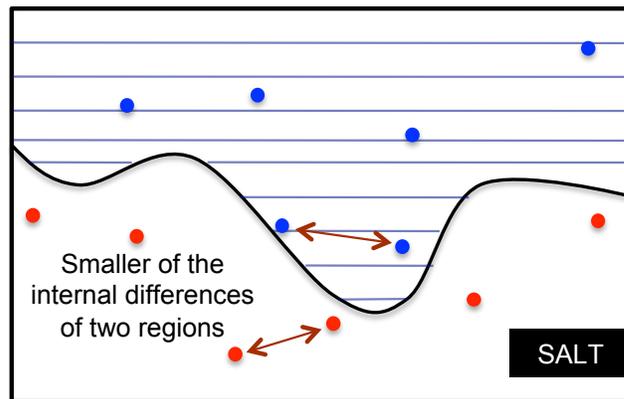
In Figure 2.2, the three definitions above are used to partition two groups of pixels separated by a putative salt boundary. While this is a relatively simple procedure, it is designed to allow highly heterogeneous regions to be segmented as a single component of an image – an important capability when handling noisy images. Additionally, Felzenszwalb and Huttenlocher (2004) note that their algorithm produces segmentations that are “neither too coarse nor too fine,” referring to the global capabilities of the segmentation process.

ADAPTATION FOR SEISMIC IMAGES

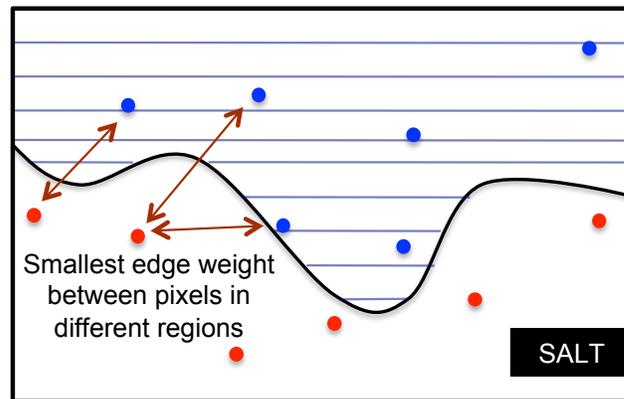
Seismic images are distinct in many ways from conventional photographs and medical images for which this and most image segmentation algorithms are designed. The consequences of these differences can be seen in Figure 2.3(b), the result of using the un-altered PRC algorithm to segment a 2D image from the Gulf of Mexico (Figure 2.3(a)). In this and all subsequent depictions of segmentation results, the interpreted segments are assigned a random color and overlaid on the image for reference.



(a)

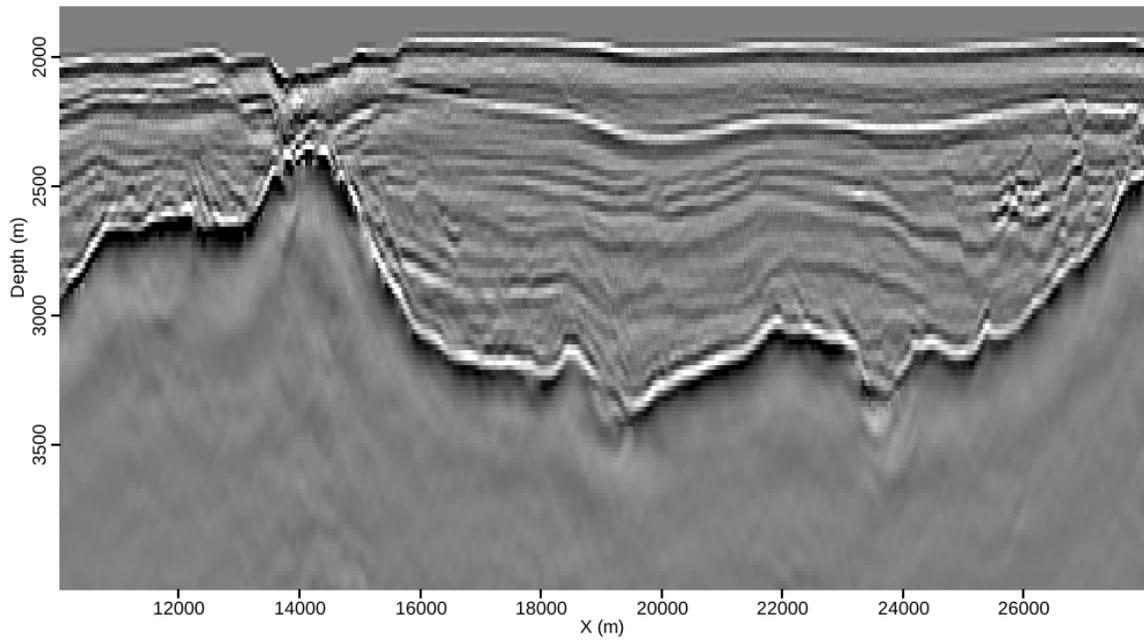


(b)

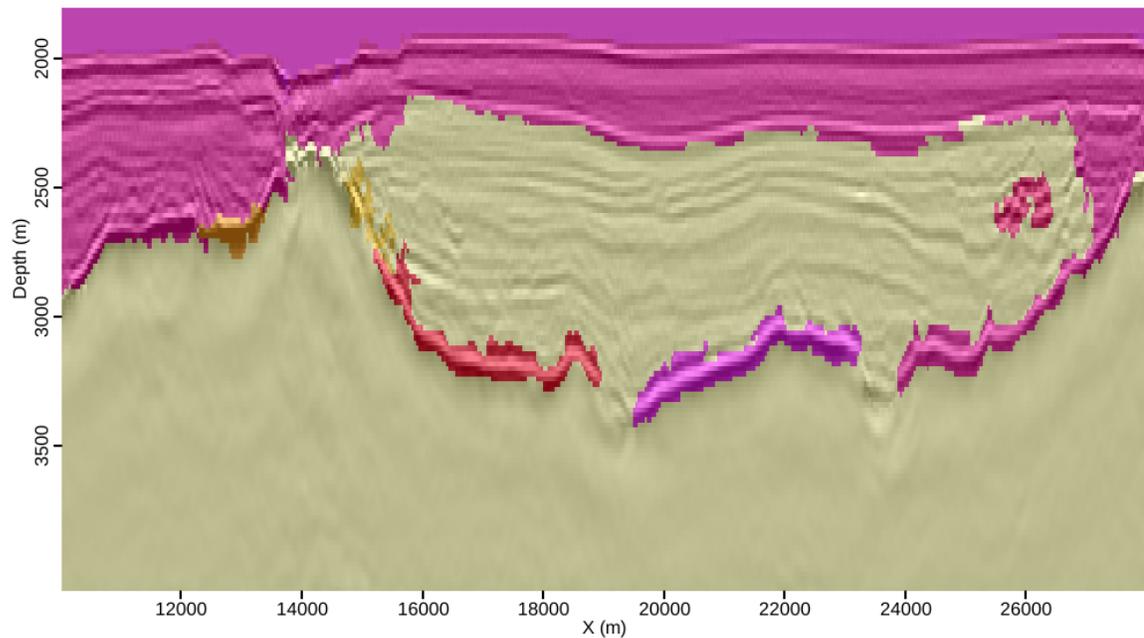


(c)

Figure 2.2: Illustrations of three crucial definitions for PRC segmentation, on two groups of pixels that may be separated by a boundary. (a) The *Internal difference* of a group of pixels is the largest edge weight between two pixels in that group; (b) the *Minimum internal difference* is the smaller of the internal differences from each group; (c) the *Difference* is the smallest edge weight between two pixels in different groups. [NR] chap2/. Int,MID,Dif



(a)



(b)

Figure 2.3: (a) A 2D field seismic image, and (b) its corresponding segmentation using the original algorithm from Felzenszwalb and Huttenlocher (2004). Prior to modification, the algorithm performs poorly because of the unique characteristics of seismic data. [NR] `chap2/. uno-img,uno-origseg2`

Transformation of input data

An initial hurdle is that seismic data may be thought of as signals with amplitude and phase varying as a function of time (or depth). This could present problems for any segmentation algorithm, and we may see an indication of this in Figure 2.3(b). At the boundary between the salt body and the surrounding rocks, the seismic waves change phase rapidly; this is common behavior when the waves encounter an interface and reflect back to the surface. As originally written, the algorithm may interpret the area around the boundary as several regions, instead of an interface between just two regions. In this case, the boundary itself becomes its own “region” in several locations. To avoid this situation, we would like the seismic image to be represented as amplitude information only, since this would indicate a single boundary between two regions. As Taner et al. (1979) point out, seismic data may be represented as a complex valued function:

$$A(z)e^{i\phi(z)}, \quad (2.4)$$

where z can be time or depth. The exponential term in this expression represents the phase information for the seismic data, while the leading term represents the amplitude information. By transforming the data such that amplitude is the only information present, the problem described above may be avoided. Figure 2.4 displays such an envelope calculation for the image in Figure 2.3(a).

Edge-preserving smoothing

In many disciplines, pre-processing images by smoothing is common prior to automatic image segmentation; however, traditional smoothing blurs boundaries and removes high spatial-wavelength features, which is counterproductive for seismic imaging and interpretation. For example, Figure 2.5(b) is the result of performing traditional box-filter smoothing on the 3D image in Figure 2.5(a). However, an edge-preserving smoothing technique based on a method called “directional maximum homogeneity” (Zahedi and Thomas, 1993) can remove unwanted incoherent noise from

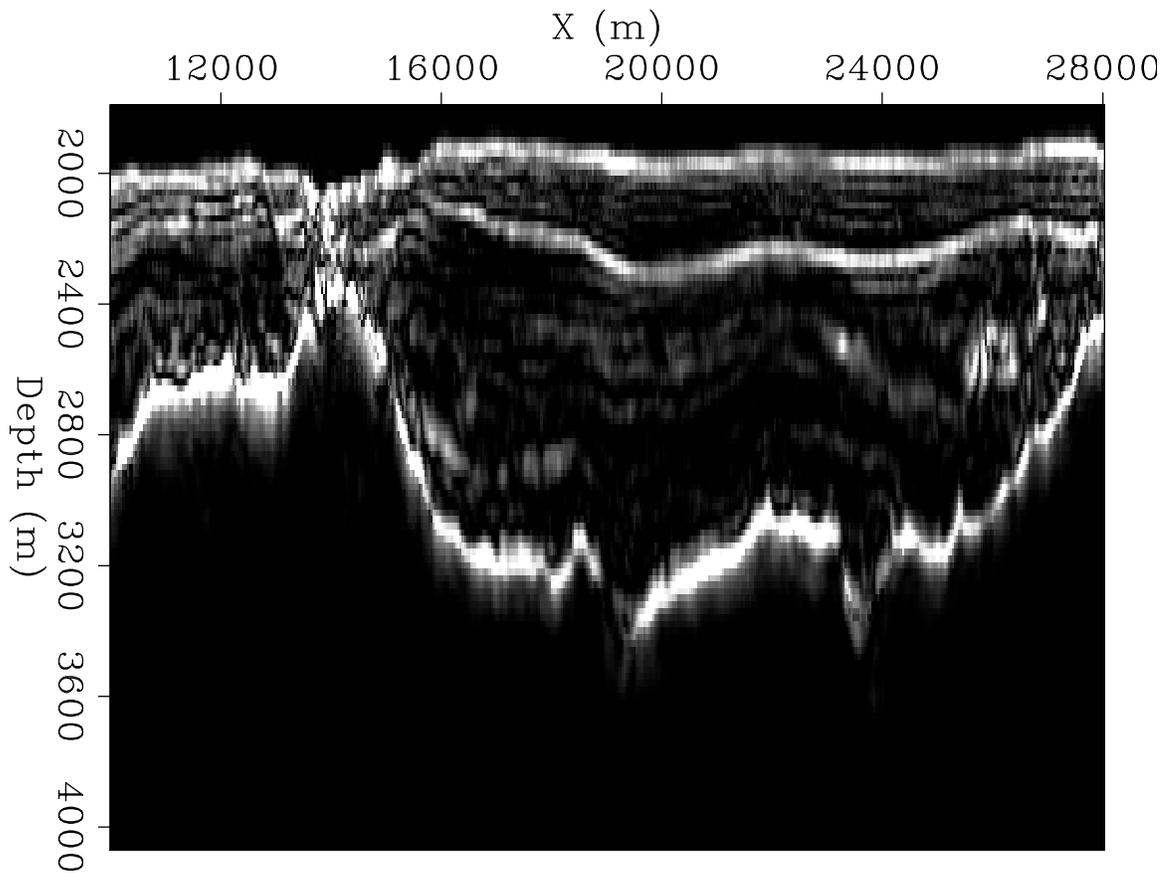
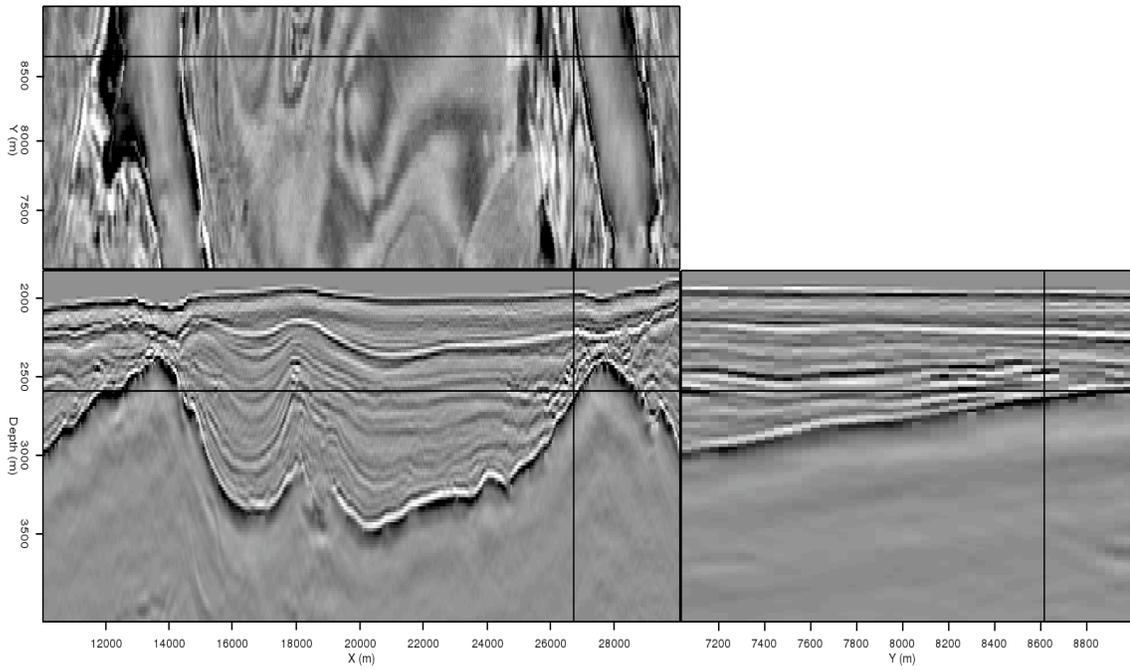


Figure 2.4: Amplitude of the envelope calculation for the raw image in Figure 2.3(a). This becomes the input for the PRC segmentation algorithm. [ER] chap2/. uno-env

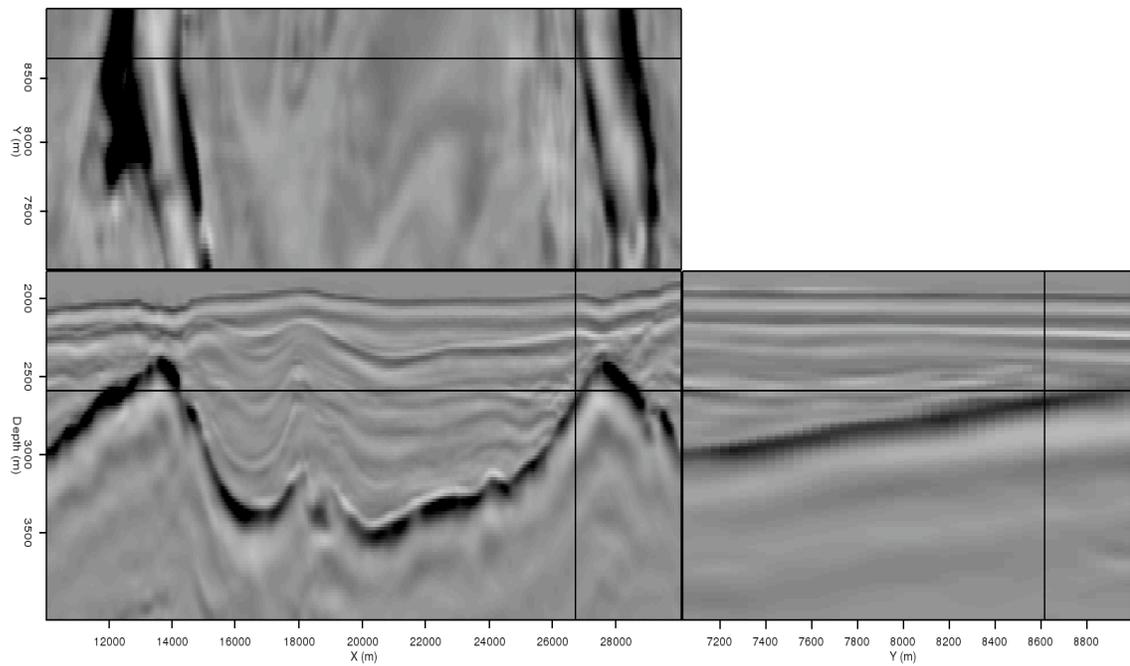
an image, while maintaining the sharp boundaries desirable for accurate segmentations. The 3D smoothing method constructs nine 1D “bar masks” extending in all directions from a central pixel, and calculates the variance of the amplitude values in each bar. By assigning the median value of the most homogeneous bar mask (smallest variance value) to the central pixel, the algorithm avoids smoothing across coherent boundaries. An indication of the superiority of edge-preserving smoothing can be seen in Figure 2.6. Differencing the image smoothed using a box filter from the original image yields a result that looks very much like the original image, indicating that a great deal of information (signal) has been removed. However, the difference result using the edge-preserving smoothing technique removes mostly “speckle” noise and very little of the signal.

Further smoothing can be accomplished by introducing a hybrid approach that combines the characteristics of box smoothing in areas without edges, and takes advantage of the edge-preserving features of maximum-homogeneity (MH) filtering when edges are present. For the MH algorithm described above, we already calculate variances for each of the bar masks passing through a given pixel. Comparing the largest and smallest of these calculated variances indicates the likelihood that an edge is present. If the ratio between the smallest and largest variances is large (close to 1), the pixel is in a relatively “isotropic” area, and an edge is unlikely to be present. Conversely, a smaller ratio implies that an edge is present in at least one of the bar mask orientations. If α is a user-determined threshold value, then if $\frac{\min(\sigma)}{\max(\sigma)} > \alpha$, traditional smoothing can safely be used in lieu of MH filtering for that particular location.

Figure 2.7 demonstrates this strategy on the 3D field data example. As the threshold value α decreases, the algorithm is more biased toward traditional smoothing; consequently, the amount of “speckle” noise decreases. Even at the smallest value of α (Figure 2.7(d)), the result is still superior to that of the traditional smoothing approach (Figure 2.5(b)) – even though the length of the bar masks and the dimensions of the box filter are equivalent. Smoothing 3D images in this manner allows the PRC segmentation algorithm to behave more robustly.



(a)



(b)

Figure 2.5: (a) A 3D image from the Gulf of Mexico; (b) The result of smoothing the image in panel (a) with a standard box-filter. [ER] chap2/. img-orig,img-smth

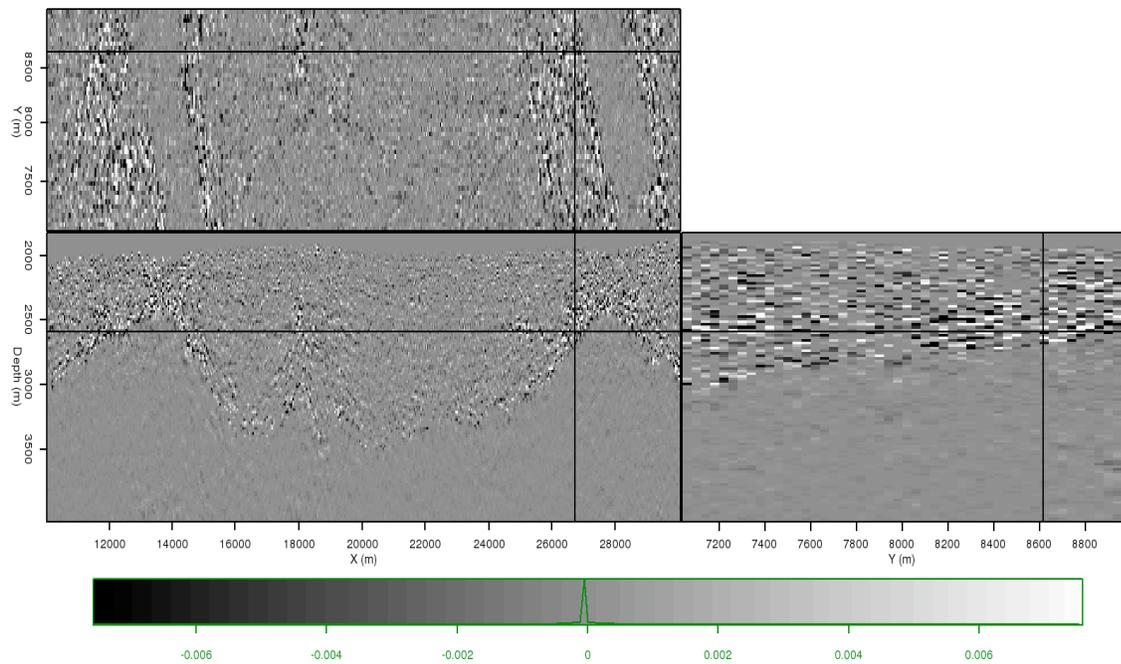
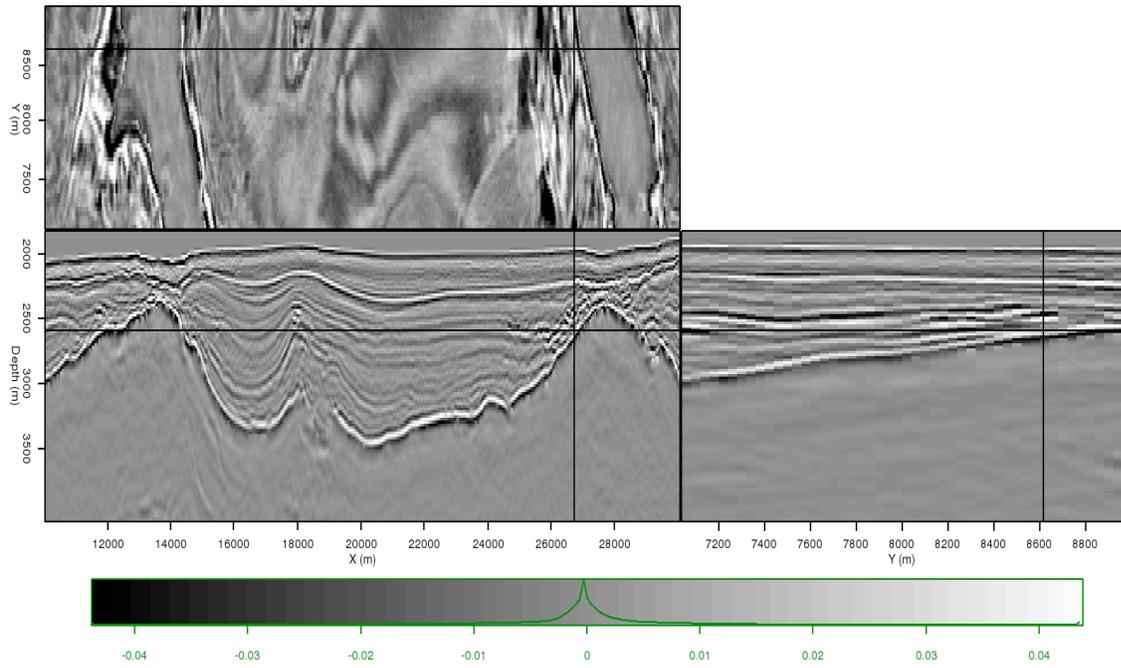
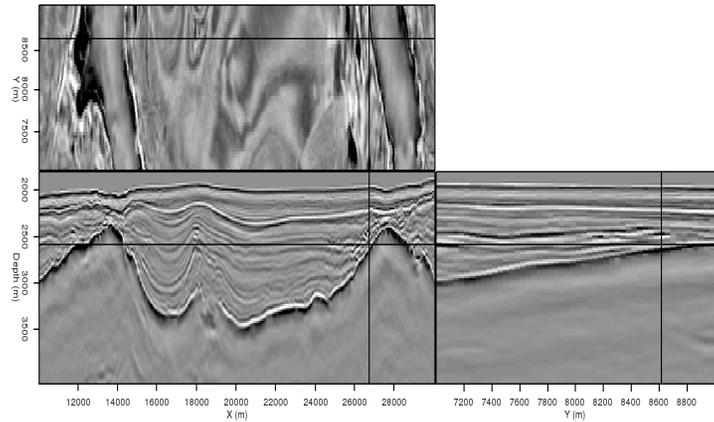
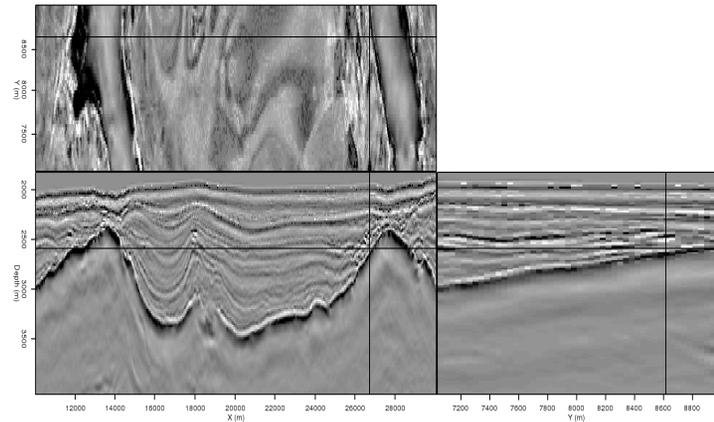


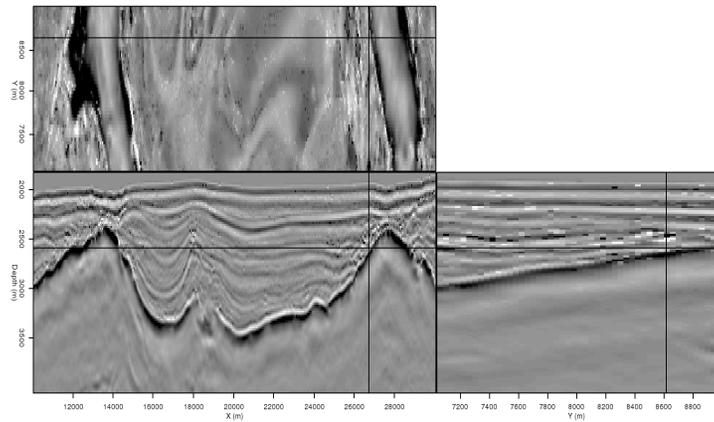
Figure 2.6: Results of differencing the original image in Figure 2.5(a) with a smoothed version obtained using (a) standard box-filtering, and (b) directional maximum homogeneity filtering. Both examples use the same size operator (box side length or bar mask length, respectively). [ER] chap2/. diff-smth,diff-mhn



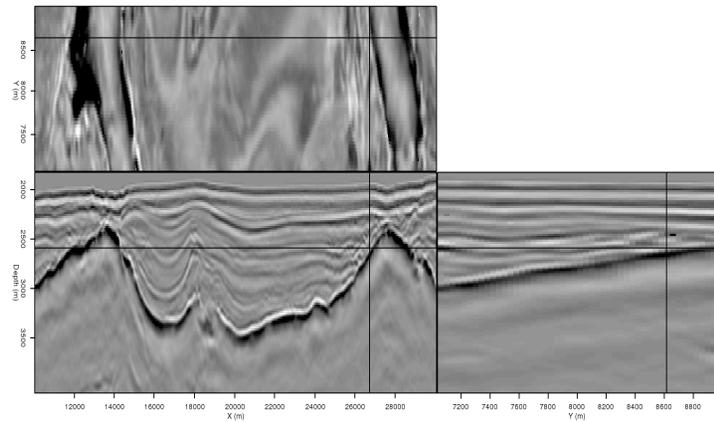
(a)



(b)



(c)



(d)

Figure 2.7: Results of applying the hybrid-MH filter on the image in Figure 2.5(a), with α set at (a) 0.5, (b) 0.2, and (c) 0.1 and (d) 0.01. While the reflector amplitudes are affected, a great deal of speckle noise is removed at low α values. [ER] chap2/. img-hyb-50,img-hyb-20,img-hyb-10,img-hyb-01

Edge construction

A distinguishing feature of seismic images is that regions such as salt bodies are most easily delineated by their boundaries, rather than, for example, color attributes used to segment photographs. Therefore, modifications to the algorithm's procedure for both constructing the graph and weighting its edges are required to obtain acceptable segmentation results for seismic images. The original implementation of the PRC algorithm creates a graph with eight edges per node (pixel). This graph is constructed by looping over every pixel, and performing four calculations at each vertex. The left side of Figure 2.8 illustrates this process – if the “active” pixel is the one in red, edges are built to each of the blue pixels. Since every pixel in the image undergoes this process, a form of reciprocity allows for each pixel to be connected to its eight immediate neighbors via edges. While this process allows for the extreme efficiency of the algorithm, the unique and often irregular nature of seismic data does not lend itself well to segmentations using so few edges per vertex or pixel. Instead, a larger stencil, such as the one on the right in Figure 2.8, has been implemented. The length of the stencil's arms is a user-defined parameter which may be adjusted based on data quality; larger stencils should be used for noisier data, but the trade-off is increased computational complexity. Increasing the size of the stencil allows for many more comparisons per pixel, and a far greater amount of information goes into the segmentation algorithm. While this approach obviously decreases the efficiency of the algorithm, the increased accuracy seen in the final results appears to make it a worthwhile trade-off. Even with the increased number of edges per node, this algorithm is still far less computationally intensive than the NCIS algorithm from Shi and Malik (2000).

Edge weighting

Finally, the edges constructed using the modified stencil in Figure 2.8 must be weighted in a manner that treats a boundary *between* two vertices as more convincing evidence for the existence of two regions than simply a difference in intensity at the

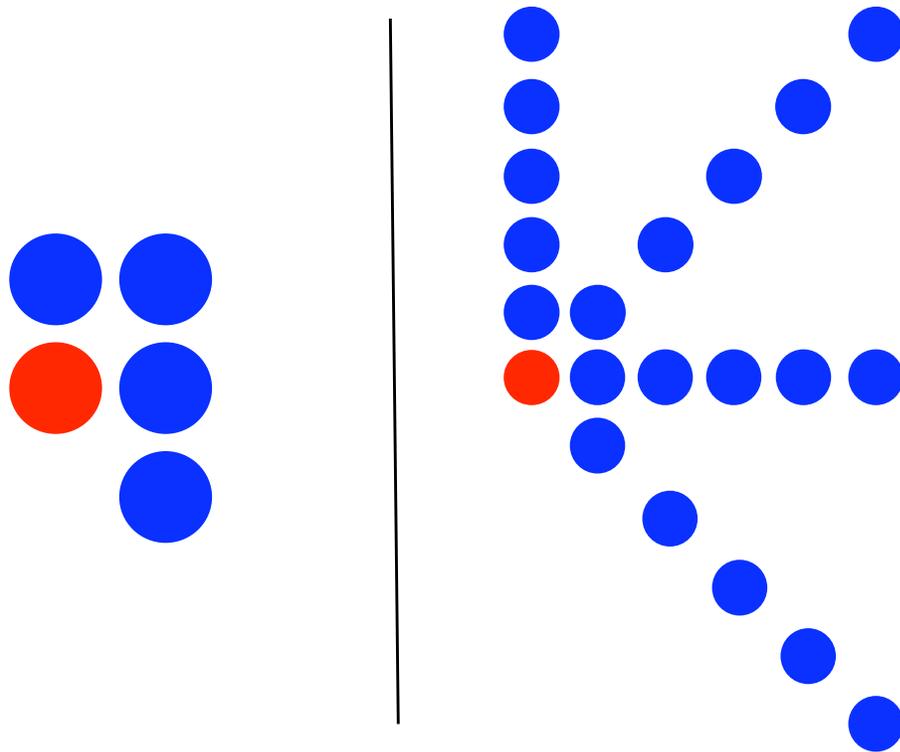


Figure 2.8: Stencils used for comparing pixel values and assigning edge weights for the graph corresponding to a 2D image. At left, the five-point stencil (8 edges per pixel) used in the original implementation from Felzenszwalb and Huttenlocher (2004); at right, a modified 21-point stencil (40 edges per pixel) used for the seismic images. For 3D images, additional stencil arms extend into the third dimension; the length of the stencil arms is a user-defined parameter. [NR] chap2/. stencils

two pixels themselves. When determining the weight for an edge with an endpoint along one arm of the stencil in Figure 2.8, I use the largest intensity value of any pixel between the two endpoints. For example, a high intensity value along one arm of the stencil would suggest that that particular arm intersects a boundary. Figure 2.9 illustrates the logic behind this process.

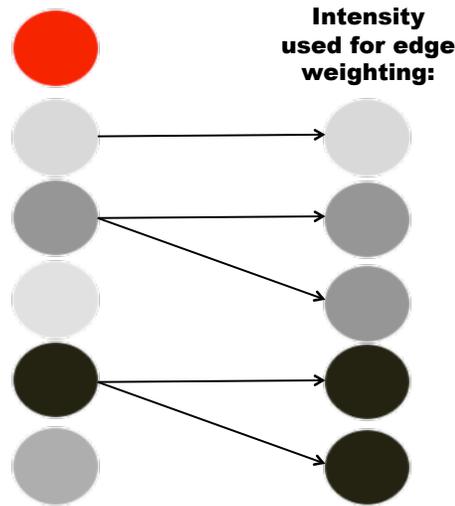


Figure 2.9: Diagram illustrating the logic behind deciding which pixel intensity value to use when calculating edge weights. Pixel intensities along one “arm” of the stencil in Figure 2.8 are shown on the left; darker colors represent higher intensities. The right column indicates which intensity value will be used when calculating the edge weight between the “active” (red) pixel and the adjacent pixel. [NR] chap2/. weights

This process obviously involves some degree of algorithmic complexity, as it requires sorting and searching the pixel intensity values along each segment. Algorithm 1 illustrates the steps for carrying out the process shown graphically in Figure 2.9. After creating the edges linking each pixel in a line segment to the “active” pixel, sort the line segment’s pixels in decreasing order of pixel intensity. Once this is done, compare the index value of the edge vertex pixel with the intensity-ranked list of pixel indices. To find the highest-intensity pixel value between the two vertices, simply take the value of the first pixel index on the sorted list that is less than the index of the vertex pixel.

Algorithm 1 Calculating graph edge weights

```

for each pixel ipix in image do
  create four line segments with five pixels per segment;
  record relative position (path.ind) and intensity (path.val) of each pixel;
  for each line segment do
    sort the segment in decreasing order of pixel intensity;
    for each pixel ix in the segment (nearest to furthest from pixel ipix) do
      for ij = 1..5 do
        if path[ij].ind <= ix then
          calculate edge weight using path[ij].val;
        end if
      end for
    end for
  end for
end for

```

Once we have selected the intensity value to use for determining the edge weight, the weight value is calculated using an exponential function:

$$w_{ij} = \exp((\max I(\mathbf{p}_{ij}))^2) \exp(d_{ij}), \quad (2.5)$$

where \mathbf{p}_{ij} is the vector of all pixels between i and j and d_{ij} is simply the Euclidean distance (in samples) between the two pixels. Recall that the $\max I(\mathbf{p}_{ij})$ term is determined according to algorithm 1. The distance-weighting d term accounts for the fact that the edges in the graph can now be much longer than with the adjacent-pixels-only approach taken in the original implementation.

Once each of the edges is assigned a weight, the segmentation of the image can proceed as described in Felzenszwalb and Huttenlocher (2004). Algorithm 2 summarizes this process, which begins with each pixel as its own image segment. Then, individual pixels, and eventually, groups of pixels, are merged according to thresholding criteria. Segments can also be merged in post-processing if they are smaller than a “minimum segment size” parameter specified by the user. Figure 2.10 is the much-improved result when the example image in Figure 2.3(a) is segmented by the modified PRC algorithm.

Algorithm 2 PRC segmentation procedure

```

for each pixel ipix in image do
  build edges to neighboring pixels;
  weight the edges according to the procedure in algorithm 1;
end for
sort the edges in increasing order;
create initial segmentation  $S_0$ 
for each edge iedge in the sorted list do
  calculate Dif and MInt values for pixels/regions connected by iedge
  if  $Dif > MInt$  then
    partition the image at iedge
  else
    merge the two pixels/regions
  end if
end for

```

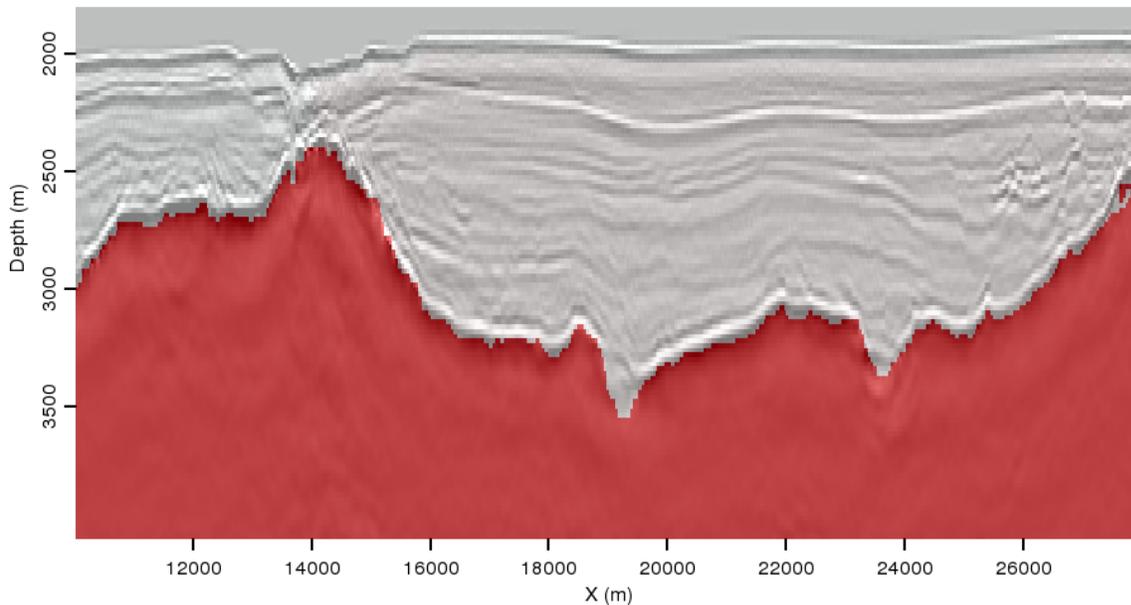


Figure 2.10: Final segmentation result of the image in Figure 2.3(a), after modifications to the algorithm. [ER] chap2/. uno-segmerge

PARAMETERIZATION AND UNCERTAINTY

A key advantage of the PRC segmentation scheme is ease of parameterization. In practice, only two user-controlled parameters must be set prior to segmentation: the length of the modified stencil arms described in the previous section, and the minimum size (in pixels) of the automatically interpreted image segments. The latter parameter is straightforward for salt segmentation, especially if a prior model provides an approximate size for the salt body. By setting this parameter to a value of the same order but smaller than the expected size of the salt body, an interpreter can maximize the likelihood of an accurate segmentation. It is advisable to err on the side of smaller minimum segment sizes; doing so may lead to a salt body being broken up into multiple segments, but this is easily remedied and creates an opportunity for interpreter guidance in difficult areas, as described in the next section.

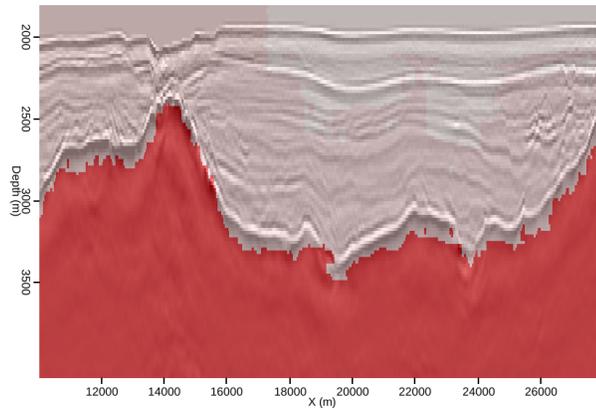
The effects of changing the stencil length parameter are more subtle, but may provide an opportunity for uncertainty analysis. The lack of a direct measure of degree of certainty or uncertainty for automatic segmentations is the primary drawback of the PRC method, compared, for example, to the NCIS method, in which the sharpness of the transition from negative to positive eigenvector values across a boundary acts as a proxy for certainty of the boundary location. While the PRC method does not provide a similar measure, the extreme efficiency of the algorithm opens the possibility of performing multiple segmentations of the same image, but with different parameters. By observing whether and where a salt boundary changes as a result of modifying parameters, we can hope to gain an understanding of where the interpreted boundary exhibits the most or least uncertainty. Because the stencil length parameter in effect controls the amount of information used to construct and weight graph edges for a given location, it is an ideal candidate for such a scheme.

Figure 2.11 shows segmentation results for the image in Figure 2.3(a), using three different values for the stencil arm length parameter. Physically, the length of the stencil arms can be related to the wavelength of the salt boundary reflector. In this case, the width or wavelength of the salt boundary throughout the image is

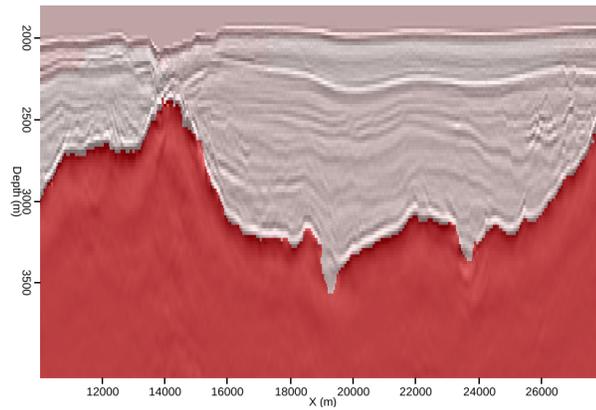
approximately five pixels, and employing a stencil of this length appears to produce the most accurate result (panel b). A larger stencil (panel a), in which the arms extend for approximately twice the wavelength of the boundary reflector, provides the least accurate result. This makes intuitive sense, as even pixels far from the boundary may be affected by the boundary's "halo". However, a smaller (three-pixel) stencil with arms extending approximately half of the reflector's wavelength (panel c) performs almost as well as the ideal five-pixel stencil, with the exception of a clear error near $x = 23000m$. At this location, the additional information taken in by the longer stencil, at increased computational cost, is a worthwhile tradeoff. Interestingly, the boundary at this location is very faint and even discontinuous compared to the rest of the image; this can be seen not only on the underlying image in Figure 2.11, but especially on the corresponding amplitude of the envelope image in Figure 2.4. In this example, varying the stencil length parameter has quickly and clearly identified a zone of high uncertainty along the boundary.

INTERPRETER GUIDANCE

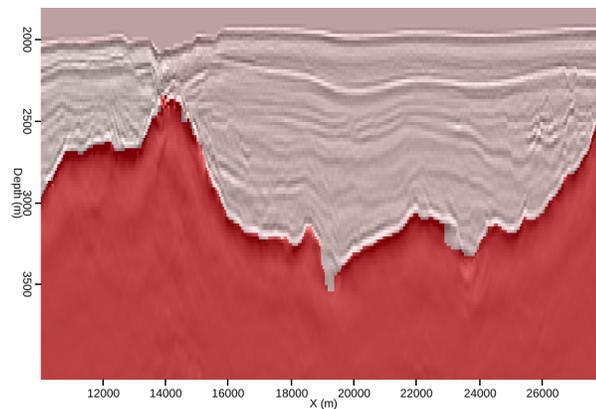
Unfortunately, a fully-automatic method will often be insufficient for obtaining an acceptable salt interpretation. In such cases, results of the automatic process can be evaluated and improved via input from a human interpreter. Interpreter input is desirable for a variety of reasons. First, in many circumstances, interpreters can spend years learning the geological features of a given basin or area of the world; this knowledge is extremely valuable for model-building purposes, and near impossible to "program" into a fully-automated procedure. A second reason human input is valuable stems from the pattern recognition advantages of humans relative to computers, especially in two dimensions. Even novice interpreters would likely have little trouble immediately recognizing the salt body on the 2D section in Figure 2.12, even though, as we will see shortly, automatic interpretation algorithms may struggle. However, in three (or more) dimensions, taking advantage of computational resources could allow interpreters to overcome visualization challenges while streamlining the interpretation process. Ideally, a semi-automatic interpretation workflow would allow interpreters



(a)



(b)



(c)

Figure 2.11: Automatic segmentation results for the image in Figure 2.3(a), using three different lengths for the stencil arms depicted in Figure 2.8: (a) 10 pixels; (b) 5 pixels; (c) 3 pixels. [ER] chap2/. stencil10,stencil5,stencil3

to provide limited manual interpretations on 2D sections, and use that information to guide the automatic 3D seismic image segmentation. This is the goal of the method presented here.

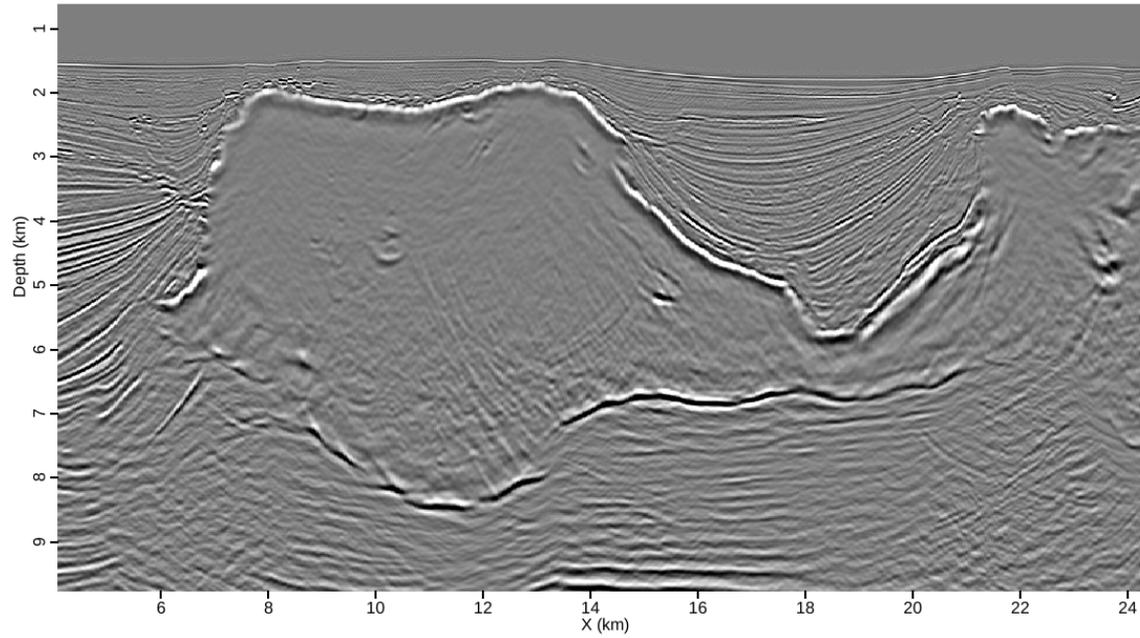


Figure 2.12: A 2D section from a Gulf of Mexico dataset. The prominent salt body provides an ideal test case for segmentation algorithms. [ER] chap2/. oct-2d

An initial opportunity to incorporate interpreter guidance is apparent from the segmentation result in Figure 2.13(a), obtained using another 2D example from the Gulf of Mexico (Figure 2.12). In this result, the algorithm has identified several distinct segments within the salt body. It would be much easier to evaluate the segmentation results if the entire salt structure were instead a single segment. This is easily achieved if an interpreter quickly identifies which segments belong inside the salt; a single click can accomplish this task, as shown by the red dots in Figure 2.13(a). Merging these segments provides a much more interpretable result, seen in Figure 2.13(b). An additional benefit of this capability involves the model-building aspect of segmentation. Assigning a user-defined value to the new, merged segment allows for a simple means of setting velocities for specific regions, such as a salt body. Furthermore, in some cases there may be a question of whether or not a certain segment actually belongs in a salt body; by choosing to include or exclude the segment, an interpreter can quickly generate multiple possible models. This is ideal for scenario testing, an issue I will revisit in Chapter 4.

Acquisition, model-building, and imaging challenges all contribute to situations in which salt boundaries appear faint, discontinuous, or not to be present at all. In Figure 2.12, for example, there are locations along both the top and base of the salt body where the boundary is poorly imaged. This can result in “leakage” of the automatically interpreted salt segments seen in Figure 2.13(b). In these cases, valuable interpreter insight should be incorporated into the procedure. Figure 2.14 shows manual salt boundary interpretations in areas where leakage is apparent in Figure 2.13(b). The most efficient way to include this information in the PRC algorithm is to modify the input image by increasing intensity values at the manual pick locations. Instead of assigning arbitrarily large values, however, we define a new amplitude value (A) for a manually-picked pixel at position (x,y,z) in terms of the highest-amplitude pixel in a neighborhood surrounding it and a scaling factor α :

$$A_{xyz} = \alpha \max_{|x-i|\leq 5, |y-j|\leq 5, |z-k|\leq 5} A_{ijk}. \quad (2.6)$$

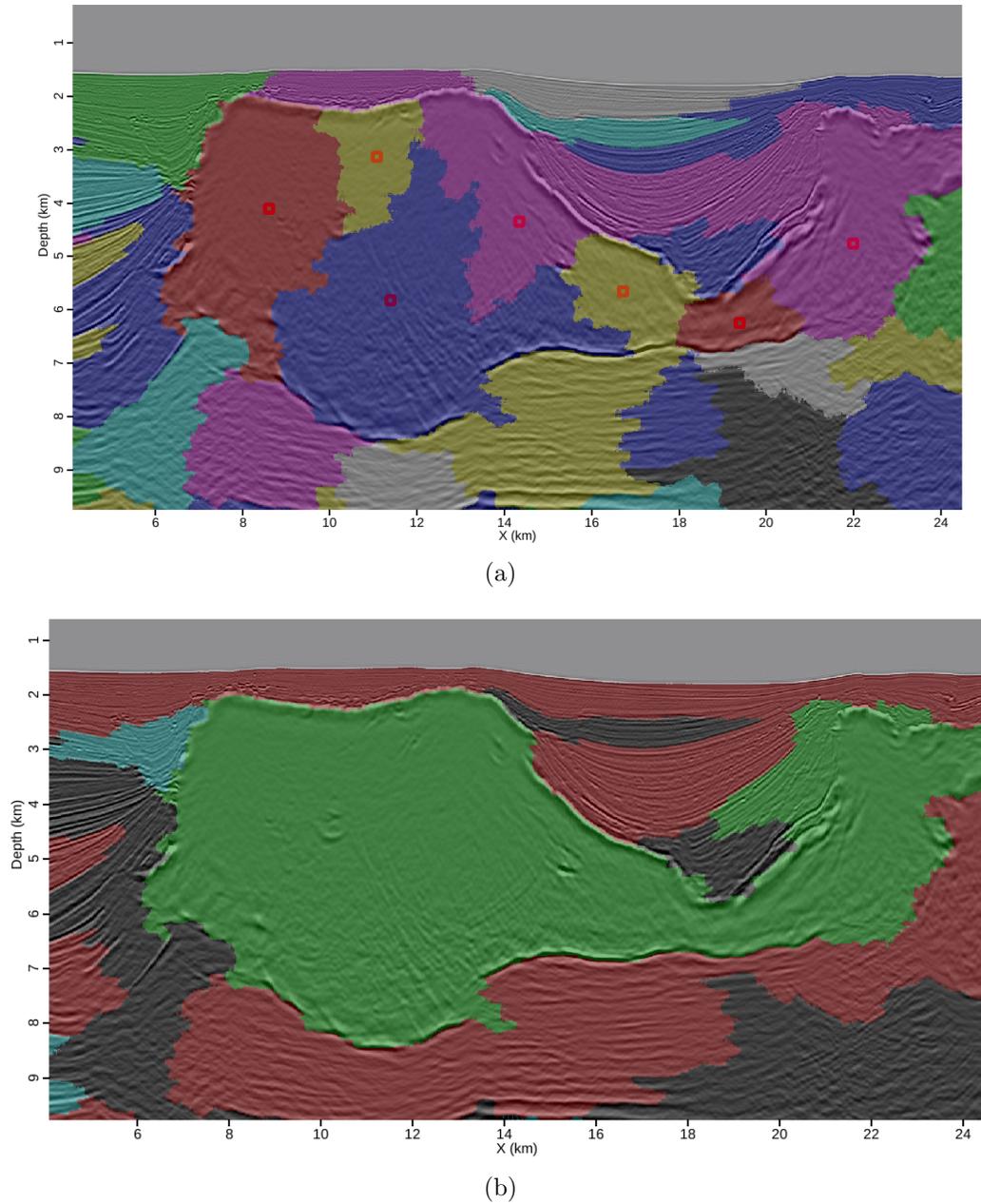


Figure 2.13: (a) A segmentation of the image in Figure 2.12 according to the algorithm presented in section 2. The salt body is divided into several segments, so an interpreter can select which segments (denoted by the red dot) belong with the salt body. (b) Result after merging the segments marked in panel (a). It is now much easier to properly evaluate the segmentation results. [ER] chap2/. o2d-picks,2d-orig

This ensures that the picked boundary will not appear radically different from its surroundings, which could present challenges for the automatic segmentation algorithm. Now, segmenting the new input image with parameters identical to the original segmentation yields the result seen in Figure 2.15. The segments conform to the manual picks seen in Figure 2.13(b), while the rest of the image is segmented as accurately as the original result in Figure 2.13(b).

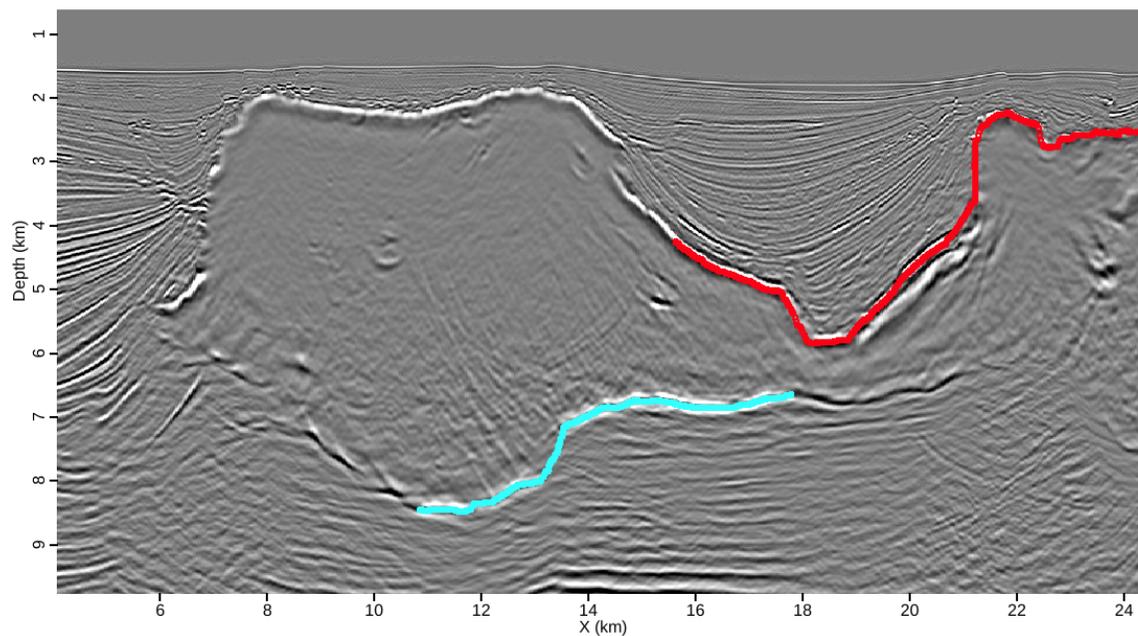


Figure 2.14: Manual salt picks supplied to guide the automatic segmentation. These picks are designed to correct the leakages seen on the segmentation result in Figure 2.13(b). [ER] chap2/. 2dtopbase

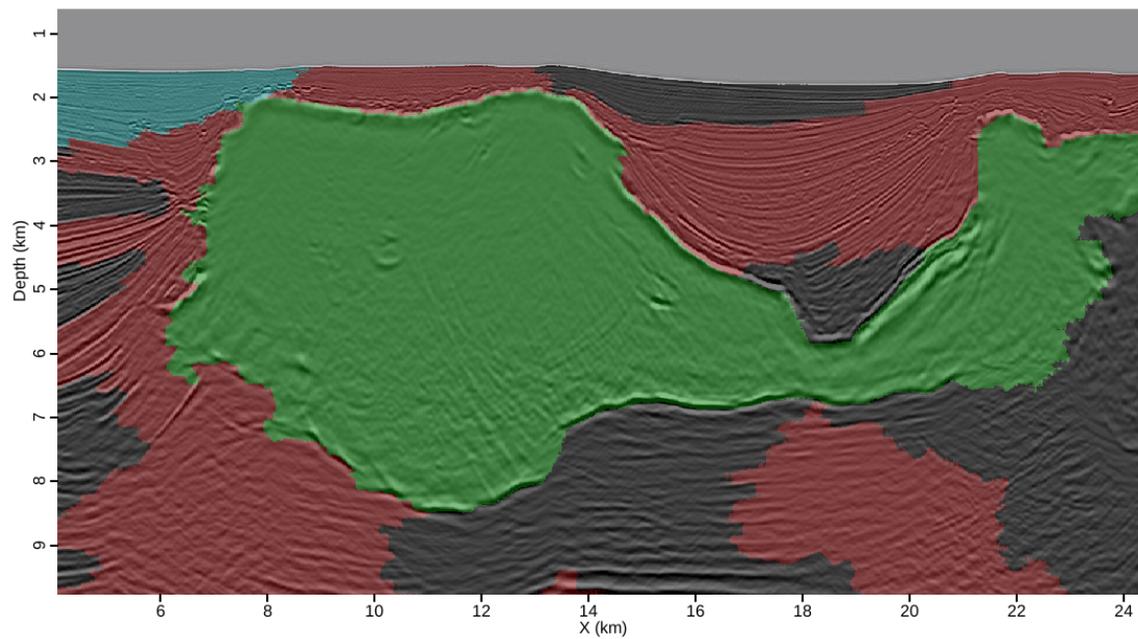


Figure 2.15: Segmentation of Figure 2.12 after incorporation of interpreter guidance. [ER] chap2/. seg-final

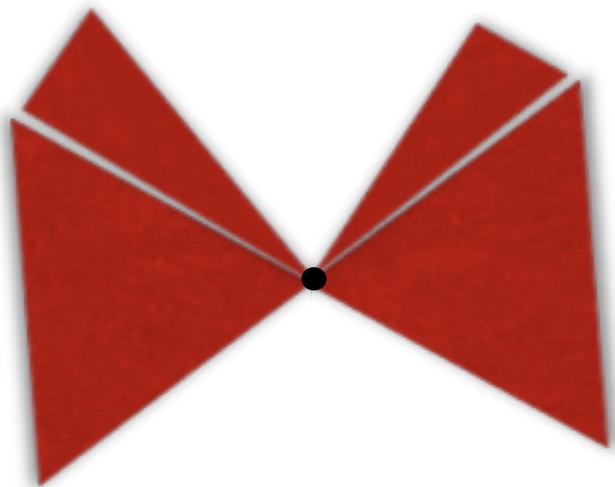
Additional modifications are required for 3D images. Because segments are much larger in 3D, amplitude changes on a single 2D section are not significant enough to alter 3D segmentation results. Instead, we must “project” an interpreter’s manual picks on an inline section, into the third (crossline) dimension. To do this, we make the assumption that a salt boundary will not fluctuate by more than two pixels per slice in the crossline direction, and construct a square pyramid in the crossline direction like the one depicted in Figure 2.16. The pyramid has sides of length $2h$, where h is the number of crossline samples between the base of the pyramid and its apex, which is the manually interpreted point. Now, for any pixel Q that falls within a pyramid with an apex at point P , the new amplitude value at point Q is

$$A_{\text{new}}^Q = A_{\text{orig}}^Q + \frac{A_0}{\|PQ\|}, \quad (2.7)$$

where A_0 is the amplitude value at point P as determined by equation 2.6, and $\|PQ\|$ is the distance between the two points. The expression is additive to ensure that any hint of the boundary already present will not be overwhelmed by the interpretation on a nearby slice. In addition, the influence of the manually-picked points decays with distance from those locations, to allow for natural variations in geology.

Figure 2.16: Depiction of the pyramid used to “project” an interpreter’s picks from a single 2D slice into the third dimension. The influence of the pick (at the apex of the pyramid) decays with distance. [NR]

chap2/. pyramid

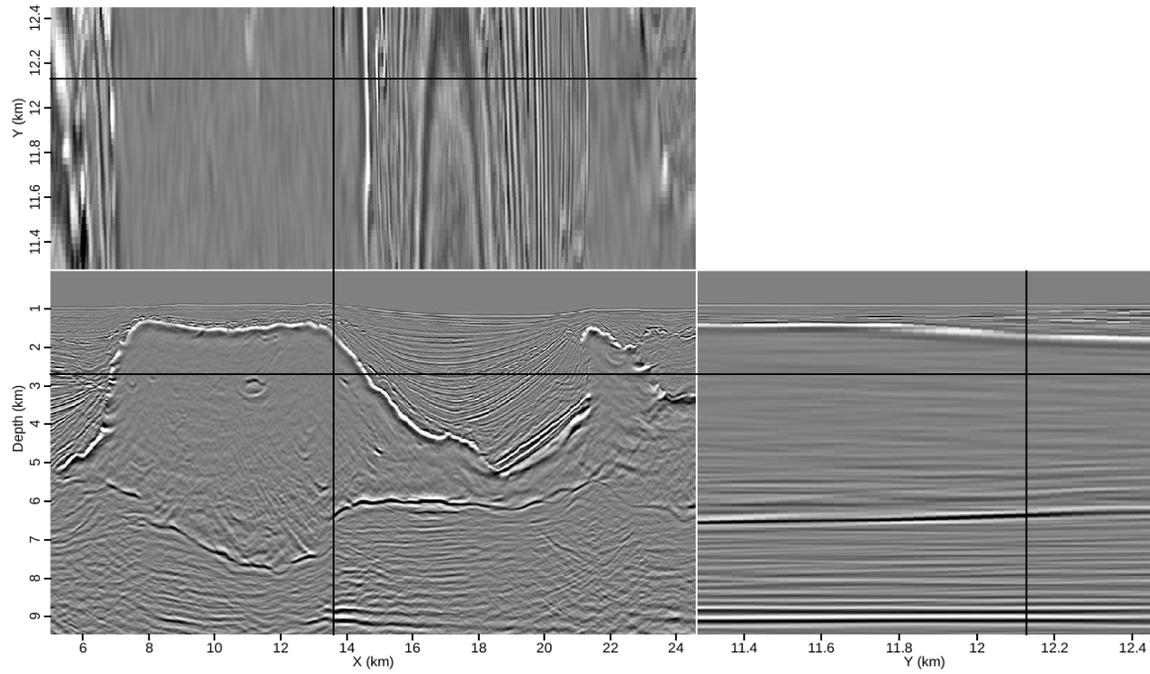


3D field data example

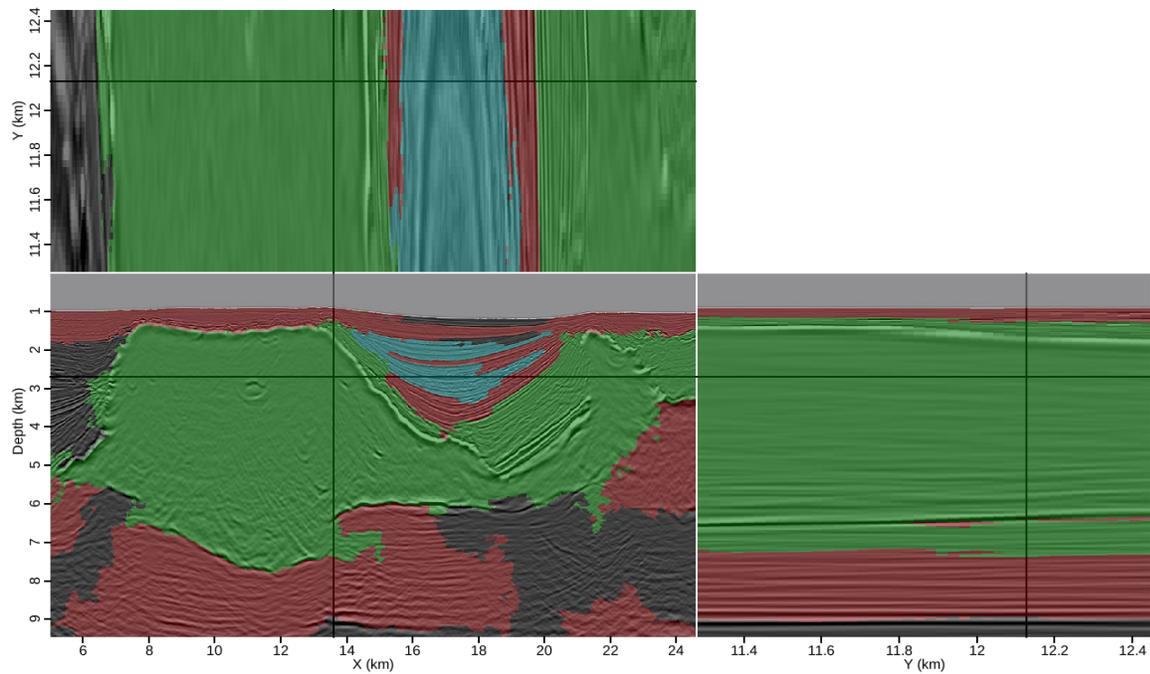
Figure 2.17(a) shows slices through a 3D image cube from a Gulf of Mexico dataset. From the initial segmentation result (Figure 2.17(b)), it is clear that the salt boundary discontinuities present challenges for the automatic segmentation algorithm. To correct the apparent leakages, manual interpretations are supplied for selected locations at two crossline values (Figure 2.18). The effects of these manual picks on the input amplitude data are shown in Figures 2.19(a) and 2.19(b). Not only are higher amplitudes obvious at the pick locations themselves, but the procedure described in the previous section has clearly influenced the intensity values at neighboring crossline values. Now, the updated segmentation result (Figure 2.20) is improved on both the inline and crossline sections. To emphasize the improvement, we should examine a location farther away from any manual picks. Figure 2.22(a) is the initial segmentation result for such a location, and leakages similar to those in Figure 2.17(b) are apparent. The effects of the 3D interpreter guidance scheme can be seen in Figure 2.21 – amplitudes have been boosted along the salt boundary in panel (b), even though the manual picks were supplied farther away. The improved segmentation result in Figure 2.22(b) demonstrates that even far from the actual picks, the automatic segmentation process is significantly more accurate when incorporating interpreter guidance. It is important to note, however, that the new segmentations are still not perfect; in particular, a small but noticeable inclusion within the salt body (near $X = 11km, Y = 11.5km$) is undetected. This is an artifact of the user-controlled minimum segment size parameter. In a model-building scenario like the one explored in Chapter 4, inclusions like this one can be isolated and accurately segmented.

COMPARISON WITH PREVIOUS METHOD

As expected, the algorithm is extremely efficient. This 3D example had over 30 million pixels, and over 700 million graph edges were constructed. In terms of cost functions, this translates to a computational requirement on the order of 5.2×10^8 operations for the PRC algorithm, compared to 9×10^{14} operations for the Normalized Cuts



(a)



(b)

Figure 2.17: (a) Slices through a 3D image cube from the Gulf of Mexico; (b) Segmentation result prior to interpreter guidance. [ER] chap2/. oct-3d,3d-origseg

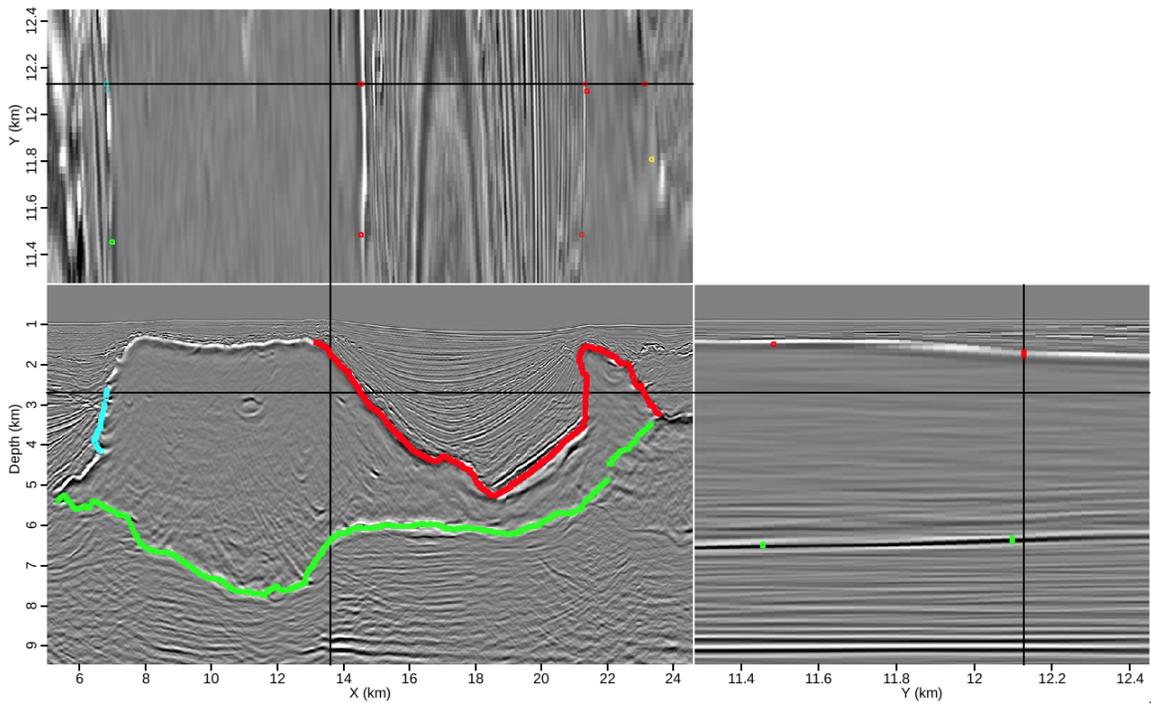
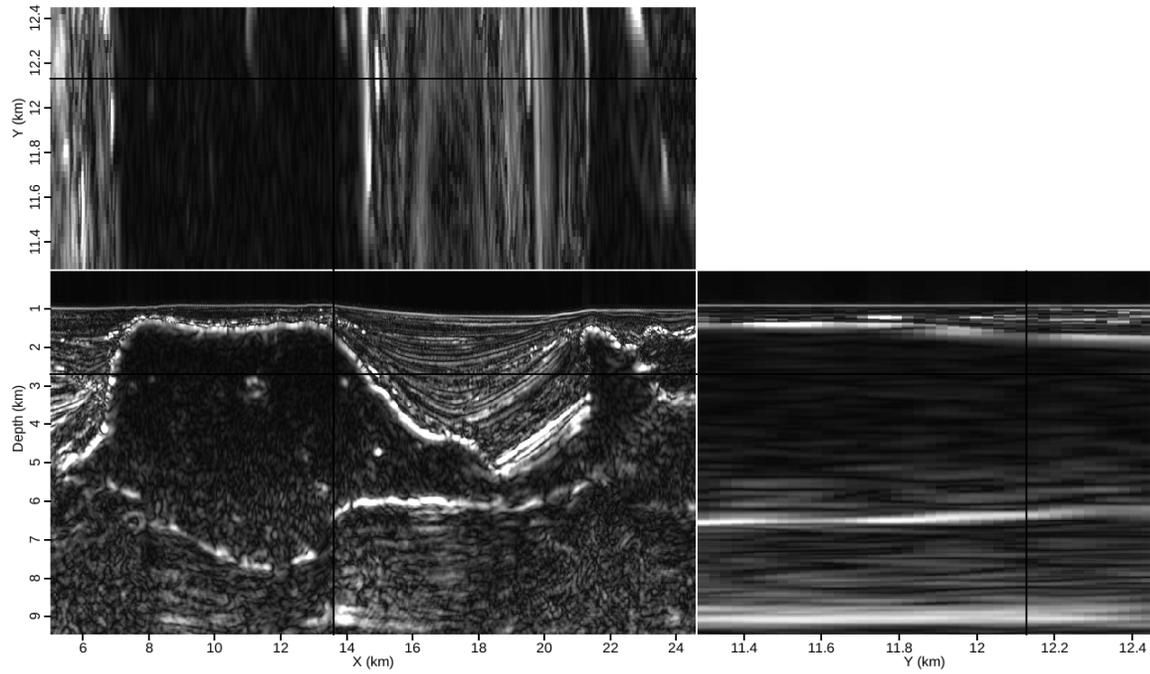
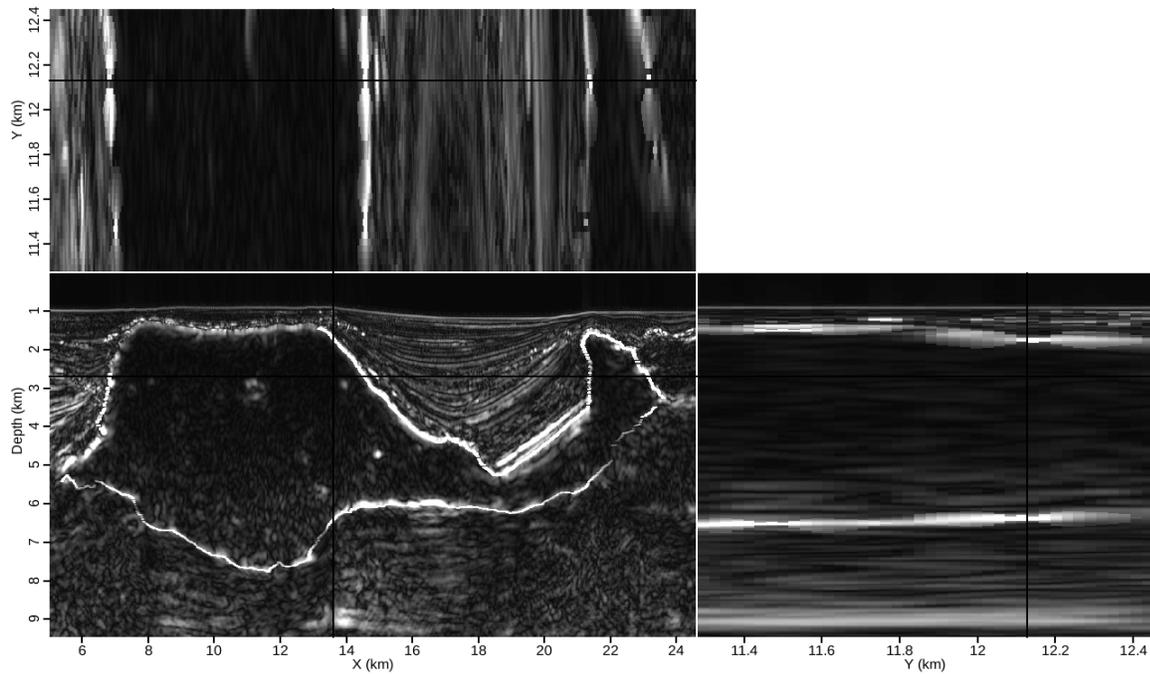


Figure 2.18: Manually-interpreted salt picks used to guide the automatic 3D segmentation. [ER] chap2/. oct-3d-picks



(a)



(b)

Figure 2.19: Amplitude of the envelope volumes (a) before and (b) after modification according to the interpreter guidance scheme. [ER] chap2/. 3d-env-orig,3d-env-new

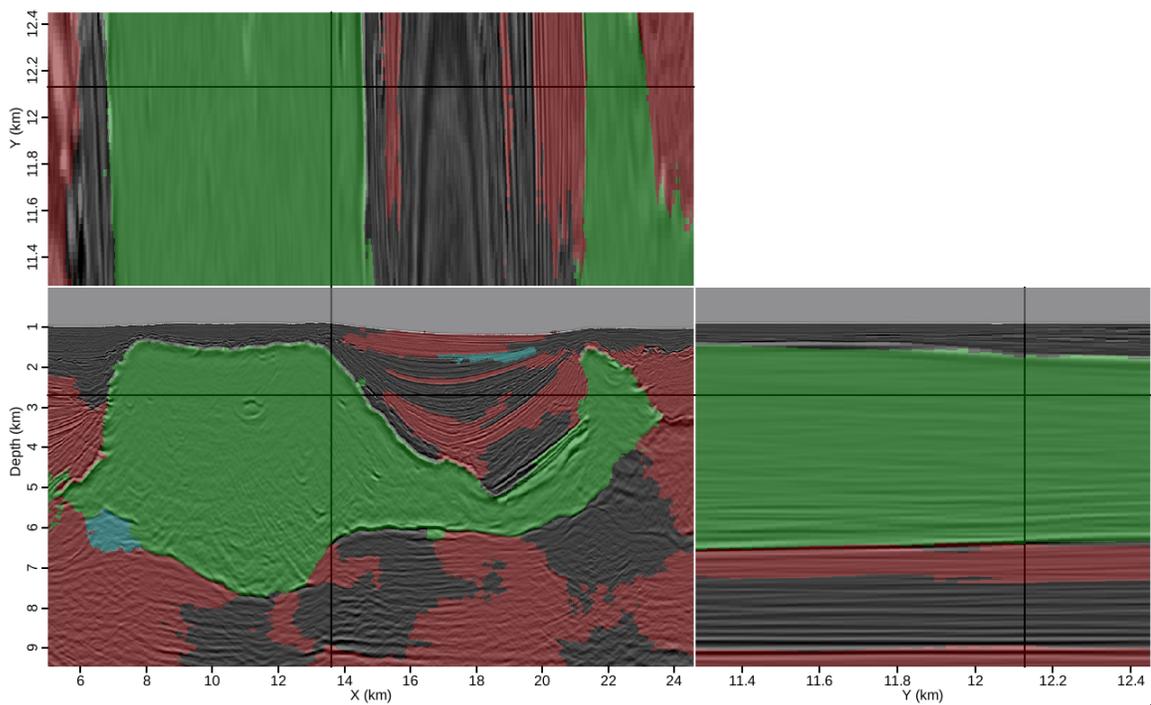
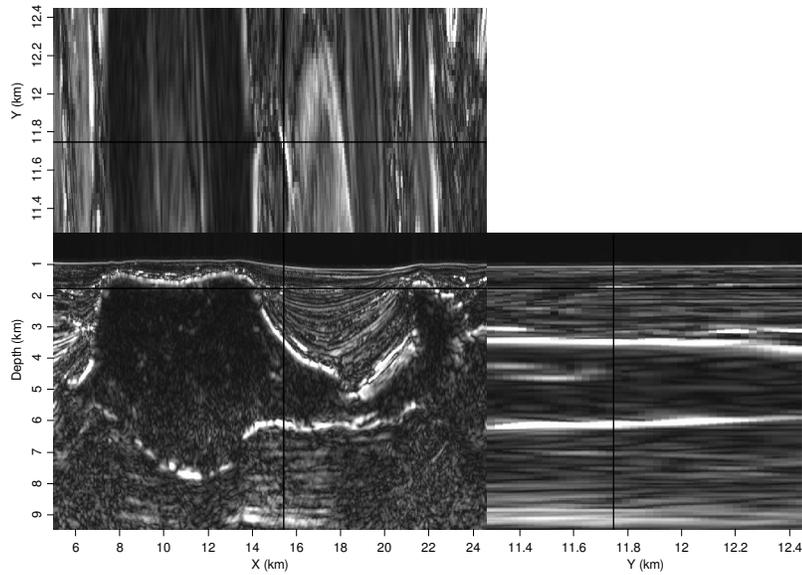
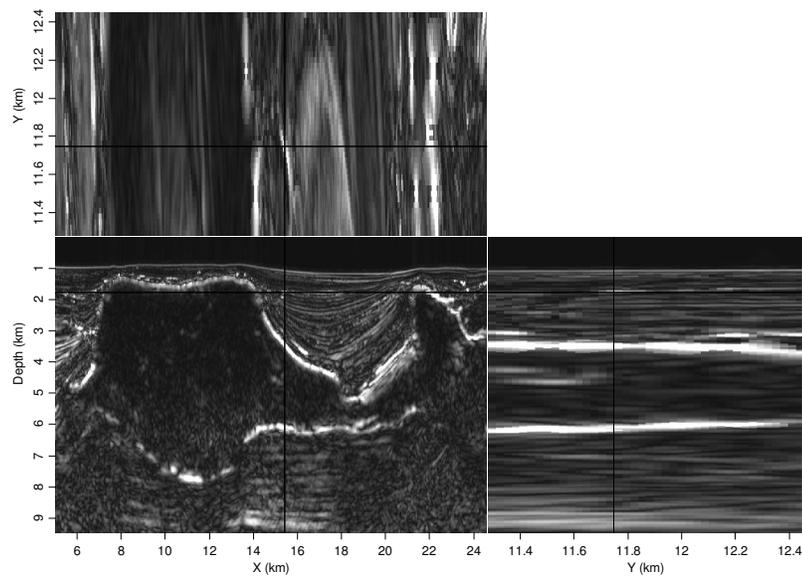


Figure 2.20: Segmentation result incorporating interpreter guidance. While not perfect, this result is much more accurate than the initial result in Figure 2.17(b). [ER]
chap2/. 3d-newseg

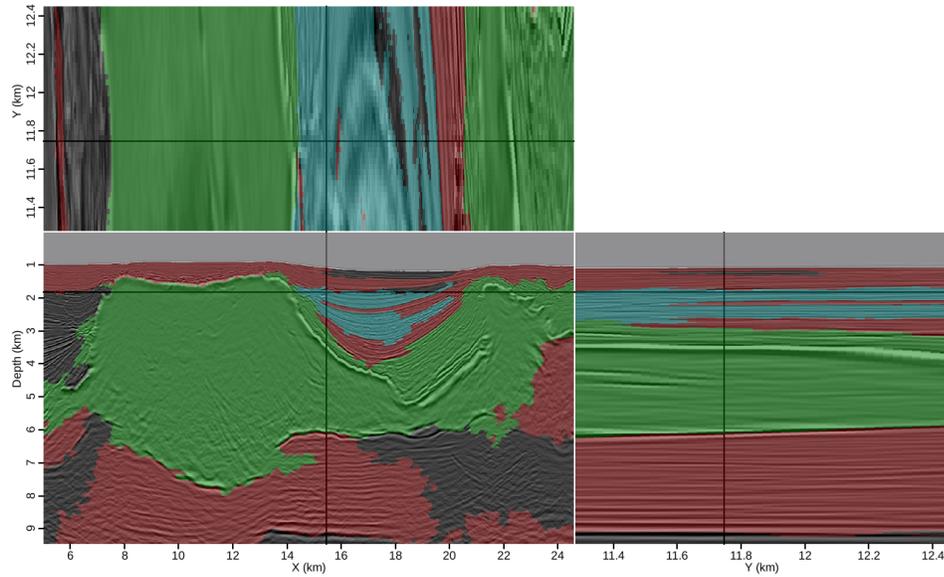


(a)

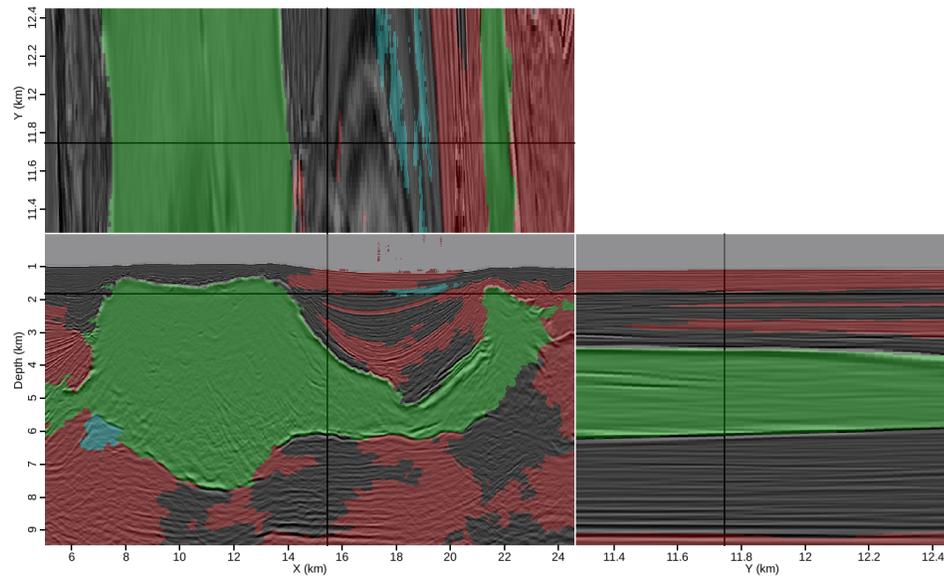


(b)

Figure 2.21: Amplitude volumes at a location far from any manual picks both (a) before and (b) after modification via 3D interpreter guidance. The effects of the interpreter guidance scheme can be seen along the salt body boundaries, even though the interpreter's picks are farther away. [ER] chap2/. o3d-env-origfar,o3d-env-new-far



(a)



(b)

Figure 2.22: Segmentation results for another set of slices through the 3D cube, (a) before and (b) after interpreter guidance. The result is greatly improved, even far from where the manual salt picks were supplied. [ER]

chap2/. 3d-origseg-far,3d-newseg-far

(NCIS) method. A single CPU performed the PRC segmentation in less than three minutes, while a full 3D segmentation of an image this size would not be feasible using the NCIS method, due to the complexity of the necessary eigenvector calculations. A direct comparison can be made, however, using a smaller 2D image such as the example in Figure 2.3(a). Segmenting this image using the PRC method as described here required less than one second, while the NCIS algorithm required 31 seconds using the same computational resources. Due to the differences in the cost functions of the NCIS and PRC algorithms, the computational advantage would be even greater for the PRC algorithm for larger and more realistic images.

Of course, computational efficiency means little if the resulting segmentation is not accurate. Again, a direct comparison of the relative accuracy of the NCIS and PRC methods can be obtained via analysis of the field data results. Figure 2.23 shows both calculated salt boundaries overlain on the image: the NCIS boundary is green/light-colored, and the PRC boundary is red/dark. Visually, we can see very little difference between these two results; in many locations, they are almost exactly on top of one another. The most noticeable difference between the two results is near $X = 20000m$, where the PRC boundary dips deeper than the NCIS boundary. Examination of the input image in Figure 2.3(a) suggests that in this location, an error in the migration velocity model has led to a discontinuity in the boundary image. The new method appears to do a better job of correcting the error; this may be a result of the denser network of graph edges created using the PRC method compared to the particular NCIS implementation shown here. This result serves to increase confidence in the viability of the PRC segmentation scheme as a useful interpretation tool.

CONCLUSIONS

Because of its global segmentation capabilities and extreme computational efficiency, the Pairwise Region Comparison (PRC) algorithm is an attractive candidate for seismic image segmentation. After modifications to the algorithm to account for the

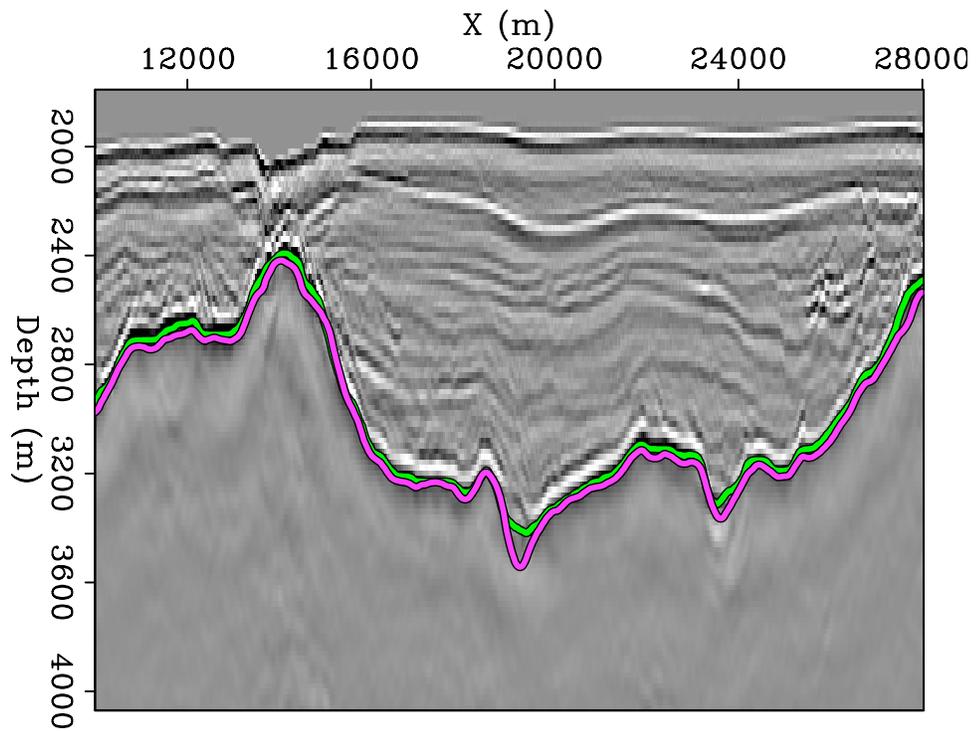


Figure 2.23: Salt boundary identified on a 2D image using two automatic segmentation techniques: NCIS (green/light color) and PRC (red/dark). [NR]

chap2/. uno-segcomp

unique nature of seismic images, the PRC scheme successfully delineates salt structures on 2D and 3D seismic images. While fully automatic segmentations are sometimes successful, limited manual interpretations on one or more 2D slices can be used to guide a 3D, semi-automatic segmentation process. This allows for improved results throughout the image cube, not just near manual pick locations. The new algorithm performs extremely efficiently compared to other automatic interpretation techniques, and operates on the full seismic image or cube rather than a limited domain around a salt structure. These advantages could make it an important tool for interactive interpretation procedures and help streamline the model building workflow.

ACKNOWLEDGMENTS

I would like to thank Unocal/Chevron and Schlumberger Multiclient for providing the field datasets and images used here for examples, and SmaartJV for providing the Sigsbee synthetic model.

Chapter 3

Efficient velocity model evaluation

Building an accurate seismic velocity model is essential for obtaining an acceptable image of the subsurface. When the subsurface is especially complex, for example in geological settings dominated by irregularly-shaped salt bodies, this task becomes particularly challenging. The large contrast between salt and sediment velocities magnifies the effects of inaccurate salt interpretation, resulting in a poor image. Unfortunately, velocity model-building is a time-consuming process that often requires several iterations. A typical salt-interpretation and model-building workflow involves iterative sediment- and salt-flood migrations to identify the top and base of the salt bodies (Mosher et al., 2007). In situations where the top or (especially) base salt interpretation is uncertain or ambiguous, several different salt scenarios may be geologically feasible. Therefore, a means of quickly testing the effects of several different possible velocity models would be extremely useful for judging and refining salt interpretations. In this chapter, I propose a Born modeling and migration scheme that allows for fast remigrations of data synthesized from an initial image, while incorporating prestack velocity information from the initial image's subsurface offset gathers.

An interactive interpretation and imaging environment would be a valuable model-building tool, and several different approaches have been proposed. Wang et al. (2008) introduced a fast migration scheme built on Gaussian beam imaging (Hill, 1990), that

can quickly test different salt scenarios. This method relies on seismic demigration and redatuming of wavefields to reduce the computational expense of migrating with several different velocity models. However, this method operates in the poststack image domain, neglecting velocity information that can be obtained from prestack images, and is limited by the approximations inherent to beam imaging. A similar approach has been proposed using reverse time migration (RTM) in a “layer-stripping” manner (Wang et al., 2011), but this remains too computationally intensive to test more than a very small number of possible models. Chauris and Benjema (2010) have proposed another method using RTM, which aims to reduce computational expense by summing over time-delays in the subsurface rather than sources. However, at present this method has only been demonstrated in two dimensions, and it remains unclear if an extension to 3D is feasible. Finally, fast migrations may also be achieved through the use of simulated datasets derived from an initial image. Guerra (2010) synthesized wavefields using prestack exploding reflector modeling (Biondi, 2006) as a means for performing wave-equation tomography in the image space. While the amount of preprocessing required for the velocity-updating portion of this method, especially in 3D, makes it less appealing for interactive velocity modeling, the idea of using prestack images to synthesize datasets is central to the scheme presented here.

Born modeling (Stolt and Benson, 1986) is based on a single-scattering approximation of the wave equation. By taking advantage of this approximation, we can simulate a new dataset (Tang and Biondi, 2011; Tang, 2011) from an initial image, whose size and acquisition geometry can be selected independently from those of the original dataset. Furthermore, the simulated data can be migrated using generalized sources, drastically reducing the number of shots required. In most of the examples I show, only a single shot is required, allowing for migrations well within an interactive time frame. In order to improve the accuracy of this method, I use a generalized source function derived from subsurface offset gathers of the initial image. This allows for a more accurate and data-driven result than if a simple wavelet were used as the source function; in addition, including non-zero subsurface offset information into this source function incorporates important velocity information available from the initial image. In this way, inaccuracies in the initial velocity model can be identified

and improved in subsequent iterations.

In the following sections, I review the Born modeling methodology and outline the procedure for obtaining the generalized source and receiver wavefields mentioned above. I then demonstrate the method using simple 2D synthetic models. Crosstalk artifacts arising from the modeling procedure limit these examples to isolated image points along subsurface reflectors; however, these tests show that the method can effectively provide information about the effects of different velocity models on an image. Finally, I apply the method to 2D and 3D portions of a field dataset from the Gulf of Mexico. To assist in interpreting the results, I also introduce a quantitative measure of image quality based on the proportion of an image’s energy focused at or near zero subsurface-offset. Even using an initial image created with an intentionally distorted velocity model, the method successfully identifies a more correct model. As we will see in Chapter 4, the efficiency of this method makes it a potentially powerful tool for interactive interpretation and imaging, especially when integrated with model building tools like image segmentation.

METHOD

The goal of the procedure I will describe is to use Born modeling to synthesize a new dataset that is much smaller than the original dataset used to generate an initial migrated image. Since the synthesized data can be “recorded” at any location in x , y , and even z , this procedure is effectively target-oriented. There are three basic steps needed to reach our goal of efficient velocity model evaluation:

1. Generate an areal source function using one or more subsurface offset gathers from the initial prestack image.
2. Using the new source function and a reflectivity model based on the initial image, employ Born modeling to generate a new dataset with acquisition geometry best suited to image the target area.
3. Migrate the simulated data obtained in Step 2, using the source function from

Step 1. This step is extremely computationally efficient compared to a full migration of the original data, allowing for testing of several possible velocity models in a fraction of the time it would take to evaluate them using standard migration techniques.

In the following sections I detail the theoretical basis for each of these steps.

Generalized source function

In conventional modeling and migration, a simple wavelet or plane wave is often used as the source function. However, here we can take advantage of the fact that the procedure described above begins with a migrated image. This allows us to perform post-stack “exploding reflector” (Claerbout, 2005) modeling of a reflector or point diffractor in the subsurface; the upward-continued wavefield can be recorded at any location, and then injected as an areal source function during Born modeling. Mathematically, this areal source is described as

$$S(\mathbf{x}_s, \omega) = \sum_{\mathbf{x}'} \sum_{\mathbf{h}} G_{v_0}^*(\mathbf{x}' - \mathbf{h}, \mathbf{x}_s, \omega) m(\mathbf{x}', \mathbf{h}), \quad (3.1)$$

where $\mathbf{x}_s = (x_s, y_s, z_s)$ are the arbitrarily defined locations where the wavefield will be recorded; \mathbf{h} is the vector of subsurface half-offsets; ω is angular frequency; \mathbf{x}' is the location of the exploding image point in the subsurface; and G_{v_0} is the Green’s function connecting the source to the image point (here, * denotes the adjoint). The Green’s function is computed using the same velocity model (v_0) that was used to image the originally-recorded data, meaning that the recorded wavefield should be independent of the original velocity model choice. However, since this velocity model is unlikely to be correct, the initial image should contain valuable information about the accuracy of this model in the form of subsurface offset gathers. Subsurface offset gathers from true-velocity images will have all energy focused at zero subsurface offset, while an incorrect velocity model will result in a “smearing” of energy to nonzero offsets (Biondi, 2005). Thus, the inclusion of the subsurface offset term \mathbf{h} in

equation 1 is designed to incorporate this information into the modeling. To illustrate the advantage gained by incorporating this information, Figures 3.1(a) and 3.1(b) show two recorded source wavefields from an image point that is actually located at $z = 1000$ in the subsurface, but was initially imaged with a velocity that was 15% too slow. Both recorded wavefields have been reverse-propagated back to zero time to facilitate comparison. The source function in panel (a) was modeled using only the zero subsurface offset $h = 0$ data from the initial image, while the result in panel (b) uses the non-zero offset information as written in equation 1. When only zero subsurface offset data are used, the source appears to focus at the incorrect depth; when the nonzero offset data are used, the effects of using the wrong velocity are apparent. Using the source function in Figure 3.1(b) therefore supplies more useful information for the Born modeling and migration scheme described in the next section.

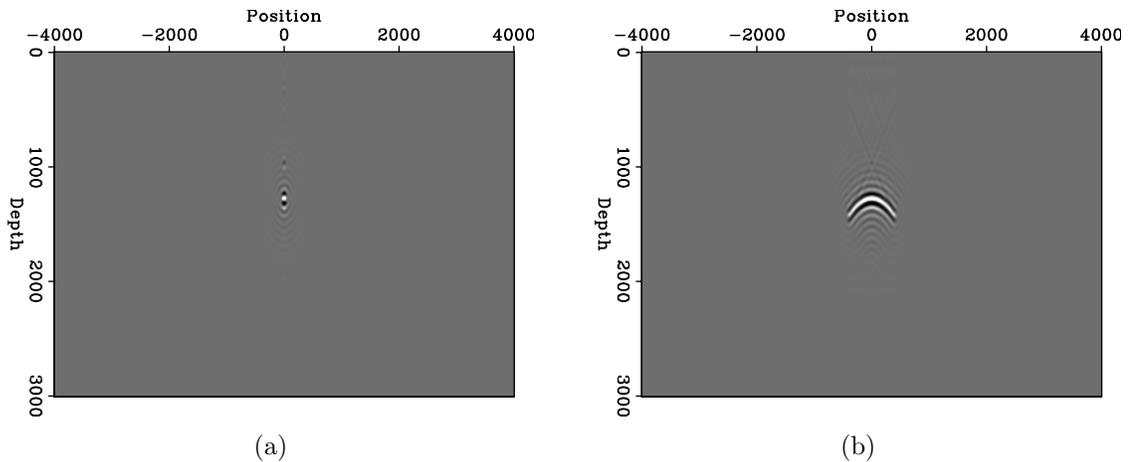


Figure 3.1: Recorded source wavefields that have been reverse-propagated to zero-time; the result in (a) does not include information from the nonzero subsurface offsets of the initial image, while (b) does include this information. Both the initial migration and the modeling used a velocity model that was 15% too slow. [ER]

chap3/. pt-0,pt-n0

Born modeling and migration

I now use the modeled areal source to generate a new data set via Born modeling. To do this, I define the simulated dataset d' recorded at arbitrary receiver locations \mathbf{x}'_r :

$$d'(\mathbf{x}_r, \omega) = \sum_{\mathbf{x}'} \sum_{\mathbf{h}} \Gamma(\mathbf{x}_s, \mathbf{h}, \omega) G_{v_0}(\mathbf{x}' + \mathbf{h}, \mathbf{x}'_r, \omega) m(\mathbf{x}', \mathbf{h}). \quad (3.2)$$

Here, m is the reflectivity model (in our case, the initial image), and the Γ term is defined as

$$\Gamma(\mathbf{x}_s, \mathbf{h}, \omega) = \sum_{\mathbf{x}_s} S(\mathbf{x}_s, \omega) G_{v_0}(\mathbf{x}_s, \mathbf{x}' - \mathbf{h}, \omega), \quad (3.3)$$

where S is as defined in equation 3.1. Crucially, the Green's functions in equations 3.1 and 3.2 are computed using the same velocity model - the one used to generate the initial image. As proven in Tang (2011), this means that the “data” synthesized using equation 3.2 will be kinematically invariant of this initial model. Even though this model is likely to be wrong, we can still confidently make use of the data synthesized from it.

Because the placement of the receiver locations in equation 3.2 can be arbitrarily determined, they do not necessarily need to be on the surface, like the original recorded data. Placing the receivers at depth can improve the efficiency of this method by providing the capability for target-oriented imaging; if a velocity model is well-determined down to a given depth, the synthesized data can be recorded below that depth, avoiding unnecessary computation. This has a similar effect to re-datuming the wavefields, an approach taken by Wang et al. (2008) in their fast image updating strategy.

Now that we have new source and receiver wavefields, we can produce an image using standard wave-equation migration techniques:

$$m'(\mathbf{x}', \mathbf{h}) = \sum_{\omega} G_{v_1}^*(\mathbf{x}' - \mathbf{h}, \omega) \sum_{\mathbf{x}_r} G_{v_1}(\mathbf{x}' + \mathbf{h}, \mathbf{x}'_r, \omega) d'(\mathbf{x}_r, \omega). \quad (3.4)$$

It is important to note that this formulation is a conceptual description of wave-equation migration; in practice, a computer propagates the wavefields in discrete time steps and correlates them to form an image. Crucially, the Green’s functions in equation 3.4 can be computed using any velocity model (v_1), and not necessarily the same one used to generate the source and receiver wavefields in previous steps. This can allow for testing of multiple possible velocity models. Furthermore, since subsurface offset gathers are generated during the imaging, we can develop a quantitative means of judging the accuracy of these various models. We expect a “perfect” velocity model to focus all an image’s energy at zero subsurface-offset. Therefore, a simple measure of image quality calculates what proportion of the energy indeed resides at zero or near-zero subsurface-offset:

$$F = \frac{\sum_{i=\mathbf{p}} |A_i|}{\sum_{i=\mathbf{p}} |A_i| \exp(\alpha \frac{|h_i|}{h_{\max}})}, \quad (3.5)$$

where \mathbf{p} is the set of all image points, A_i is the amplitude at a given point, h_i is the subsurface offset at that point, and α is an optional user-specified weighting parameter. Using this measure, a value of $F = 1$ means that all energy is perfectly focused at zero offset; as F decreases toward zero the image becomes progressively less focused. Ideally, a measure such as this one would allow a more rigorous comparison among possible models when a more qualitative comparison fails to yield an obvious result.

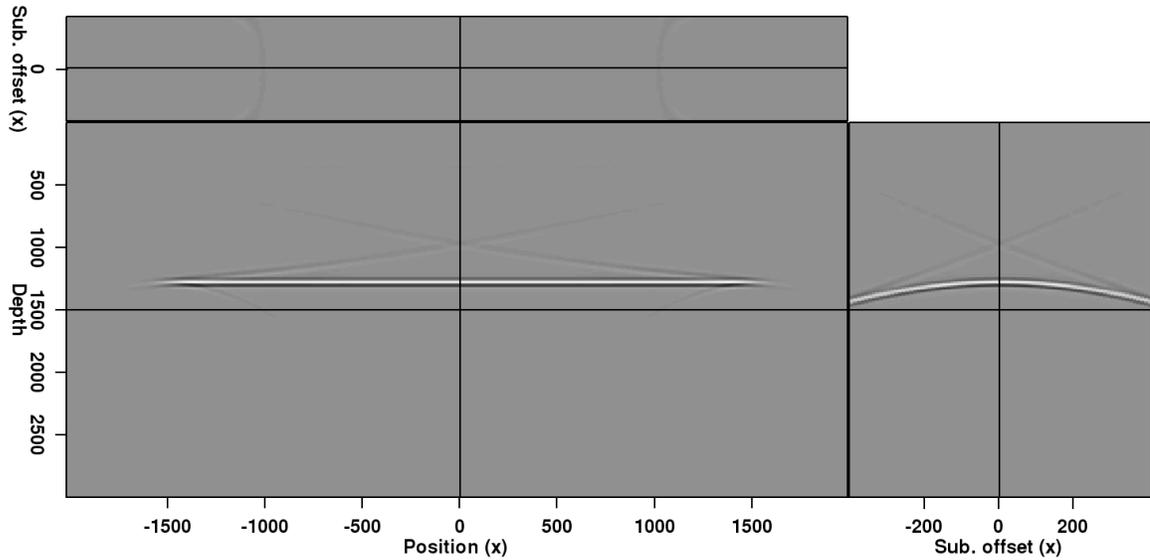
Unfortunately, the imaging procedure as written in equation 3.4 can also generate crosstalk artifacts, since areal source data is used. While various methods such as phase-encoding (Romero et al., 2000; Tang, 2009) are available to help attenuate these artifacts, we restrict our examples in the next section to isolated points in the subsurface, spaced far enough apart to limit the effects of crosstalk. This issue will be explored in more detail in the following sections.

SINGLE-REFLECTOR TESTS

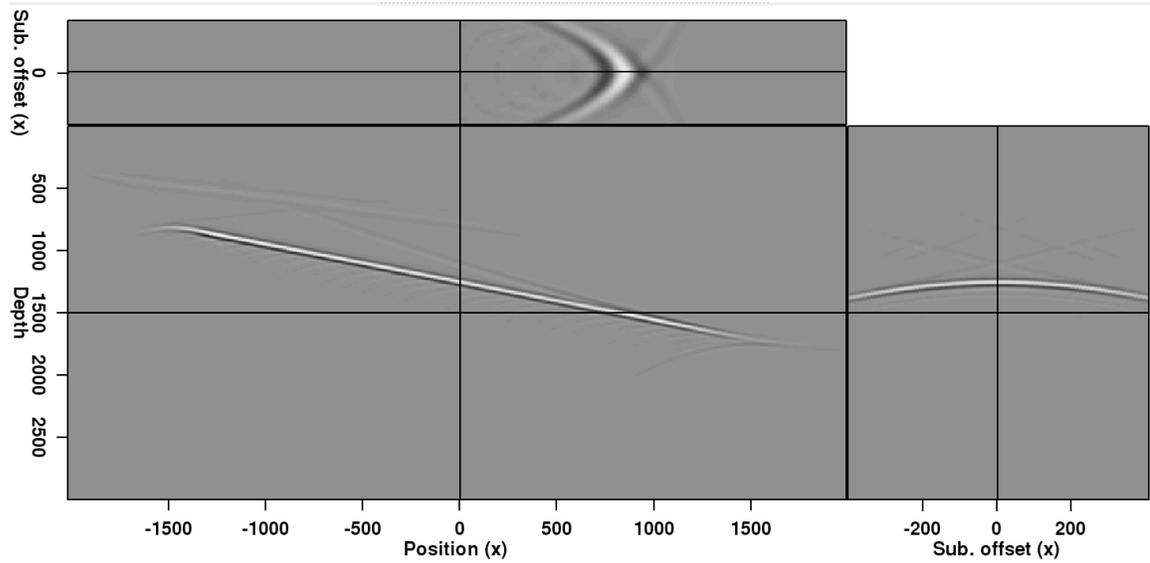
To demonstrate the feasibility of the method outlined above, I will first investigate two simple synthetic test cases: a single flat reflector in the subsurface (Figure 3.2(a)), and a single reflector dipping at 20° (Figure 3.2(b)). Both examples in Figure 3.2 were generated by migrating with an incorrect velocity model (15% slower than the constant-velocity model used to generate the original dataset). The effects of using an incorrect velocity can be seen clearly on the subsurface offset gather (non-focused event). A key goal of the Born modeling procedure is to replicate this behavior when the same velocity model is used to migrate the Born-modeled data. To test this, I sample isolated points from the reflectors in Figure 3.2, and use these points to generate the areal source function described in the previous section. In order to avoid unwelcome crosstalk between these points during the modeling process, they are separated by a distance that is twice the maximum subsurface offset, as seen in Figure 3.3.

Once the source function is “recorded,” Born modeling is performed using the sub-sampled images in Figure 3.3 as reflectivity models. The results of migrating this Born-modeled data, using the same velocity model used to produce the images in Figure 3.2, are seen in Figure 3.4. Because these images were migrated using an areal source function, only a single shot was necessary; this means that the images in Figure 3.4 were produced in seconds, nearly three orders of magnitude less time than was necessary to compute the images in Figure 3.2. Comparing the subsurface offset gathers for both of these figures, we see that while amplitudes differ, the kinematics have been accurately preserved in the Born-modeled result. If our goal is to evaluate the velocity model used, the quickly-obtained results in Figure 3.4 should be sufficient.

As mentioned in the previous section, an advantage of this Born modeling strategy is that the synthesized data may be recorded at any depth, effectively re-datuming wavefields prior to migration. This can lead to significant computational savings, especially if velocities are well known until a certain depth. To verify that this capability does not effect the accuracy of migration results, I recorded both the areal

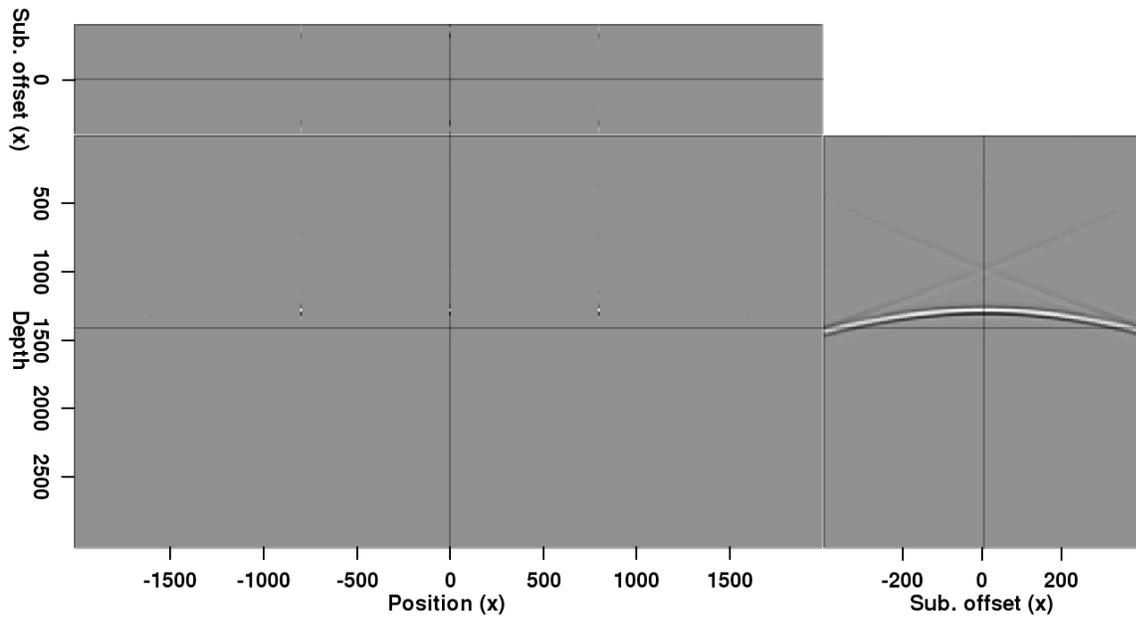


(a)

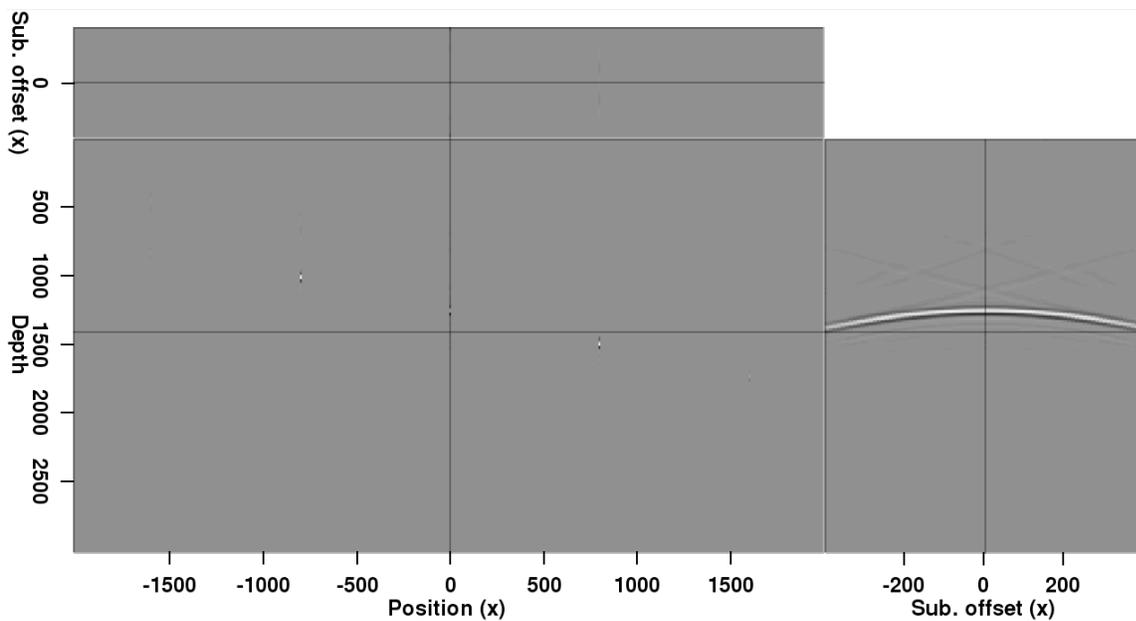


(b)

Figure 3.2: Prestack depth migration images of (a) a flat reflector and (b) a reflector dipping at a 20° angle. The images were migrated with a constant velocity 15% too slow compared to the true velocity, causing the noticeable artifacts and lack of focusing in the subsurface offset dimension. [ER] `chap3/. flat-orig,dip-orig`

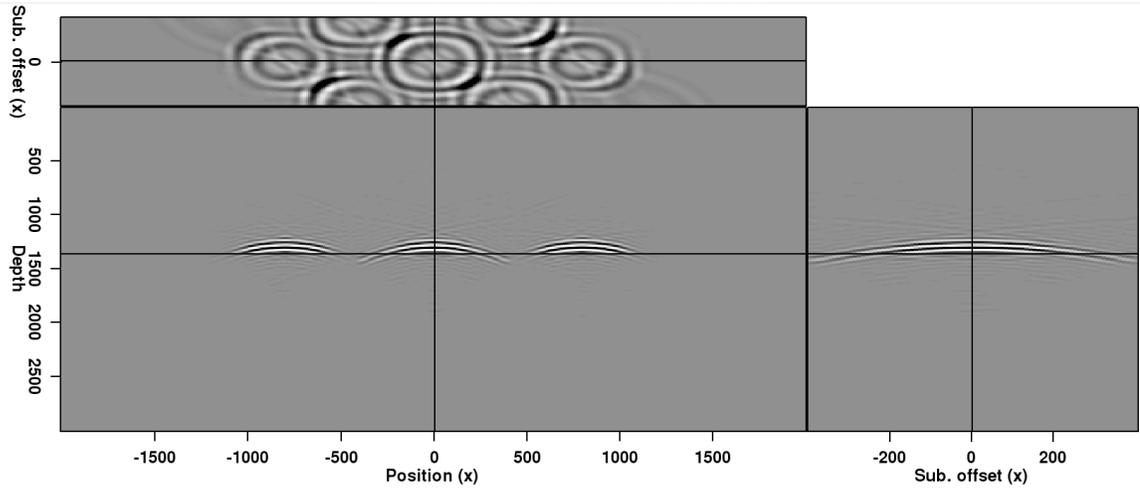


(a)

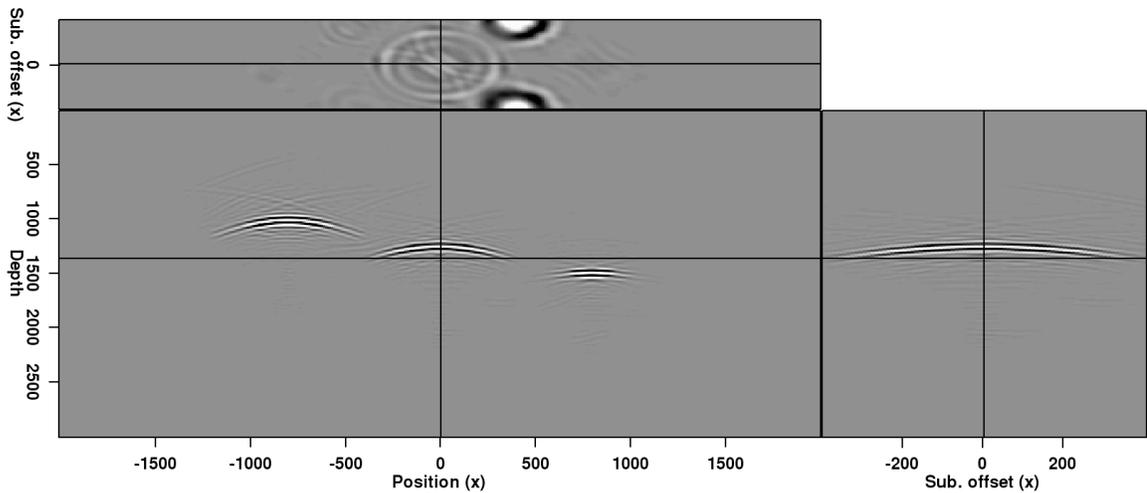


(b)

Figure 3.3: Isolated image points from Figures 3.2(a) and 3.2(b) used for the modeling procedure. The points are separated by twice the maximum subsurface offset value in order to avoid crosstalk artifacts in the modeling. [ER] chap3/. flat-sp,dip-sp



(a)



(b)

Figure 3.4: Migrated images after Born modeling using the images in Figure 3.3 as the reflectivity model. Although the amplitudes differ, the kinematics of the events in both figures match. [ER] chap3/. flat-sp-born,dip-sp-born

source wavefield and the Born-modeled data at depth $z = 750$, instead of at the surface. Figure 3.5 shows the result of migrating this data in the dipping reflector case. Comparison with Figure 3.4(b) confirms that the two results are virtually identical for the area of interest.

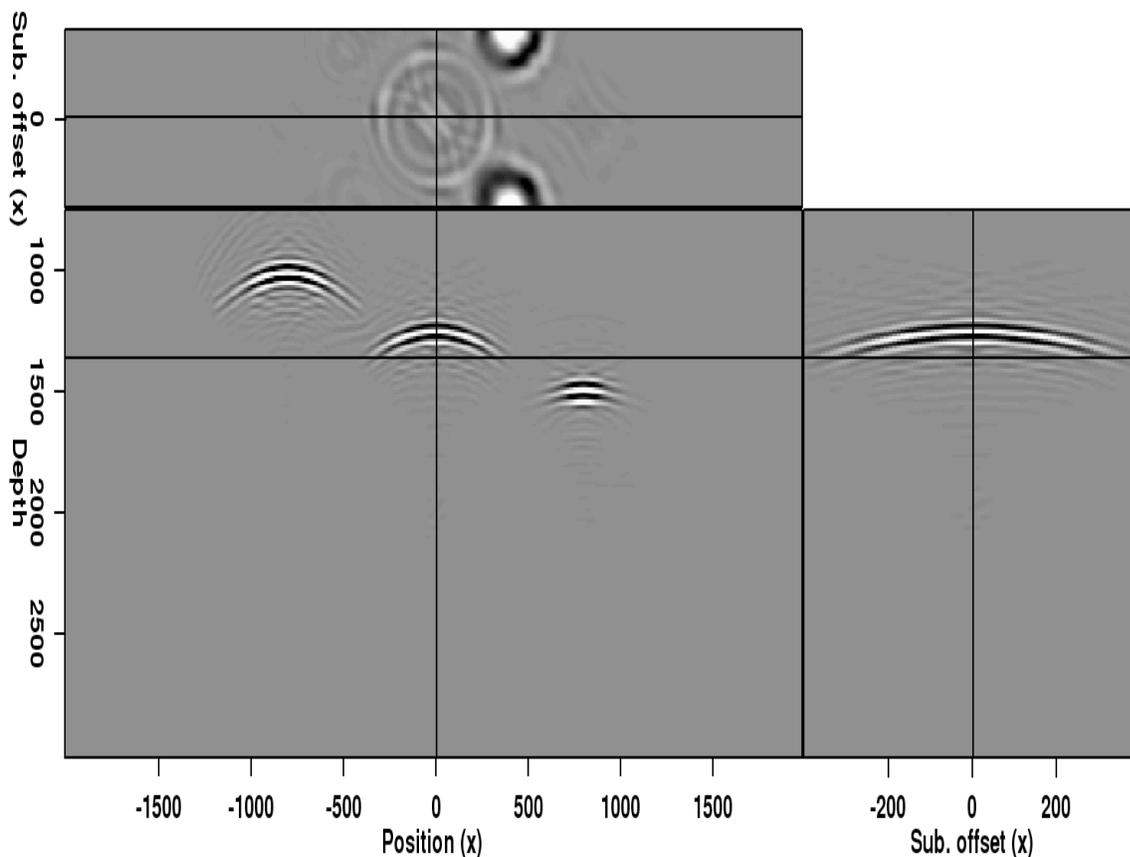


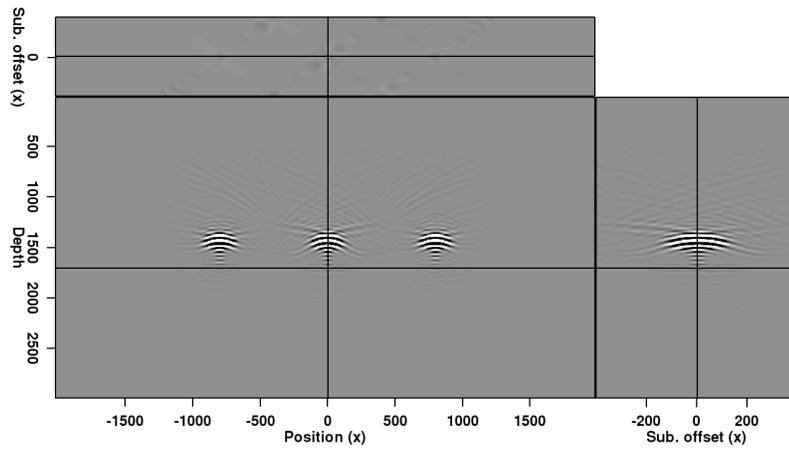
Figure 3.5: Migration result using Born-modeled data from the model in Figure 3.3(b). In this case, the synthesized data was recorded in the subsurface instead of on the surface, effectively re-datuming the wavefields. [ER] chap3/. dip-short

Finally, we wish to test the ultimate purpose of this method: quickly evaluating multiple velocity models. Once the Born-modeled dataset has been synthesized, we can use any velocity model to image the data. Again, we are able to perform these migrations very quickly, on the order of seconds for the examples here. Figure 3.6

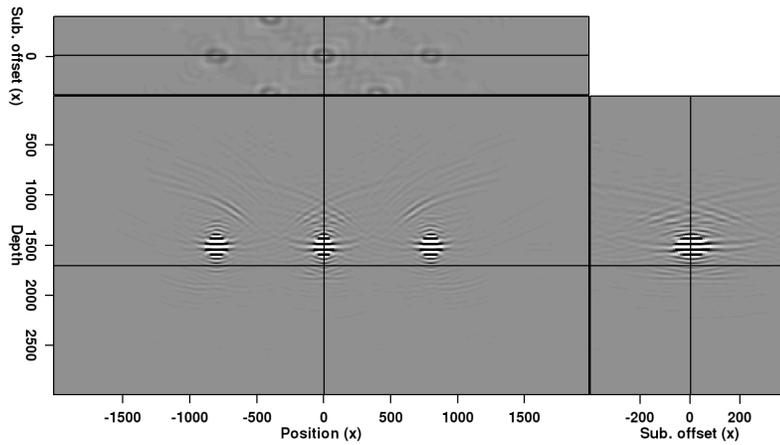
compares the results of using three different velocity models to image the Born-modeled data: one that is 5% slower than the true velocity (Panel a); one that is exactly the true velocity (Panel b); and one that is 5% faster than the true velocity (Panel c). From these results, it is clear that the velocity model used to produce Panel b's result is the most accurate – the subsurface offset gather is flat and relatively focused, and, unlike Panels a and c, there are no signs of over- or under-migration on the zero-subsurface offset image. Because the velocity differences between these three models are relatively small, this is an encouraging sign that this method can ultimately allow us to quickly test more complex models for both synthetic and field data.

Crosstalk mitigation

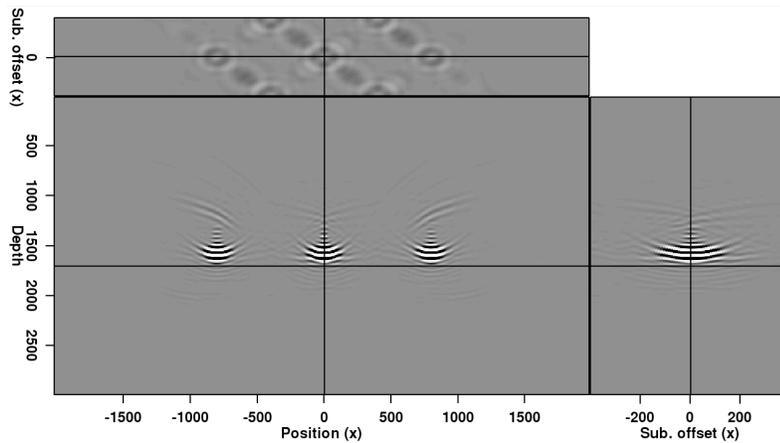
The importance of correctly spacing the image points used for the modeling is illustrated in Figure 3.7. Here, locations from the flat reflector image in Figure 3.2(a) have been sampled twice as frequently, at a spacing equal to the maximum subsurface offset. Figure 3.7 shows the result of using these points to create the areal source function, and then performing Born modeling and migration as before. Now, crosstalk between the closely-spaced image points results in severe artifacts, including spurious events on the zero-subsurface offset image. While the necessity for isolated image locations is a limitation of this method, it can be partially overcome by performing two or more passes of the model evaluation procedure using different image locations. For example, Figures 3.8(a) and 3.8(b) show the results of performing the model evaluation procedure on, respectively, four and three properly-spaced locations from the flat reflector. Taken together, these locations are identical to those used to create the crosstalk-contaminated image in Figure 3.7. However, if the two sets of locations are imaged separately and then summed, the result (Figure 3.8(c)) is free of crosstalk artifacts. While the computational requirements for this method necessarily double for each set of locations imaged in this manner, the advantages gained by obtaining a clearer picture of the reflector may be worth the cost, and still much cheaper than performing full migrations.



(a)



(b)



(c)

Figure 3.6: Result of migrating the Born-modeled data with (a) 5% too slow velocity; (b) correct velocity; and (c) 5% too fast velocity. Each migration was nearly instantaneous, and the effects of the different velocity models are readily apparent. [ER] `chap3/. comp-slow,comp-act,comp-fast`

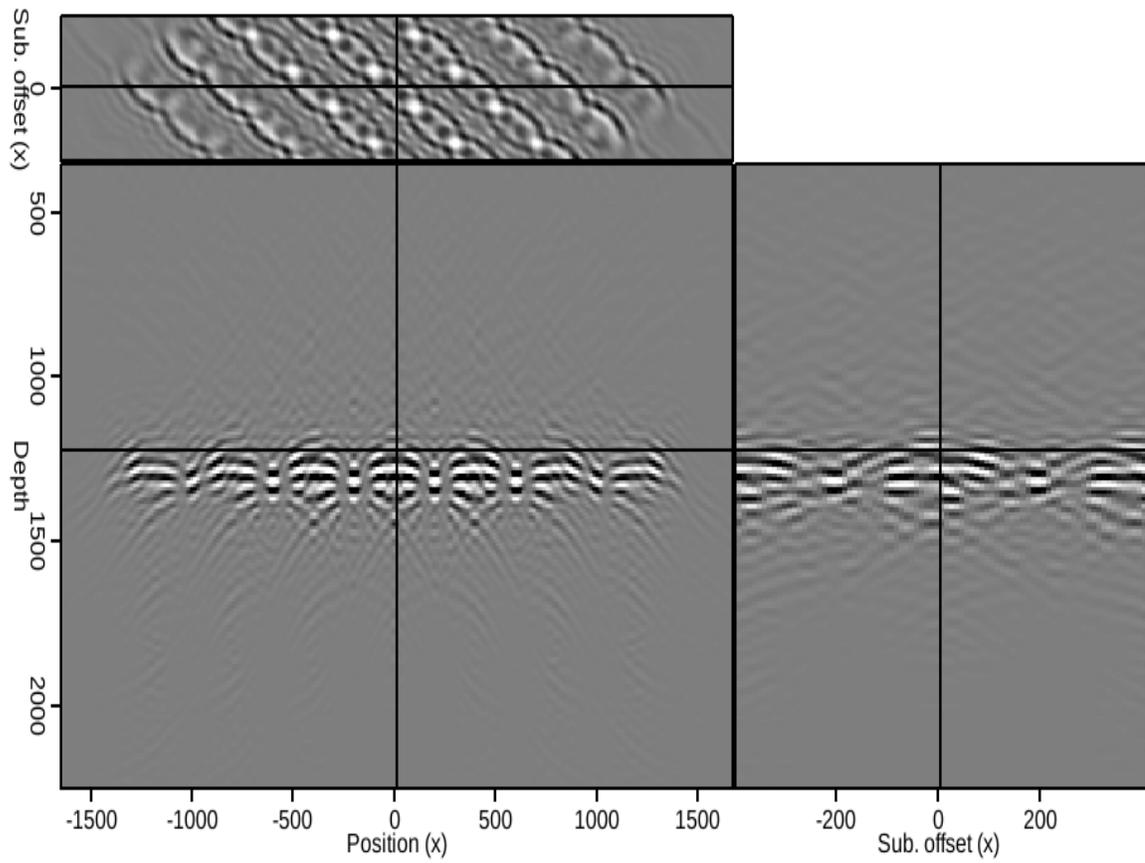
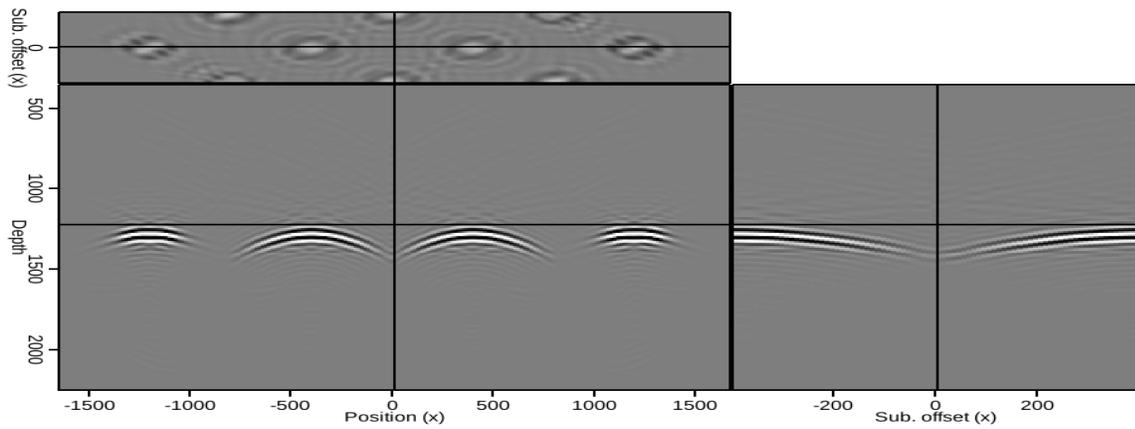
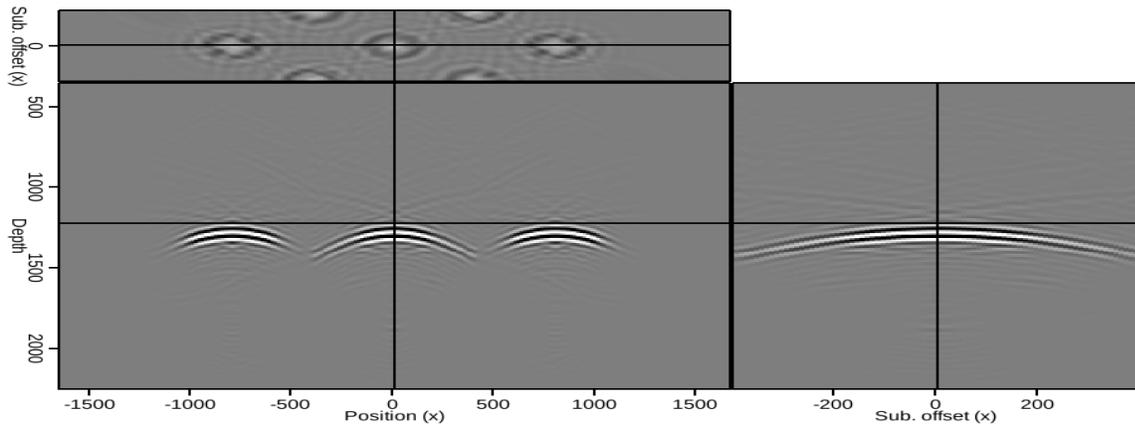


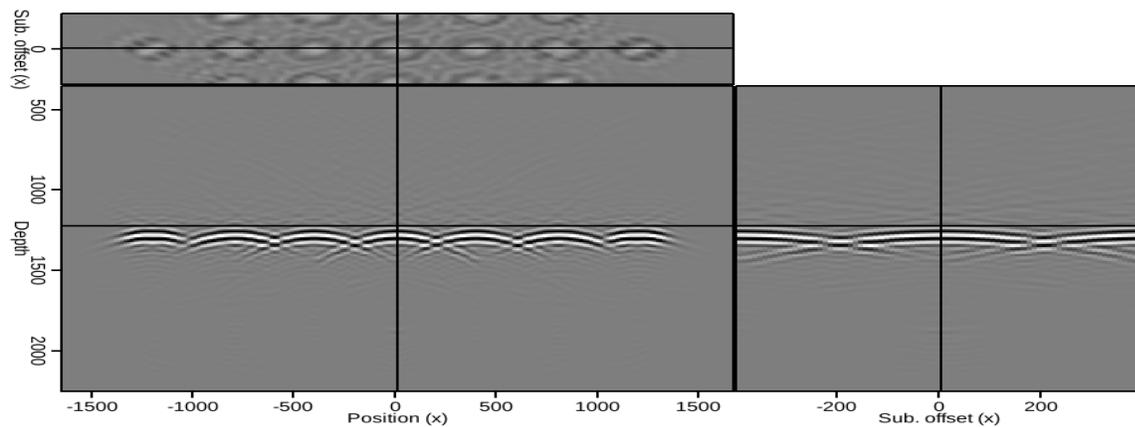
Figure 3.7: Migration result if the image points sampled from Figure 3.2(a) are spaced at less than twice the maximum subsurface offset. Crosstalk artifacts dominate the image, making interpretation extremely difficult. [ER] `chap3/. flat-xtalk`



(a)



(b)



(c)

Figure 3.8: Migration results using (a) four and (b) three image locations sampled from the flat reflector in Figure 3.2(a), and (c) the sum of these two images. Taken together, the sample points in panels (a) and (b) are identical to those used to produce the result in Figure 3.7; however, the image in panel (c) is free of the crosstalk artifacts seen in that result. [ER] chap3/. flat-a,flat-b,flat-sum

Results for a steeply dipping reflector are similar. When multiple, closely-spaced locations along a 40° dipping reflector (Figure 3.9(a)), migrated with a velocity slower than the true velocity, are imaged using a single shot, the result is made uninterpretable by strong crosstalk artifacts (Figure 3.9(b)). However, performing multiple experiments using proper spacing provides a much cleaner, more interpretable result when the images are summed (Figure 3.9(c)). In each of these examples, the slower velocity is used to both generate and image the synthesized wavefields. Taken to the limit, this procedure can even be used to fully reconstruct the reflector, if every location is used to synthesize the wavefields (Figure 3.9(d)). Figure 3.9(d) is not identical to Figure 3.9(a), mainly due to destructive interference of the “frowning” events in both the subsurface offset domain and at zero subsurface offset. A further option to help mitigate this interference is to restrict the subsurface offset domain data used to synthesize the source and receiver wavefields to only a few, specific offset values. The wavefields used to obtain the image in Figure 3.10(a) were generated using only data from $h = -200$, $h = 0$, and $h = 200$ for each x -location along the reflector in Figure 3.9(a). Now, not only is the reflector fully reconstructed at zero subsurface offset, but there is clear evidence in the subsurface offset gather that the velocity used was indeed too slow. Imaging the wavefields instead with the true velocity (Figure 3.10(b)) correctly yields an image focused at zero subsurface offset. Furthermore, the reduction in interference allows for a more faithful reconstruction of the original reflector in both Figures 3.10(a) and 3.10(b) than was possible in Figure 3.9(d), for which all the subsurface offset domain data were used.

This result suggests that particular care should be taken when applying this method to steeply dipping reflectors; however, the interpreter-driven nature of the workflow makes it simple for analysis to be constrained to relatively flat portions of a reflector if necessary. In any case, the accurate reconstruction of even the dipping reflector provides further evidence of the advantages of synthesizing multiple wavefields to image different parts of the reflector. The remaining examples in this chapter are limited to a single set of locations per reflector, but the 3D field data example in Chapter 4 will take advantage of this “split-and-sum” strategy.

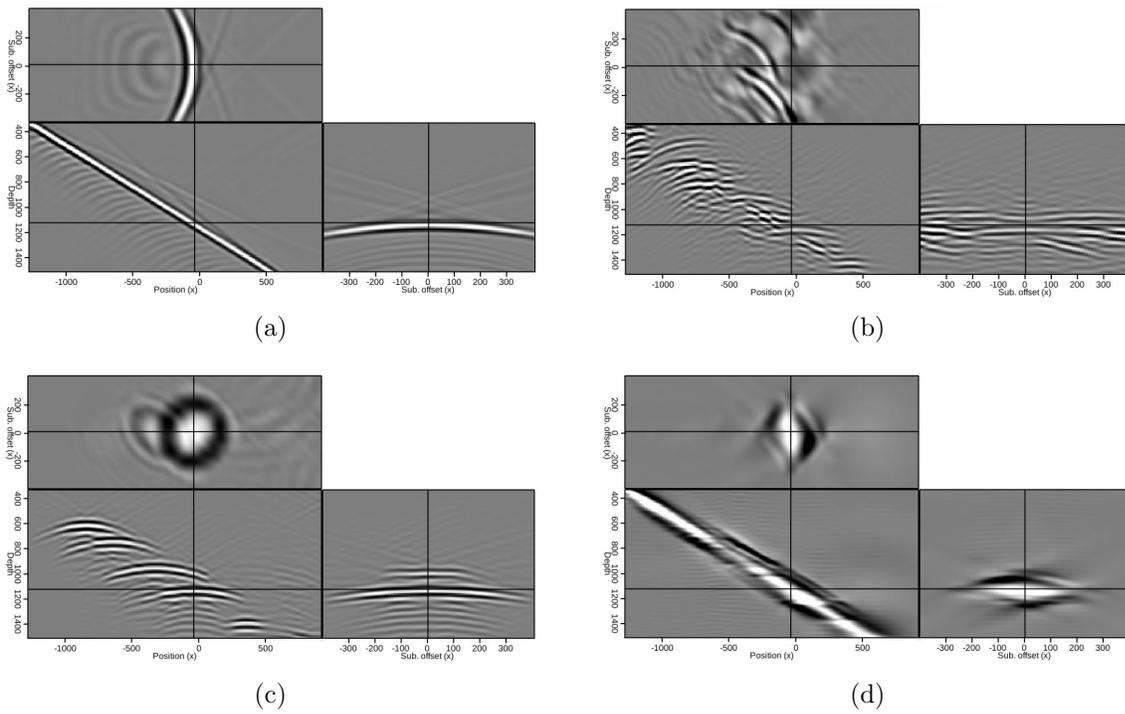
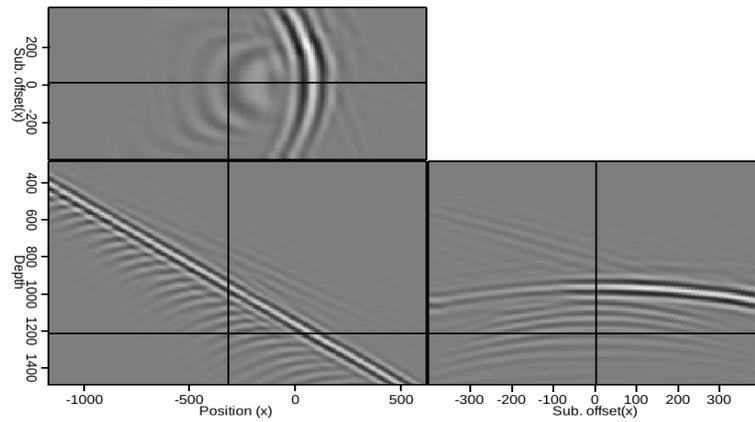
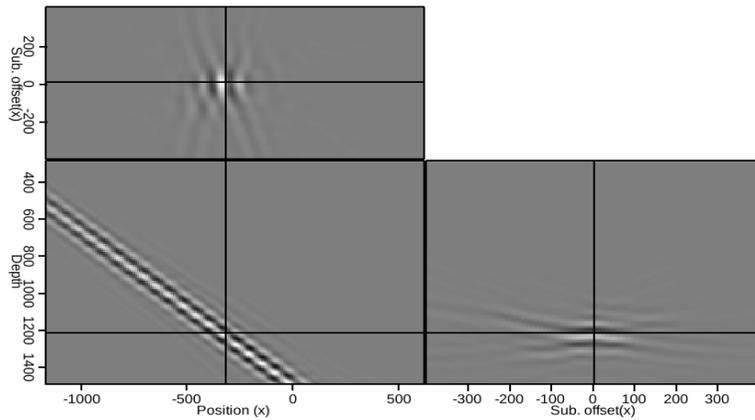


Figure 3.9: (a) Initial image of a 40° dipping reflector; (b) crosstalk-contaminated result of using a single source function derived from closely spaced locations along the reflector; (c) result of summing several images generated using wavefields synthesized from properly-spaced locations along the reflector; (d) full reconstruction of the reflector after using all locations along the reflector to synthesize wavefields and summing the individual results. [CR] chap3/. dip2,dip-xtalk,dip-cmb,dip-full



(a)



(b)

Figure 3.10: Results of imaging wavefields synthesized using every location along the reflector in Figure 3.9(a), but restricting the use of subsurface offset information to three offsets only ($h = 0, \pm 200$). In (a), the slow velocity was used to migrate the wavefields, while the true velocity was used in (b). [CR]

chap3/. dip-sel-slow,dip-sel-true

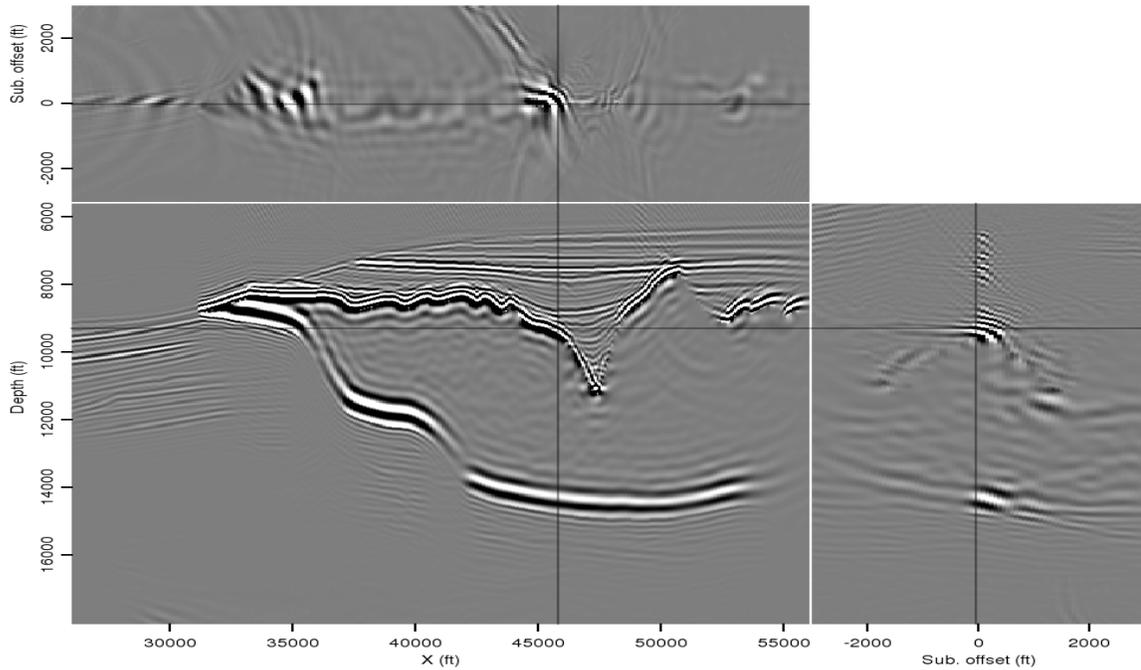
2D SYNTHETIC EXAMPLE

The above procedure will now be demonstrated using two different initial images derived from the Sigsbee synthetic model. Figure 3.11(a) is a perfect-velocity, full migration of the Sigsbee data, which will be used for the first example. Figure 3.11(b) shows a manually-picked reflector chosen for further analysis; in this case, the base salt has been chosen because it should be particularly sensitive to different interpretations of the salt body’s shape and velocity, the two model variations that will be tested. Finally, Figure 3.12 shows two image locations isolated from the selected reflector. Note that most of the energy is focused near zero subsurface-offset, since the true velocity model was used for the initial image.

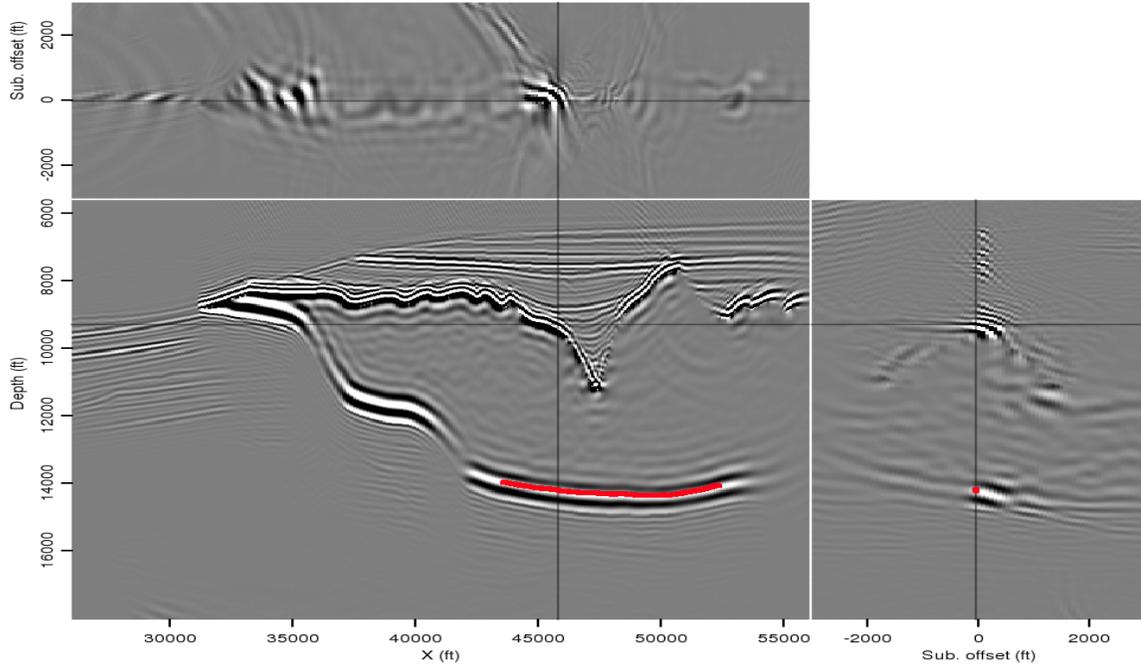
After synthesizing the source and receiver wavefields as described in the previous section, the new wavefields were imaged using three different velocity models. First, the true model, seen in Figure 3.13(a). Second, an alternative model created via automatic image segmentation, in which an interpreter has chosen to include an extra chunk of salt (Figure 3.13(b)). The third model tested was identical to the true model in salt shape, but with a salt velocity 5% slower than the true model.

Resulting images from the Born-modeled data are seen in Figure 3.14. Panel 3.14(b), the result of migrating with the extra-salt model in Figure 3.13(b), is clearly the least well-focused image. However, it is difficult to qualitatively distinguish between the other two results. Instead, the results of calculating the F value from equation 3.5 can provide a quantitative comparison of the images in Figure 3.14. These calculations are found in table 3.1; as expected, the result using the true model has a higher F value, indicating it is more well-focused.

The second example uses an initial image created with an incorrect velocity model; in this case, the “slow salt” model described above. The results corresponding to each velocity model can be seen in Figure 3.15. Again, the extra-salt model is clearly inferior, but the differences between the other two results are more subtle. The F -value calculations in table 3.2 confirm that the true model yields the optimal result, even though the slow-salt model was used to create the initial image and both synthesized



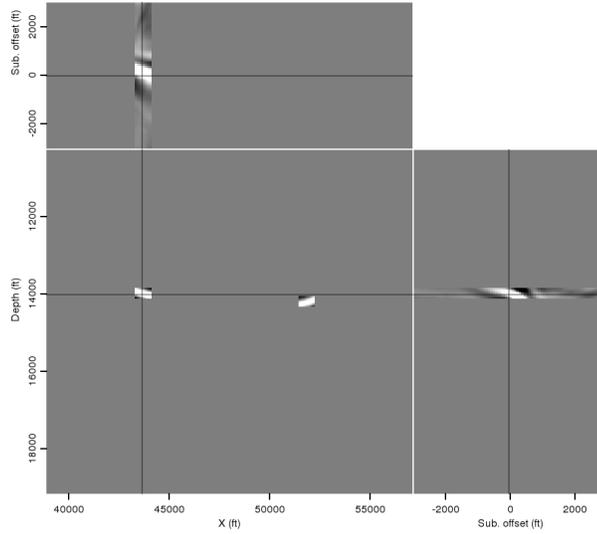
(a)



(b)

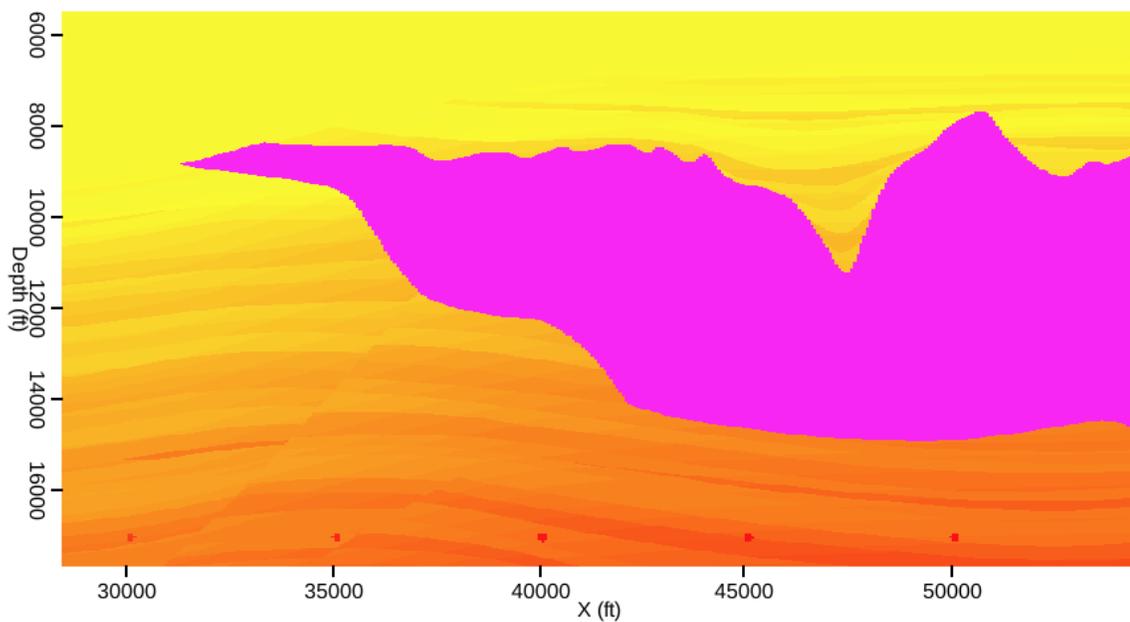
Figure 3.11: (a) A true velocity image using data from a section of the Sigsbee synthetic model; and (b) a base-of-salt reflector selected for further analysis because of its sensitivity to changes in the salt interpretation. [CR] chap3/.img-act,picks

Figure 3.12: Isolated image locations selected from the reflector picked in Figure 3.11(b). [CR]
 chap3/. act-pts

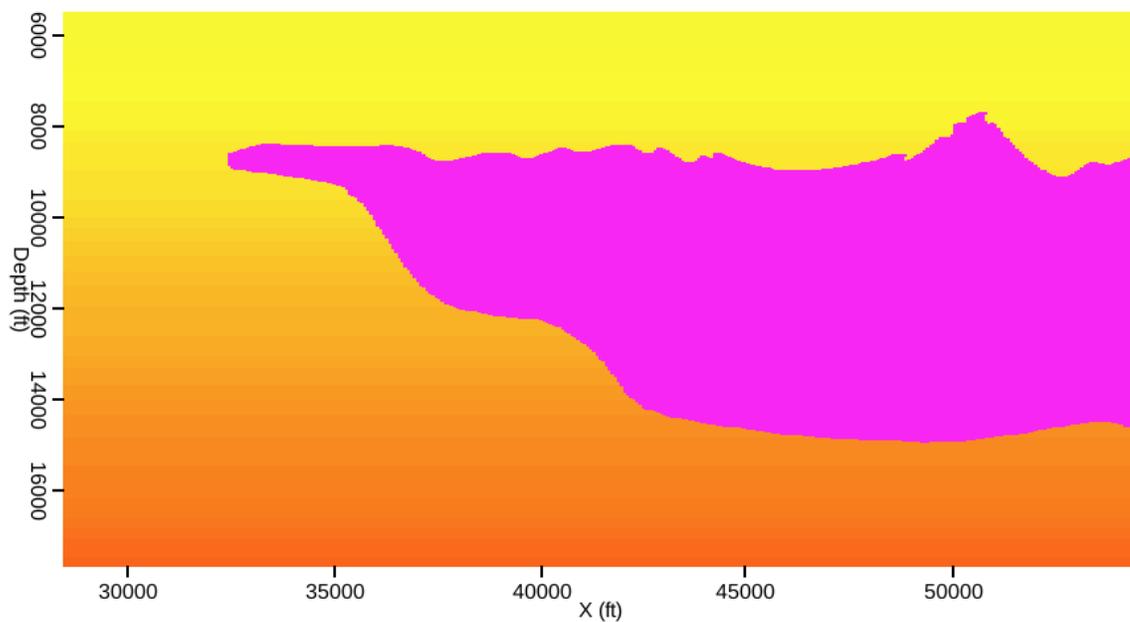


Migration model	F value
True velocity	0.883
Extra salt	0.879
Slow salt	0.864

Table 3.1: Calculations from equation 3.5 for each migration velocity model, after the initial image and synthesized wavefields were created using the true velocity model.



(a)



(b)

Figure 3.13: Two different velocity models to be tested. The model in (a) is the true Sigsbee model, while (b) is an alternate model created by one possible automatic segmentation of the initial image. [ER] chap3/. zvel-act,zvel-filled

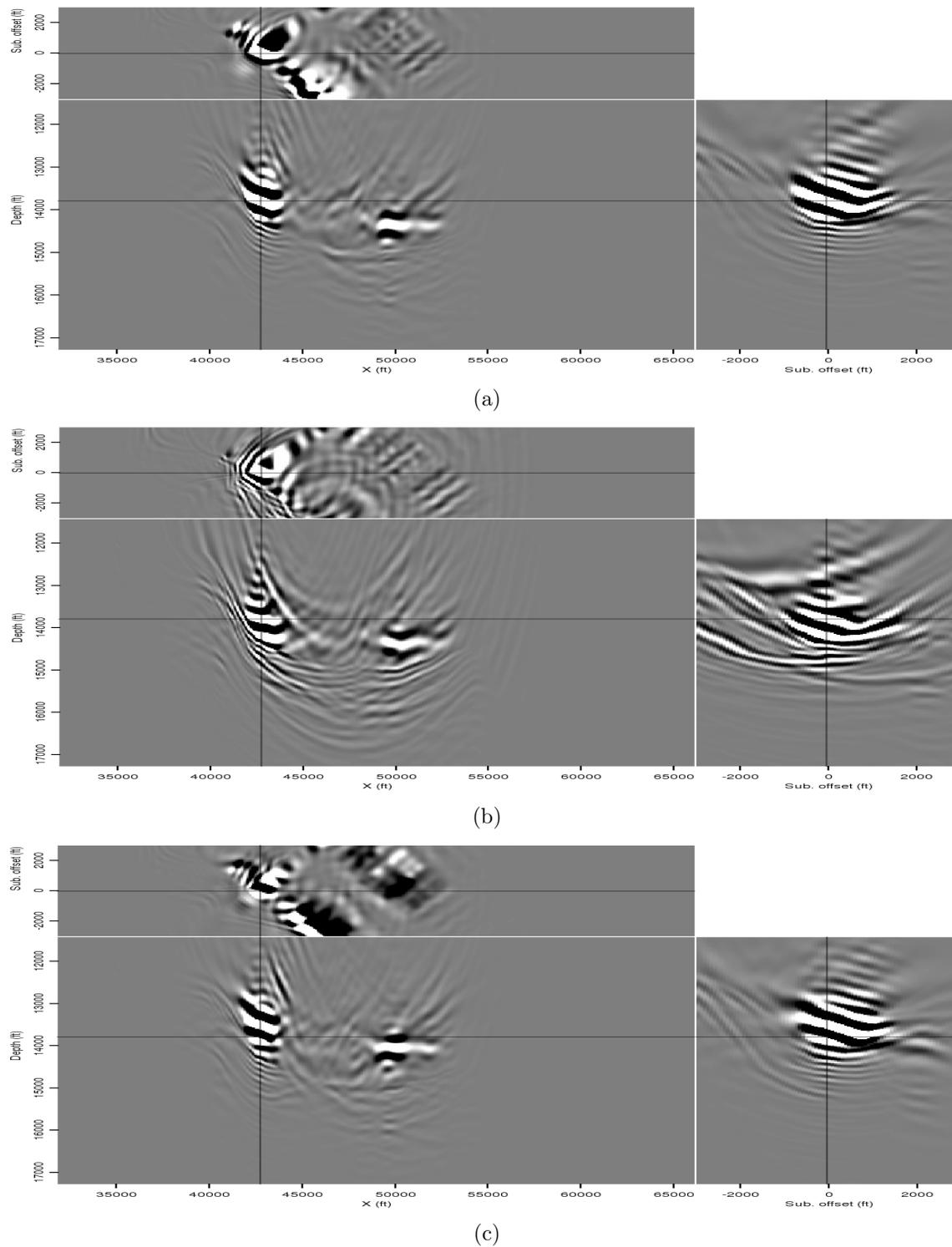


Figure 3.14: Three images of the Born-modeled data using three different migration velocity models: (a) the true model; (b) the extra-salt model; and (c) the slow-salt model. In this example, the initial image was created with the true model. [CR]

chap3/. born-act1,born-af,born-as

Migration model	F value
True velocity	0.621
Extra salt	0.561
Slow salt	0.613

Table 3.2: Calculations from equation 3.5 for each migration velocity model, after the initial image and synthesized wavefields were created using the “slow salt” velocity model.

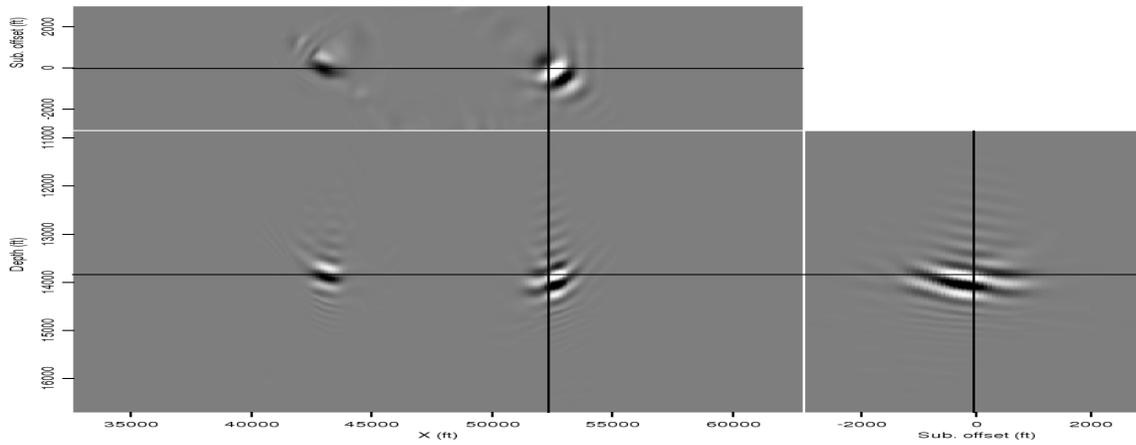
datasets.

2D FIELD DATA EXAMPLE

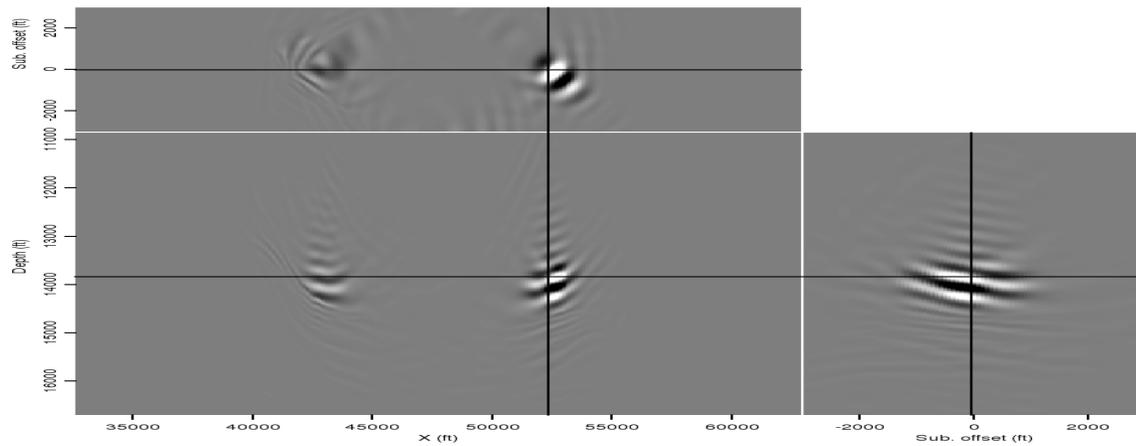
The 2D and 3D images used to demonstrate this method are derived from a wide-azimuth Gulf of Mexico dataset provided by Schlumberger Multiclient. Figure 3.16(a) is a 2D migration of a portion of this dataset, imaged using the velocity model provided with the data. To test the method, a single location along the reflector highlighted in Figure 3.16(b) was isolated and used to synthesize the source and receiver wavefields as described previously.

Figure 3.17 shows the results of imaging the new, synthesized wavefields with three different velocity models: the one provided with the data, and models scaled $\pm 5\%$ from the one provided. A qualitative examination suggests that the provided model produces the best-focused image, seen in panel (b). A more quantitative analysis using equation 3.5 confirms this; the F -values calculated for each of the images in Figure 3.17 are found in table 3.3.

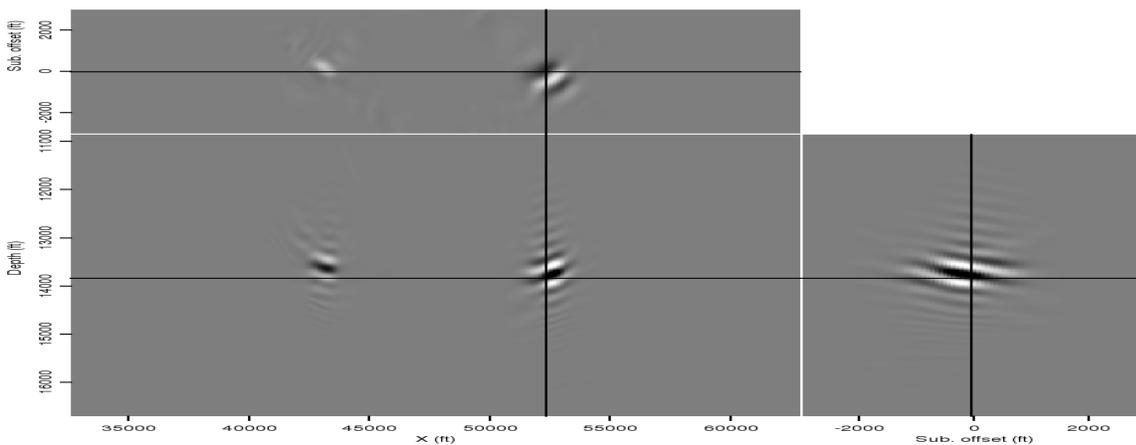
A more interesting test case is one in which the initial image was not generated using the provided velocity model. The image in Figure 3.18(a) was created using the slower velocity model of the previous example; a location isolated from the reflector indicated in Figure 3.18(b) was used to synthesize the new source and receiver wavefields. The same reflector is chosen in Figure 3.18(b) as in Figure 3.16(b), although the location is slightly shifted due to the difference in velocity models. Now,



(a)



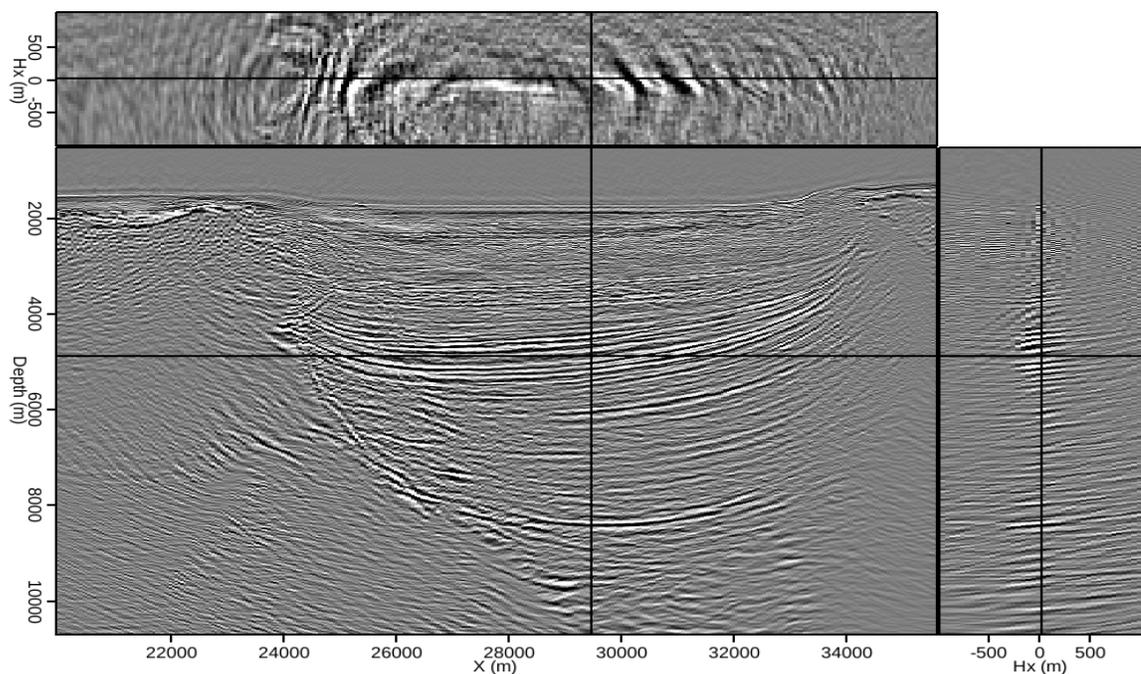
(b)



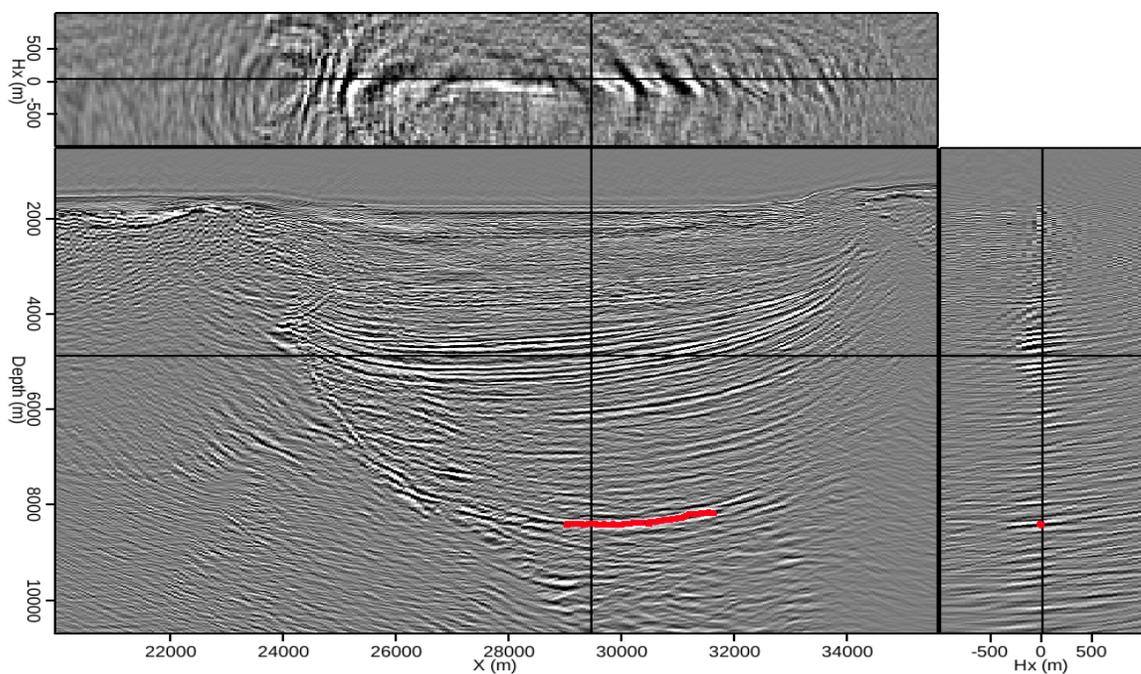
(c)

Figure 3.15: Three images of the Born-modeled data using three different migration velocity models: (a) the true model; (b) the extra-salt model; and (c) the slow-salt model. In this example, the initial image was created with the slow-salt model. [CR]

chap3/. born-sa,born-sf,born-ss



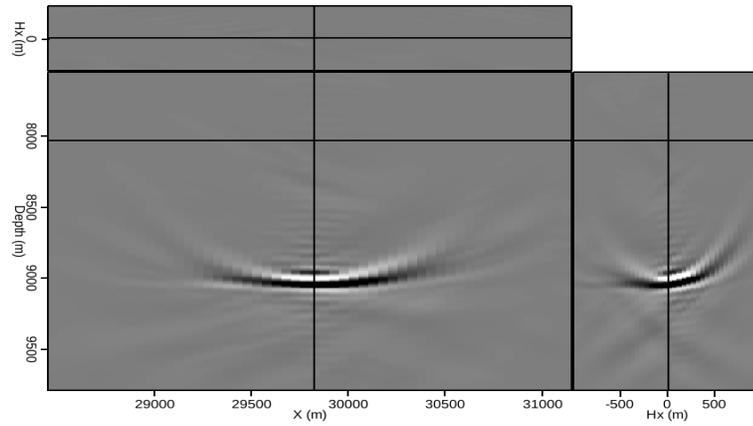
(a)



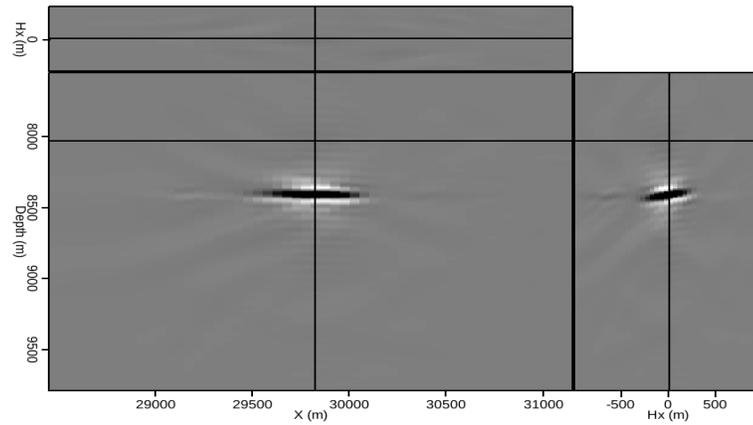
(b)

Figure 3.16: (a) A 2D field image produced using the provided velocity model, and (b) a manually-picked horizon of interest used to test three different velocity models.

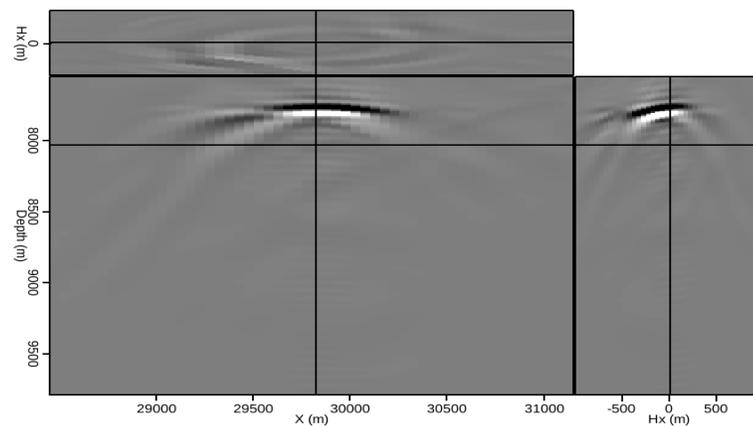
[CR] `chap3/. 2d-field,act-picks`



(a)



(b)



(c)

Figure 3.17: A single location from the reflector indicated in Figure 3.16(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% faster than the one provided; (b) the provided velocity model; and (c) a model 5% slower than the one provided. [CR] chap3/. born-fast,born-act,born-slow

Migration model	F value
Provided velocity	0.89
Fast model	0.86
Slow model	0.85

Table 3.3: Calculations from equation 3.5 for each 2D migration velocity model, after the initial image and synthesized wavefields were created using the provided velocity model.

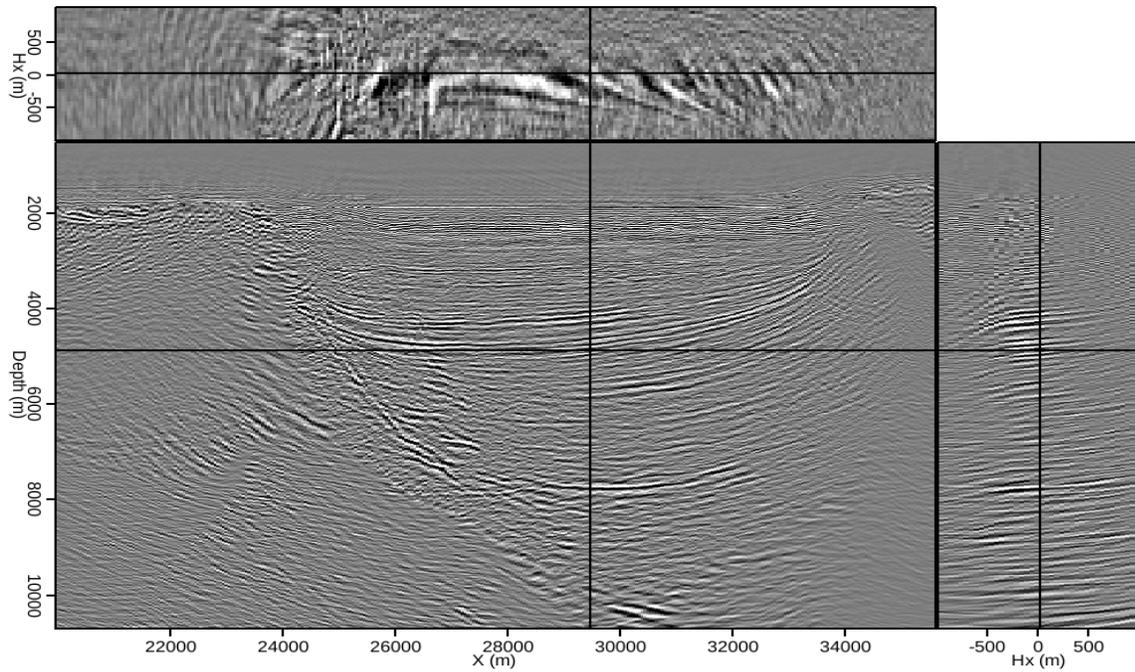
Migration model	F value
Provided velocity	0.92
Fast model	0.89
Slow model	0.89

Table 3.4: Calculations from equation 3.5 for each 2D migration velocity model, after the initial image and synthesized wavefields were created using a model 5% slower than the one provided.

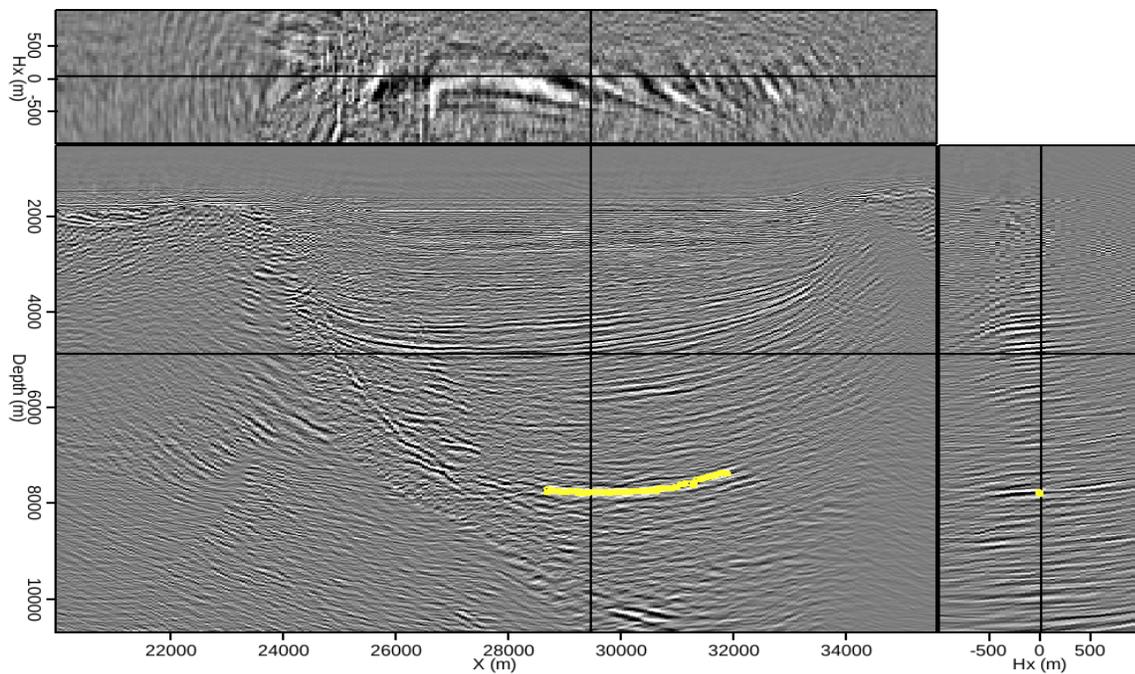
Figure 3.19 shows the results of imaging the synthesized wavefields from this initial image, with the same three models as the previous example. Again, the provided model (panel b) delivers the most well-focused result, both qualitatively and quantitatively (from the F -value results in table 3.4). This demonstrates that the method can identify a more accurate model using field data, even when the initial model is less accurate.

3D field data example

Finally, initial tests on 3D data were performed using the same strategy as in the previous section. Figure 3.20(a) shows a relatively small image cube derived from the same Gulf of Mexico dataset used for the 2D examples, and imaged using the provided velocity model. Again, a single location from the manually-interpreted reflector (this time, a 3D horizon) shown in Figure 3.20(b) was used to synthesize 3D areal source and Born-modeled receiver wavefields. Because the prestack image cubes resulting

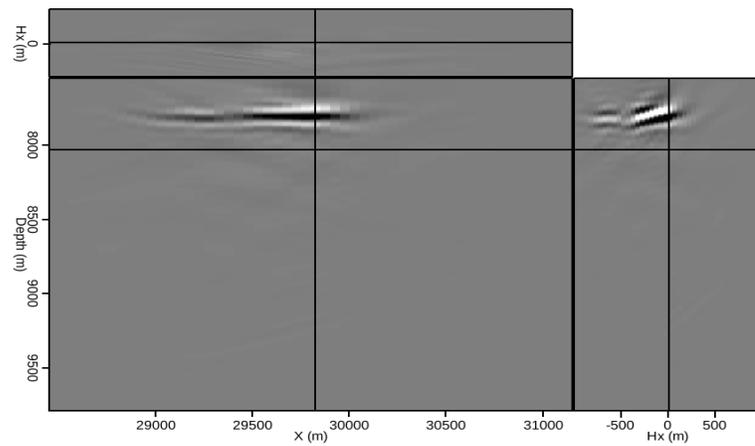


(a)

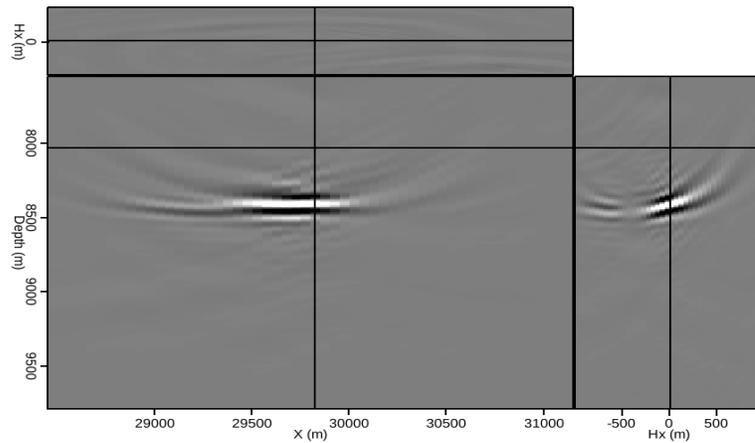


(b)

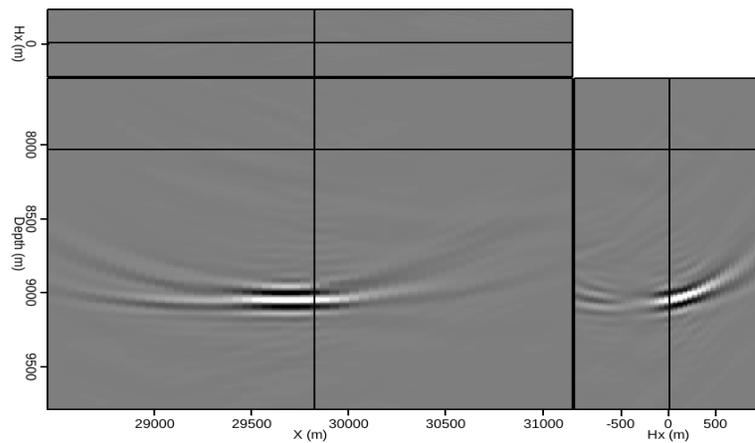
Figure 3.18: (a) A 2D field image produced using a velocity model 5% slower than the one provided, and (b) a manually-picked horizon of interest used to test three different velocity models. [CR] chap3/. 2d-slow,slow-picks



(a)



(b)



(c)

Figure 3.19: A single location from the reflector indicated in Figure 3.18(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% slower than the one provided; (b) the provided velocity model; and (c) a model 5% faster than the one provided. [CR] chap3/. slow-slow,slow-act,slow-fast

Migration model	F value
Provided velocity	0.550
Fast model	0.514
Slow model	0.518

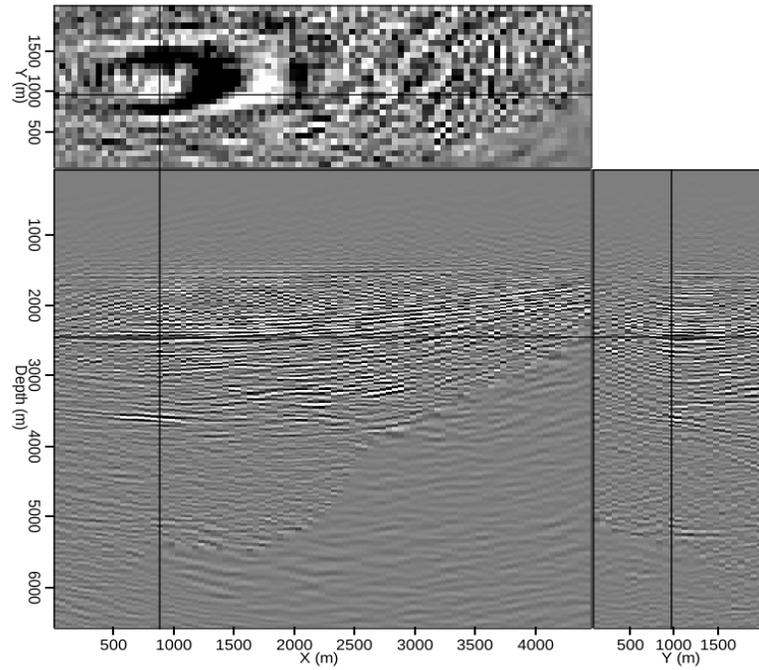
Table 3.5: Calculations from equation 3.5 for each 3D migration velocity model, after the initial image and synthesized wavefields were created using the velocity model provided with the data.

Migration model	F value
Provided velocity	0.481
Fast model	0.470
Slow model	0.466

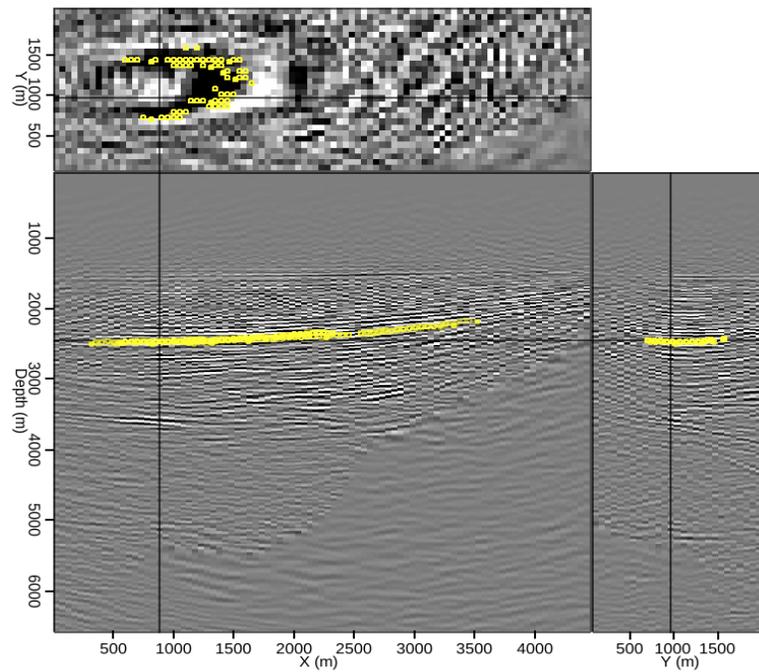
Table 3.6: Calculations from equation 3.5 for each 3D migration velocity model, after the initial image and synthesized wavefields were created using a velocity model 5% faster than the one provided with the data.

from the imaging of these wavefields are five-dimensional, the results are displayed as two separate figures. Figure 3.21 shows images extracted from the cube at zero-subsurface offset in both the x and y directions, after migrating the synthesized wavefields with the three velocity models used for the 2D examples. Figure 3.22 shows the corresponding images in subsurface offset coordinates, extracted at the indicated locations in Figure 3.21. In this example, the image generated using the provided velocity model (panel (b)) appears better focused in both domains. The 3D F -value calculations in table 3.5 provide more quantitative evidence.

The more realistic case of an initial image created using the faster velocity model also yielded encouraging results. While it is slightly more difficult to qualitatively distinguish the best-focused model either at zero-subsurface offset (Figure 3.24) or in the subsurface offset domain (Figure 3.25), the calculations in table 3.6 reveal that the provided model is indeed superior. This example demonstrates the value of a quantitative image focusing measure, especially for difficult-to-visualize 5D volumes.



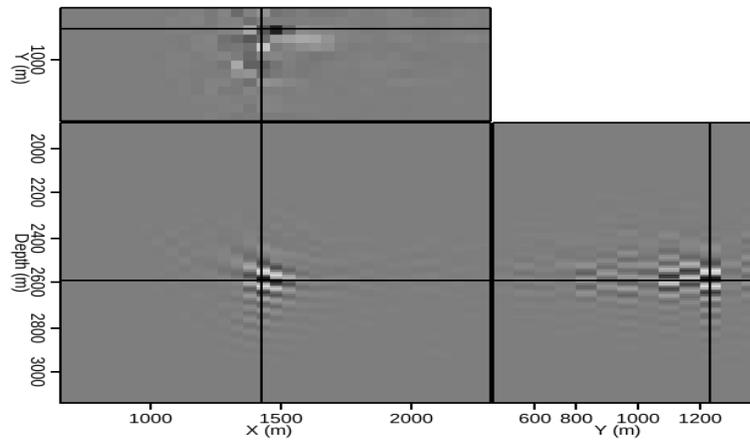
(a)



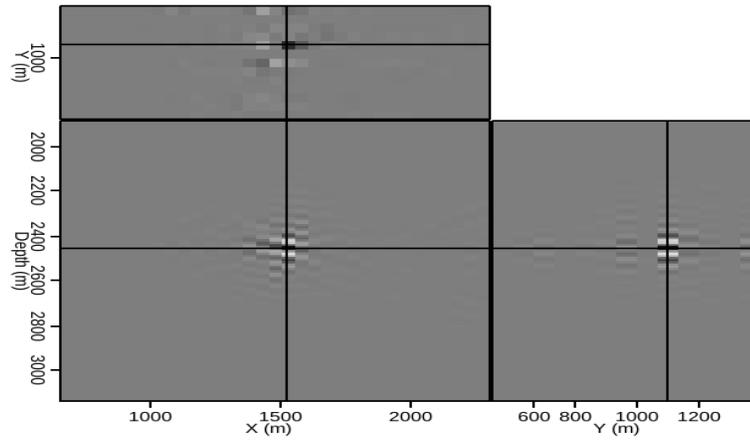
(b)

Figure 3.20: (a) A 3D field image produced using the provided velocity model, and (b) a manually-picked horizon of interest used to test three different velocity models.

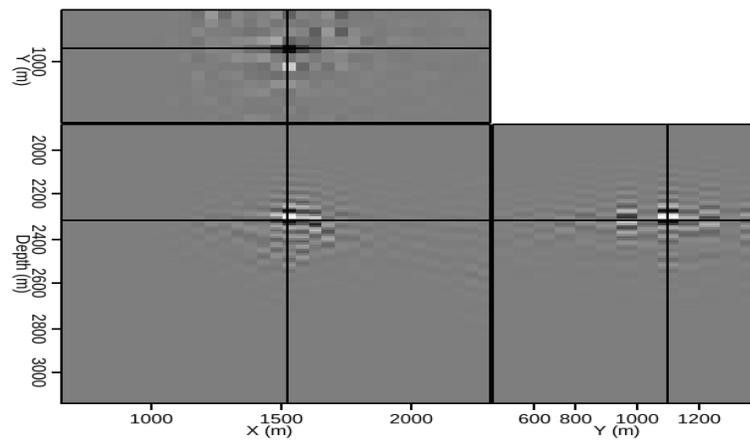
[CR] chap3/. 3d-field,3d-act-picks



(a)



(b)



(c)

Figure 3.21: A single location from the reflector indicated in Figure 3.16(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% faster than the one provided; (b) the provided velocity model; and (c) a model 5% slower than the one provided. [CR] chap3/. born-fast3d,born-act3d,born-slow3d

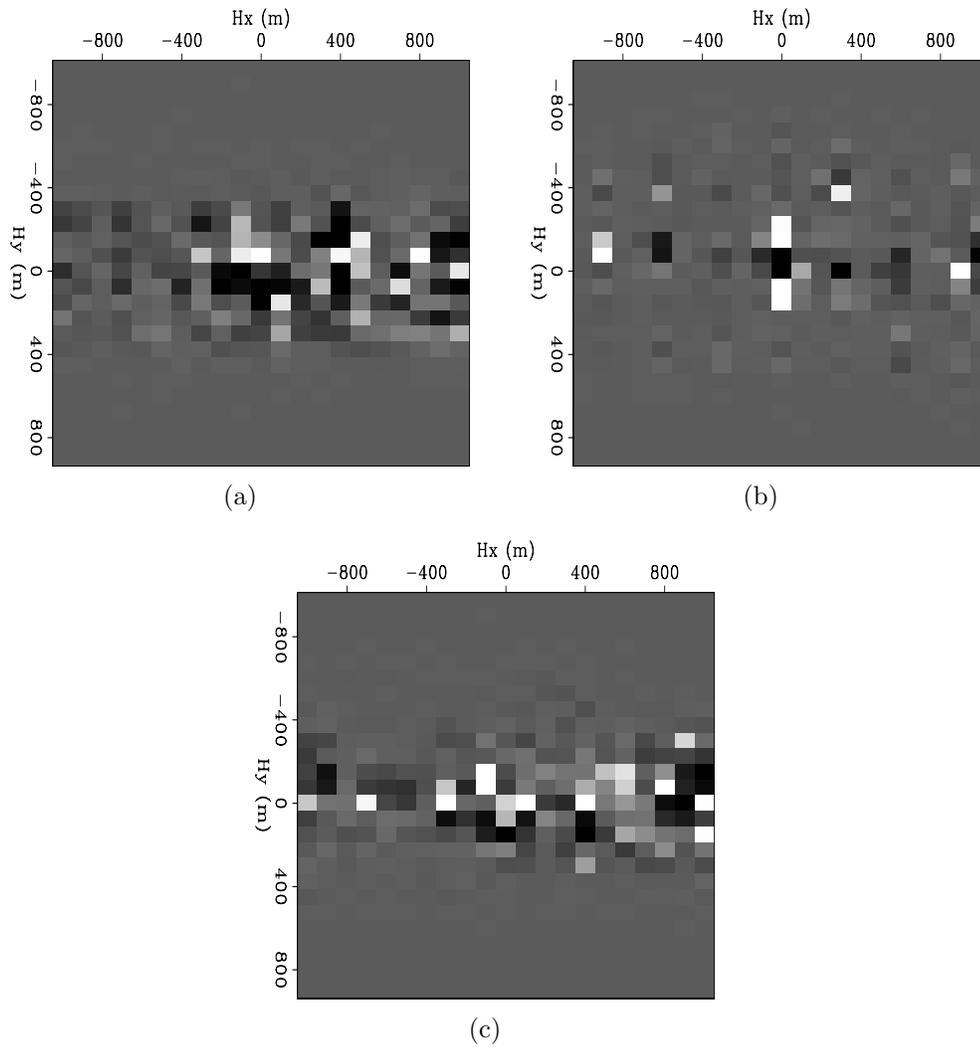
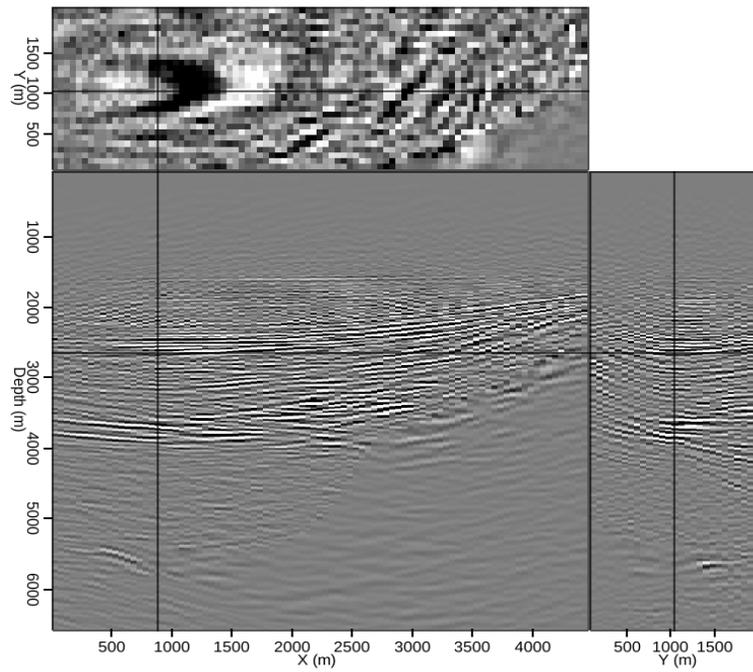
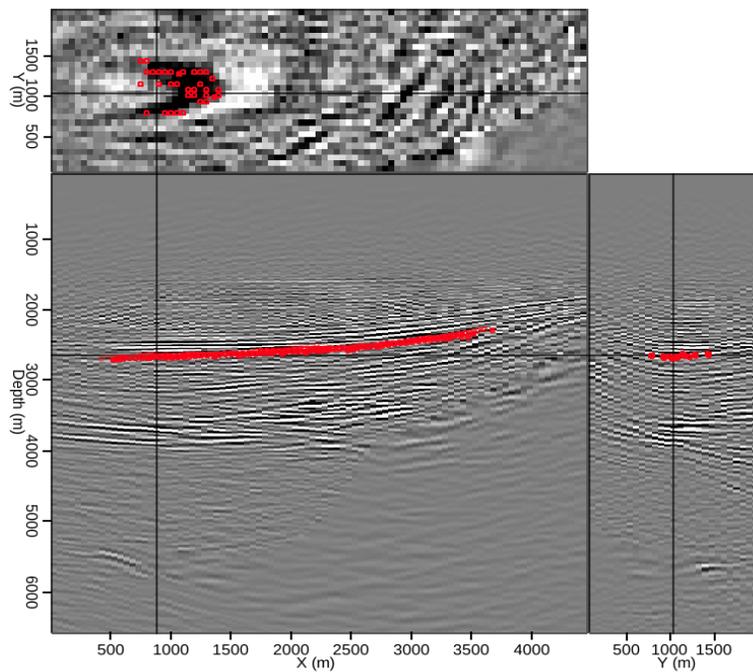


Figure 3.22: Subsurface offset images extracted from the indicated locations in Figure 3.21. The images were created using (a) the slower velocity; (b) the provided velocity; and (c) the faster velocity. [CR] chap3/. hxy-as,hxy-aa,hxy-af

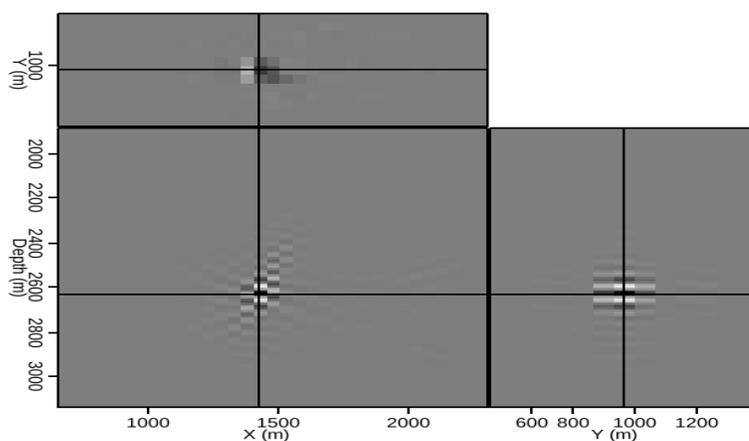


(a)

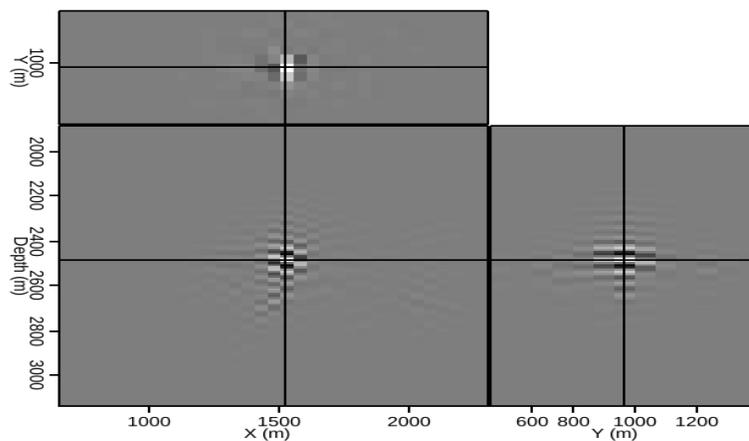


(b)

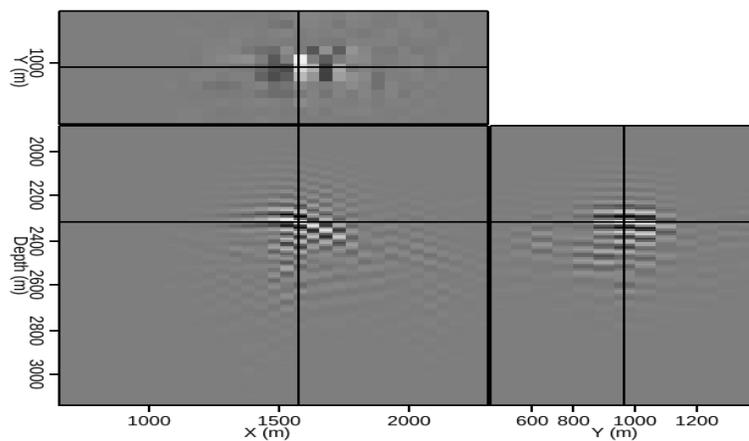
Figure 3.23: (a) A 3D field image produced using a velocity model 5% faster than the one provided, and (b) a manually-picked horizon of interest used to test three different velocity models. [CR] chap3/. 3d-fast,3d-fast-picks



(a)



(b)



(c)

Figure 3.24: A single location from the reflector indicated in Figure 3.23(b), imaged using synthesized source and receiver wavefields and (a) a velocity model 5% faster than the one provided; (b) the provided velocity model; and (c) a model 5% slower than the one provided. [CR] chap3/. fast-fast,fast-act,fast-slow

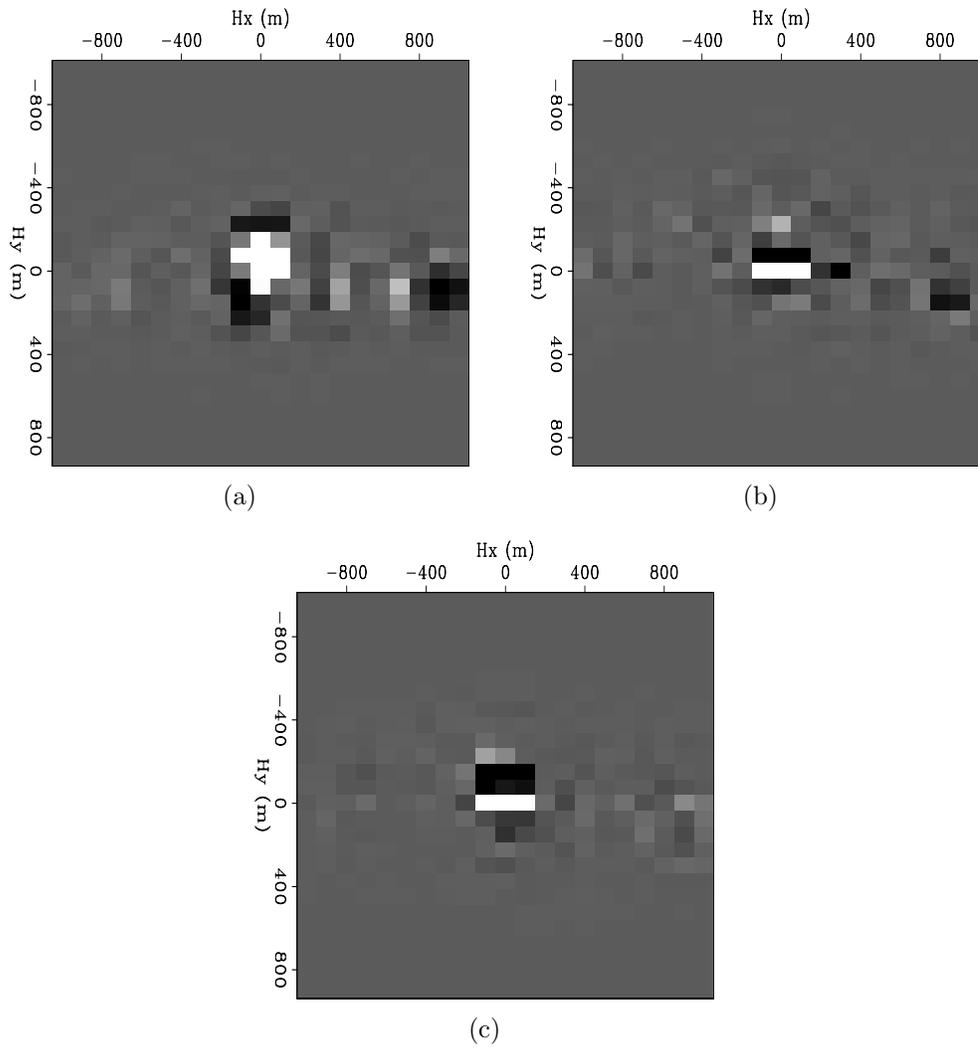


Figure 3.25: Subsurface offset images extracted from the indicated locations in Figure 3.24. The images were created using (a) the faster velocity; (b) the provided velocity; and (c) the slower velocity. [CR] `chap3/. hxy-ff,hxy-fa,hxy-fs`

CONCLUSIONS

This chapter introduced Born modeling techniques, modified to include a source function generated from an initial image, as a means to quickly evaluate multiple possible migration velocity models. By “exploding” subsurface offset gathers from an initial migrated image, we can generate an areal source function with information about the initial velocity model. More importantly, this source function is used to both generate Born-modeled data from isolated points in the subsurface, and to migrate that data to form an image. While crosstalk issues limit implementation of this method to single reflectors, multiple experiments performed on different image locations or reflectors may be used to overcome this limitation. Overall, this method can quickly and accurately reproduce useful velocity information (in the form of subsurface offset gathers) consistent with that obtained from a full migration of the original data. Furthermore, the method allows for re-datuming of wavefields prior to imaging, and can clearly distinguish between velocity models that differ only slightly. This method could form the basis for an efficient and interactive model-building tool, especially when paired with an image segmentation tool like the one presented in Chapter 2.

ACKNOWLEDGMENTS

I thank Schlumberger Multiclient for providing the field dataset used demonstrate the method, and SmaartJV for the Sigsbee synthetic model. I am also extremely grateful to Yaxun Tang for his guidance and prior work on the Born modeling and migration framework used extensively here.

Chapter 4

Integrated model building workflow

Recent increases in computing power have shifted model-building bottlenecks from computational tasks (such as imaging) and toward interpretation and similar human-intensive tasks. One approach to alleviate these bottlenecks is to develop computational interpretation tools, which can allow interpreters to take advantage of increased computational capabilities, while still allowing them to use their expertise to control the interpretation workflow. Two such tools, explored in previous chapters, are seismic image segmentation, and an efficient velocity model-evaluation method using synthesized wavefields. Here, I will use a 3D field data example from the Gulf of Mexico to demonstrate how these two tools can work together to effectively generate and test velocity models based on different salt scenarios. When an improved model is identified, re-migration with the new velocity model leads to an improved subsalt image.

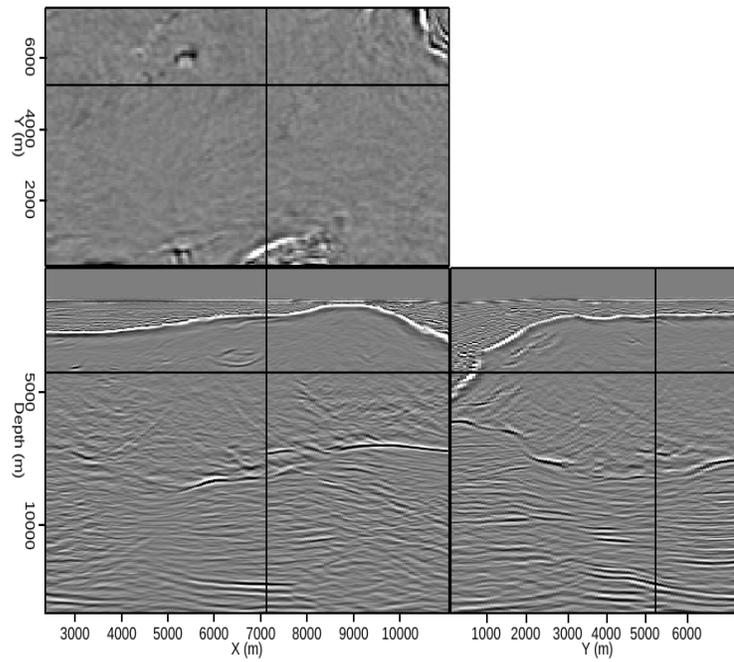
In the following sections, I will briefly review the methods used for both the image segmentation and model evaluation parts of the computational interpretation workflow. I will then demonstrate how these tools can be applied to a 3D example, using

a dataset from a wide-azimuth survey in the Gulf of Mexico, provided by Schlumberger Multiclient. The wide-azimuth nature of the survey should allow for sufficient illumination of subsalt areas to image subsalt reflectors, subject to the accuracy of the velocity model. An initial image generated using a velocity model provided with the data can be seen in Figure 4.1(a). Note that a fading of the reflectors directly beneath the salt body suggests possible errors in the velocity model provided with the data (4.1(b)). In particular, note that an inclusion within the salt body has not been assigned a velocity distinct from the rest of the salt, and that the interpretation of the base of salt is somewhat ambiguous. Both of these factors could contribute to the fading of the subsalt reflectors, and are addressed in the creation of the alternate velocity models, which are generated using image segmentation tools. Finally, I will compare the original model and the two alternate models using the synthesized wave-field methodology described in Chapter 3, and validate the comparison by showing a full re-migration of the data using the alternate model judged to be most accurate.

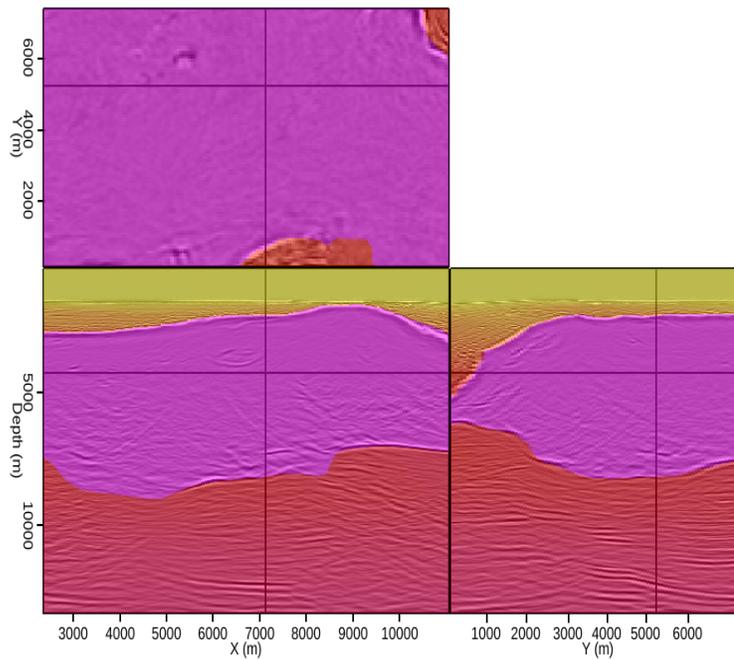
IMAGE SEGMENTATION

The Pairwise Region Comparison (PRC) image segmentation algorithm is a graph-cut technique based on the method of Felzenszwalb and Huttenlocher (2004). In Chapter 2, I described how this extremely efficient method can be adapted for use with seismic images. Recall that the goal of the example shown here is to improve continuity of the subsalt reflectors in Figure 4.1(a), which is an image obtained using one-way migration and the velocity model in Figure 4.1(b). Based on examination of the velocity model in Figure 4.1(b), two specific areas of possible improvement are the inclusion within the salt body, and the base-salt interpretation. Both of these areas can be addressed separately with image segmentation tools.

Figure 4.2(a) is a close-up image of the salt inclusion mentioned above. By isolating this smaller region for segmentation analysis, we are free to set the minimum segment size to a small number, allowing the automatic segmentation process to capture a higher degree of detail. An additional advantage of this strategy is that the



(a)



(b)

Figure 4.1: (a) A 3D image from the Gulf of Mexico (data courtesy of Schlumberger Multiclient) obtained via one-way migration with the velocity model shown in (b). A prominent sediment inclusion within the salt body, and/or a misinterpreted base of salt, may contribute to the subsalt reflectors' loss of continuity. [CR]

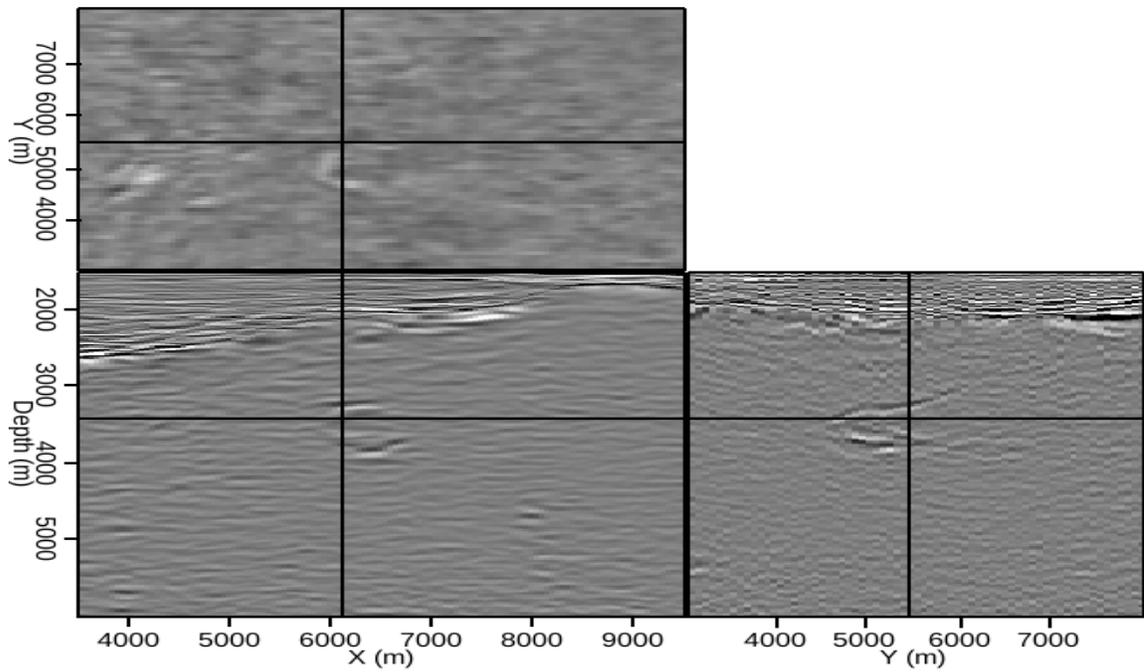
chap4/. img-init,vel-orig

limited domain allows for near-instantaneous segmentations, giving an interpreter the chance to experiment with parameters in an interactive fashion. Figure 4.2(b) is the automatic segmentation result for the salt inclusion. A new base-salt boundary can be defined by again isolating the base-salt region, and performing a detailed segmentation. By choosing which segments to include or exclude from the salt body, any number of possible boundaries can be defined. For this example, I created two different base-salt interpretations, one more aggressive in removing salt than the other. Based on the segmentations of both the salt inclusion and base salt, new velocity models were produced by assigning appropriate sediment velocities to the segmented regions which were originally salt. Figure 4.3 shows the original and two modified velocity models for this region. In this case, replacement velocities were taken at appropriate depths from the background sediment velocities in areas without salt.

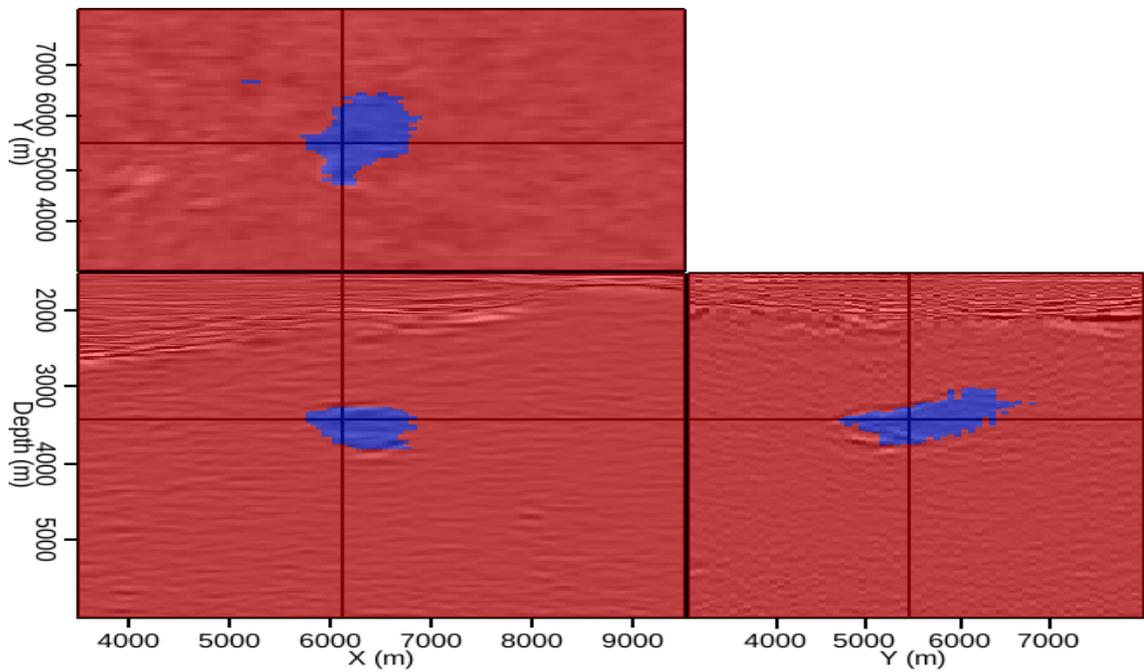
MODEL EVALUATION

Now, I will test the models created in the previous section using the efficient velocity model evaluation scheme described in Chapter 3. Recall that this method uses an initial image to generate a new areal source function, and then uses this source function to synthesize a new receiver wavefield via Born modeling, again using the initial image as a reflectivity model. Because this receiver wavefield is kinematically invariant of the velocity model used to create the initial image, we can then fairly (and efficiently) test any other models using the synthesized wavefields.

To test the models seen in Figure 4.3, I will investigate the effects of changing the model on a single reflector – in this case, the base salt reflector indicated in Figure 4.4. To do this, I performed several rounds of the evaluation procedure, in an attempt to build a clearer picture of the reflector than if only one or two locations were used in a single experiment. Following the strategy demonstrated in Figure 3.8(c), the image results from each experiment are summed into a final result. According to the procedure outlined above and described in detail in Chapter 3, new areal source and receiver wavefields are synthesized using the initial image and velocity model shown

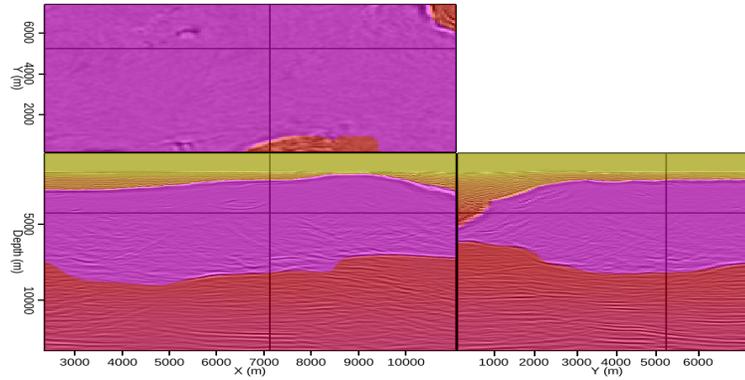


(a)

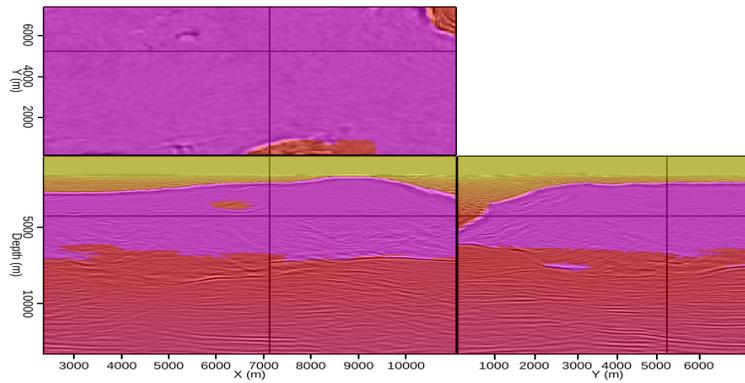


(b)

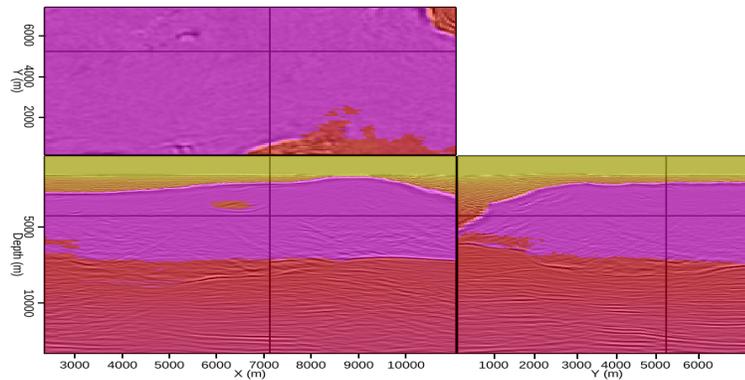
Figure 4.2: (a) Close-up view of the sediment inclusion first seen in Figure 4.1(a); (b) Interpreter-guided 3D segmentation of the image in (a). [CR] chap4/. sizoom-img,sizoom-seg



(a)



(b)



(c)

Figure 4.3: (a) The velocity model provided with the data; (b) An updated model based on the segmentation result in Figure 4.2(b) (and another defining an alternative base-salt interpretation). [CR] `chap4/. vzoom-orig1,vzoom-new1,vzoom-new2`

in Figure 4.1. Then, isolated locations from the picked reflector are imaged using the three different models in Figure 4.3. The results in Figure 4.5 are qualitatively similar; although differences are apparent due to the changing salt interpretation, it is difficult to make a judgment as to the models' relative accuracy simply using the images at zero subsurface offset. In this situation, information from the subsurface offset domain can be used to detect both qualitative and quantitative differences between the three models.

Qualitatively, we can examine panels displaying subsurface offset data in both the x and y directions, at specific x, y, z locations from the image. The white arrows in each panel of Figure 4.5 indicate equivalent locations along the base-salt reflector, at which the subsurface offset panels in Figure 4.6 were extracted. Figure 4.6(c), extracted from the image migrated with the more conservative alternate velocity model, clearly shows the highest degree of focusing near zero subsurface offset. Because it would be tedious to examine multiple locations along the reflector in this fashion, a quantitative measure of image focusing is desirable. Recall that when using the image focusing measure F from equation 3.5, a value of $F = 1$ means that all energy is perfectly focused at zero offset; as F decreases toward zero the image becomes progressively less focused. Table 4.1 displays the F value calculations corresponding to each of the images in Figure 4.5. In this case, the F value for the image obtained using the velocity model with the more conservative removal of salt was the highest. Thus, both the qualitative examination of the subsurface offset panels in Figure 4.6, and the quantitative calculations summarized in Table 4.1, agree that the more conservative alternate model yields the best-focused image.

RE-MIGRATION

To test the prediction of the model evaluation procedure, full migrations were performed using both the initial model and the alternate model identified in the previous section as the most accurate. One-way, split-step Fourier migration with interpolation

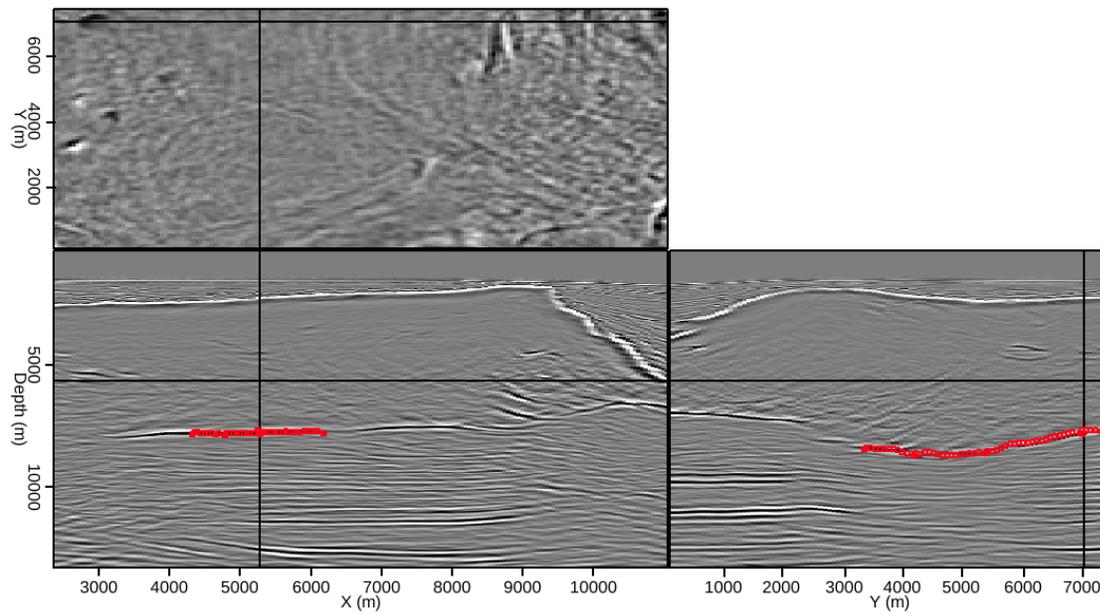


Figure 4.4: A manually-selected base-salt reflector that will be used to quickly evaluate the velocity models in Figure 4.3. [CR] chap4/.img-o2p

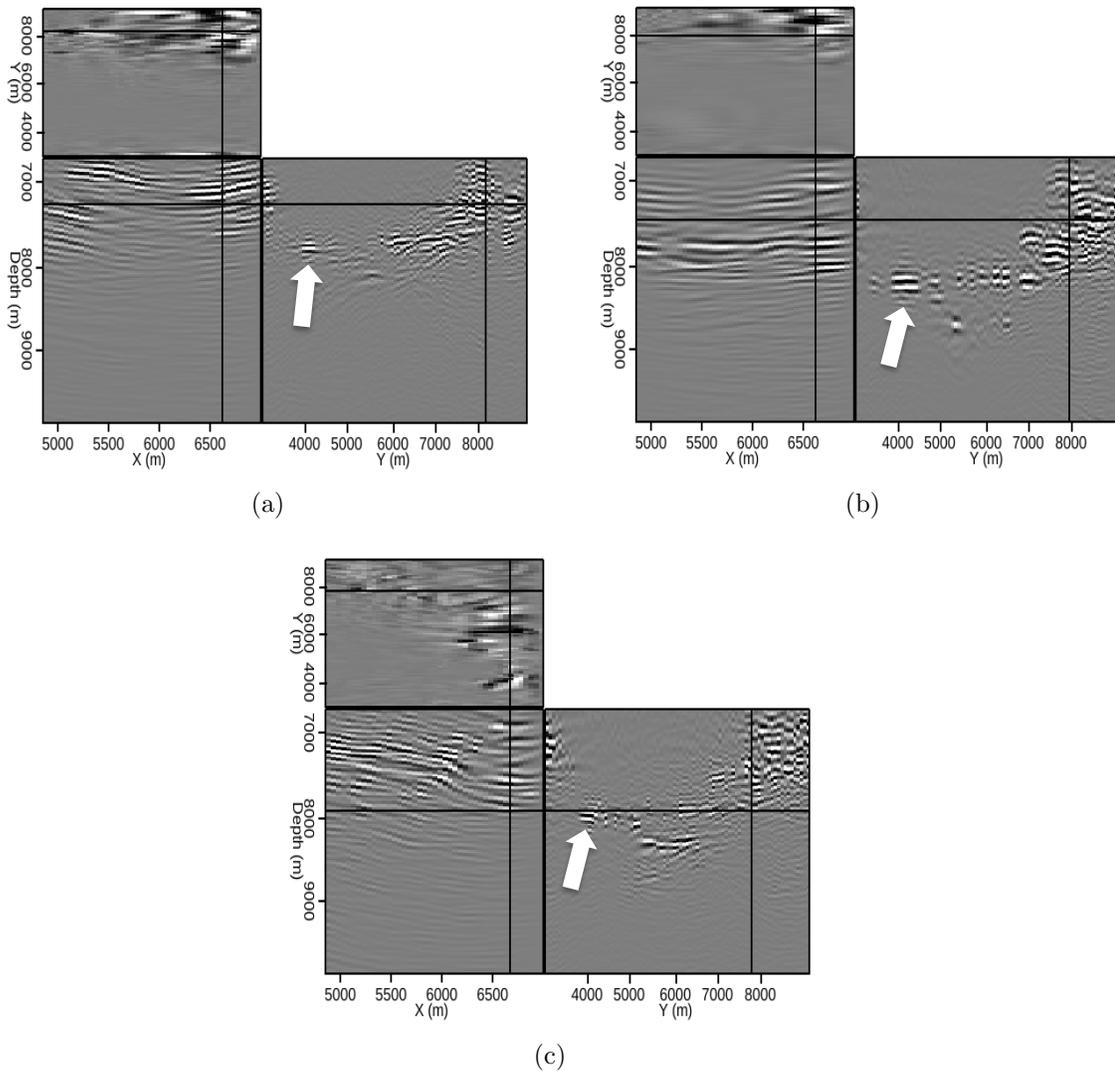


Figure 4.5: Results of performing multiple rounds of the model evaluation procedure on different locations along the reflector indicated in Figure 4.4, and summing the results. The velocity models used for the final imaging step correspond to those in Figure 4.3: (a) the original velocity model; (b) an alternate model with aggressive removal of salt; and (c) an alternate model with more conservative salt removal. The arrows indicate locations at which the subsurface offset panels in Figure 4.6 were extracted. [CR] chap4/. bsum-orig,bsum-v1,bsum-v2

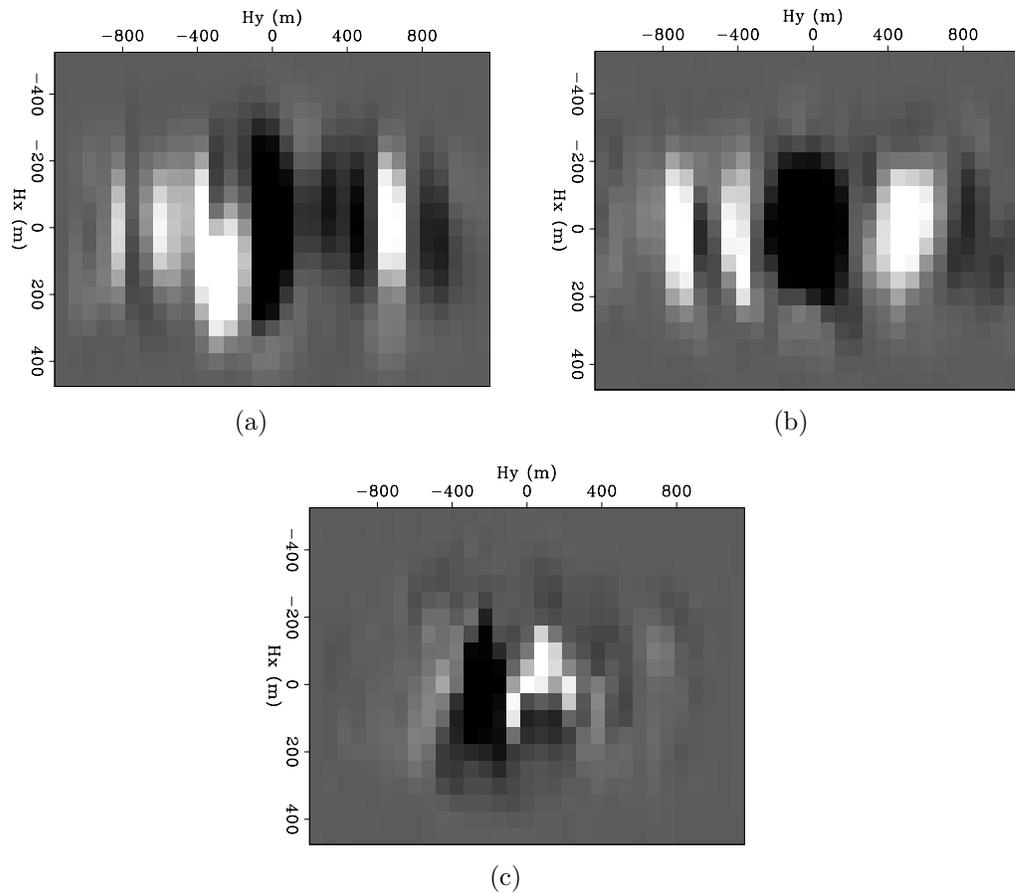


Figure 4.6: Subsurface offset panels at a single x, y, z location indicated by the arrows in Figure 4.5 for the image corresponding to (a) the original velocity model; and alternate models with (b) aggressive and (c) more conservative removal of salt. The greater degree of focusing near zero subsurface offset in (c) suggests a more accurate velocity model. [CR] chap4/. hxy-orig1,hxy-k4,hxy-k10

Migration model	F value
Initial model	0.785
Aggressive salt removal	0.791
Conservative salt removal	0.810

Table 4.1: Calculations from equation 3.5 for each migration velocity model in Figure 4.3, after the initial image and synthesized wavefields were created using the initial velocity model.

(Stoffa et al., 1990) was used, with five reference velocities selected via Lloyd’s algorithm (Lloyd, 1982; Clapp, 2004). The input data and parameters were identical for both migrations; the only difference was the velocity model. In order to compare the results, I will show both images at two different locations. Figures 4.7(a) and 4.7(b) show images at the first location, produced using the initial and new models, respectively. Figure 4.8 shows the same two images, with areas of particular improvement indicated with arrows and circles. At this location, both the base-salt reflector and the deepest subsalt reflectors show the greatest improvement with the new velocity model. Images from the second location are in Figure 4.9, with annotated versions of the same images seen in Figure 4.10. At this location, the new velocity model yields an image with improved continuity of subsalt reflectors, as well as a more accurate depiction of the salt inclusion itself.

CONCLUSIONS

Computational interpretation tools such as interpreter-guided image segmentation and efficient model evaluation using synthesized wavefields can effectively add automation to an interpreter-driven model building workflow. In this 3D field data example, image segmentation was used to delineate a salt body inclusion and define two versions of the base of salt different from that of the original model. To test new velocity models derived from these segmentations, several Born-modeled wavefields were synthesized and used to quickly image isolated locations from the base-salt reflector. When summed, the results of these experiments provided a more complete view of the reflector than would be possible using only sparse locations from a single experiment. Qualitative and quantitative analysis of the results suggested that a new model with conservative removal of salt would produce a better-focused image, and full migrations using both the initial model and this updated model confirmed that the new model produced improved continuity in both the base of salt and subsalt reflectors.

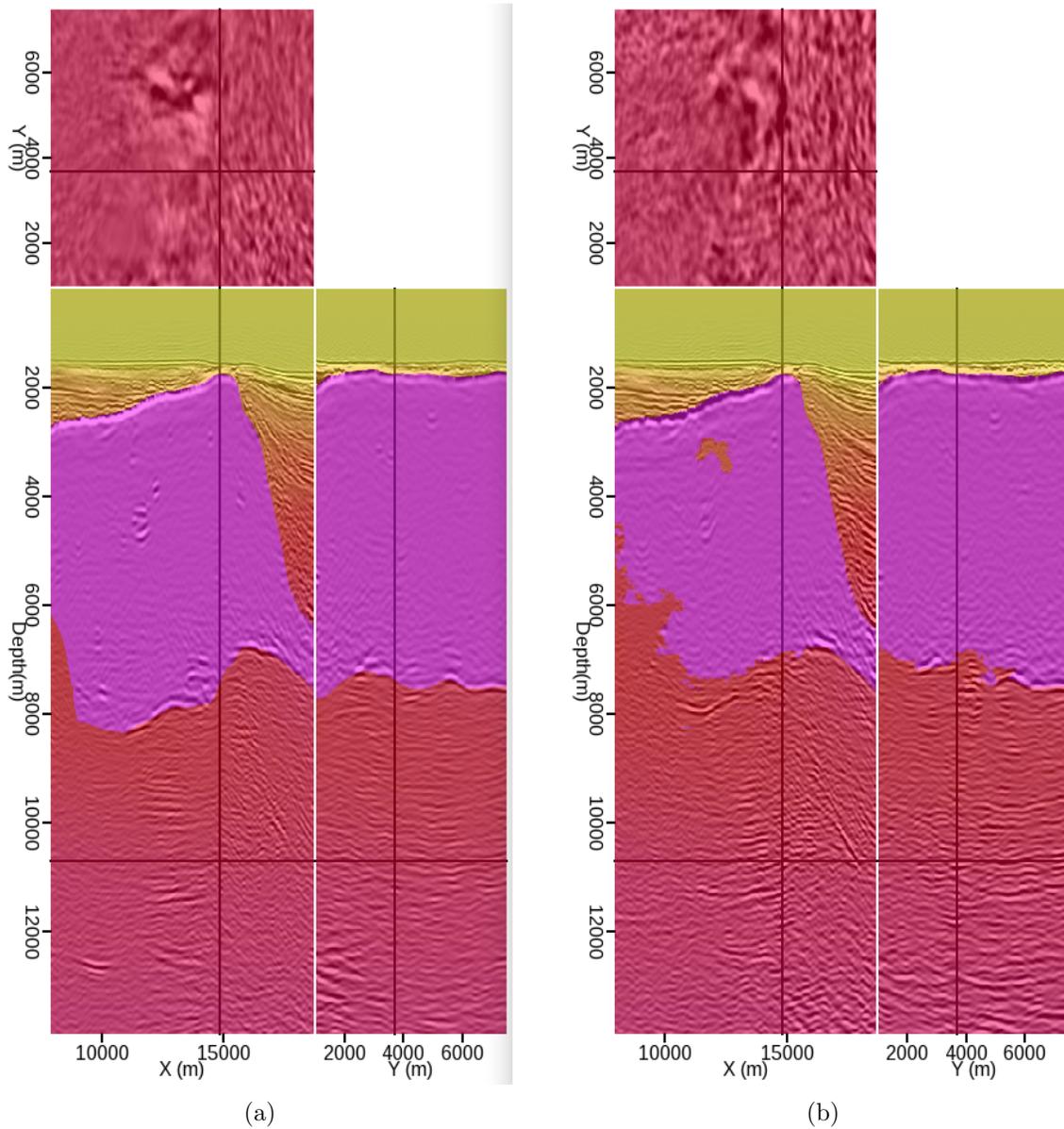


Figure 4.7: Full, one-way migrations with identical parameters using (a) the original velocity model, and (b) the updated model. [CR] `chap4/.imgv-yz0a,imgv-yz1a`

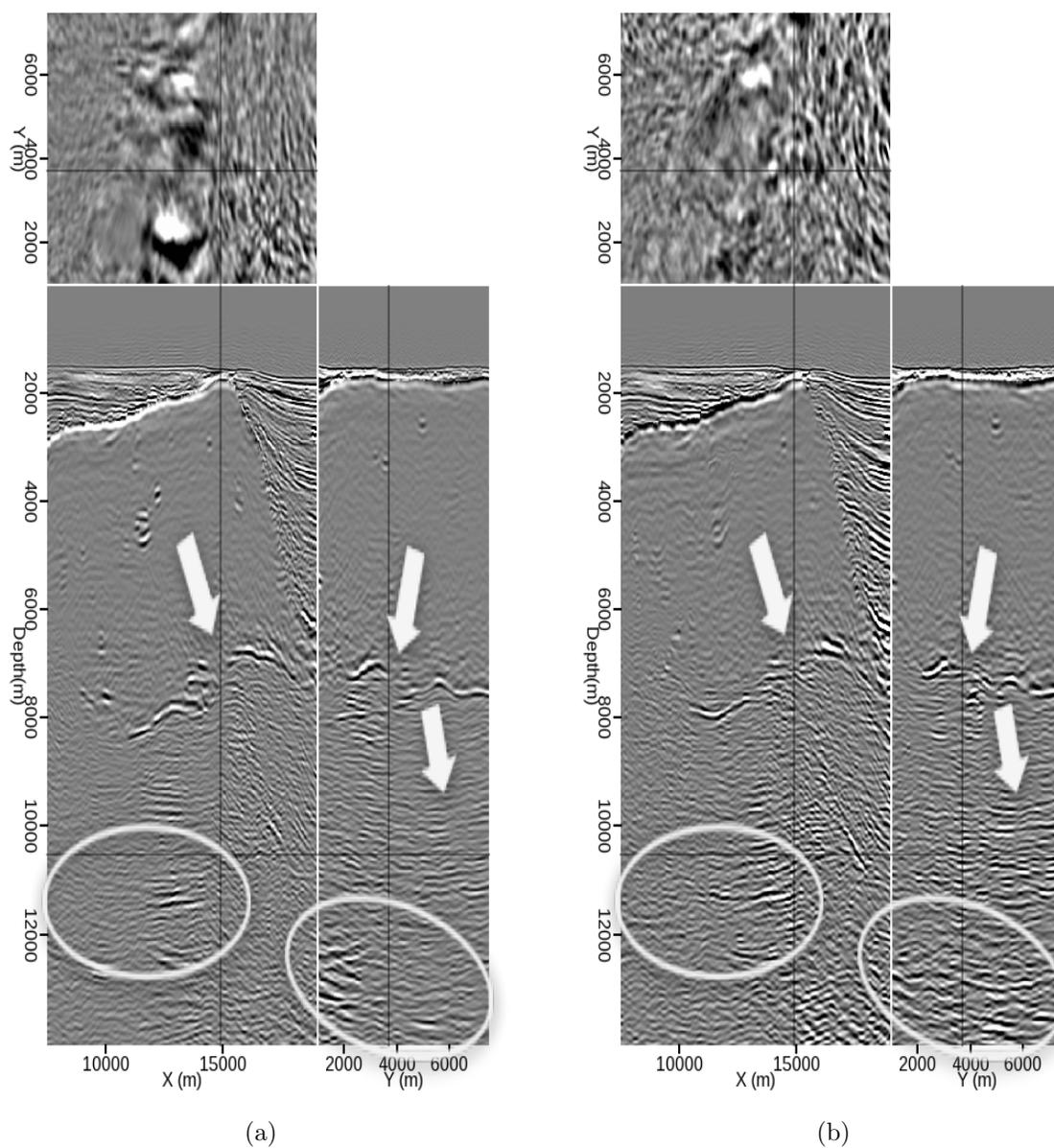


Figure 4.8: Same as Figure 4.7, but with areas of interest indicated. The deep subsalt reflectors and the base-salt reflector show particular improvement on the image generated with the new velocity model (b). [NR] chap4/. imga-yz0a,imga-yz1a

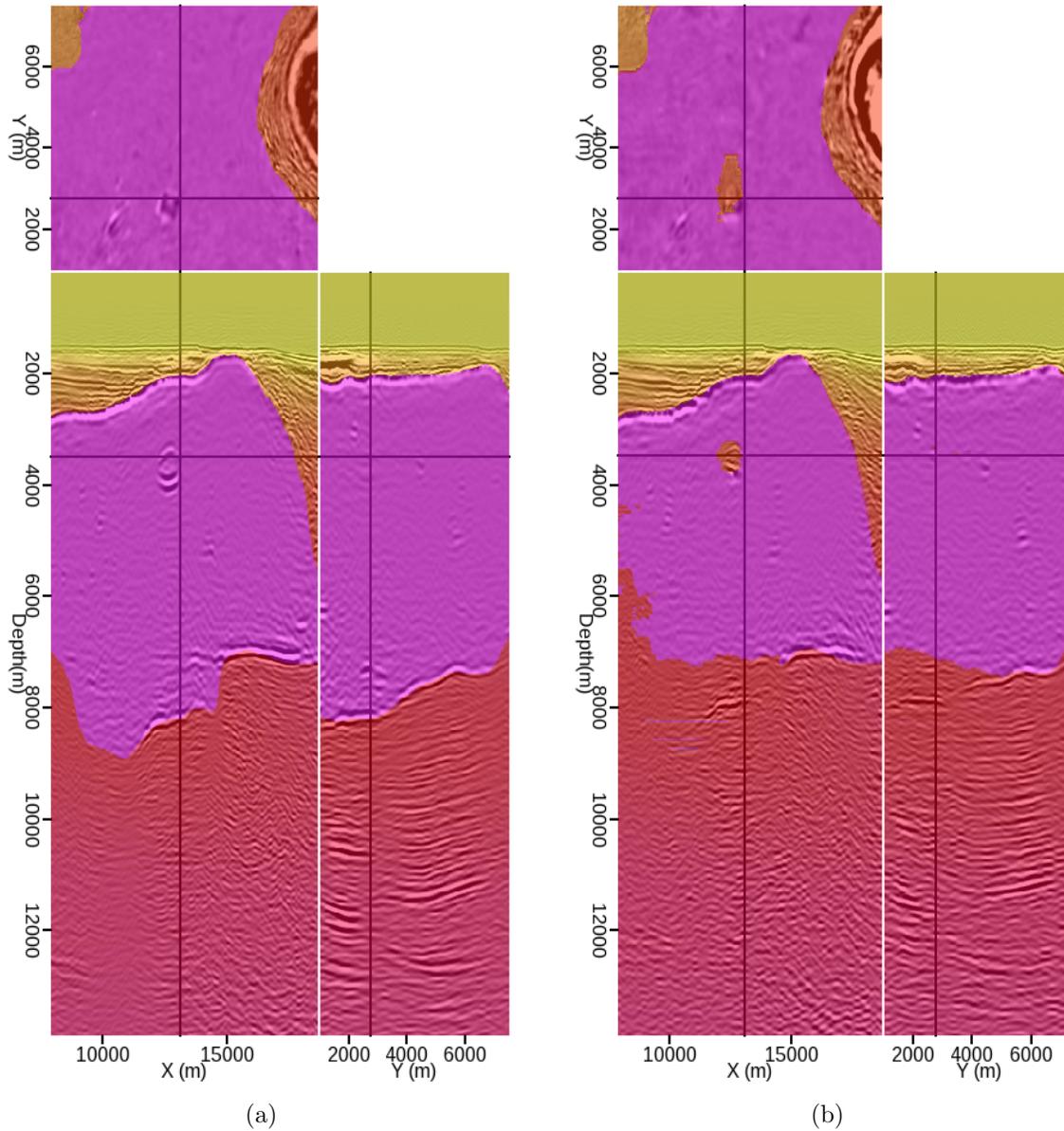


Figure 4.9: Images at a second location, obtained using (a) the original velocity model, and (b) the updated model. [CR] `chap4/.imgv-yz0b,imgv-yz1b`

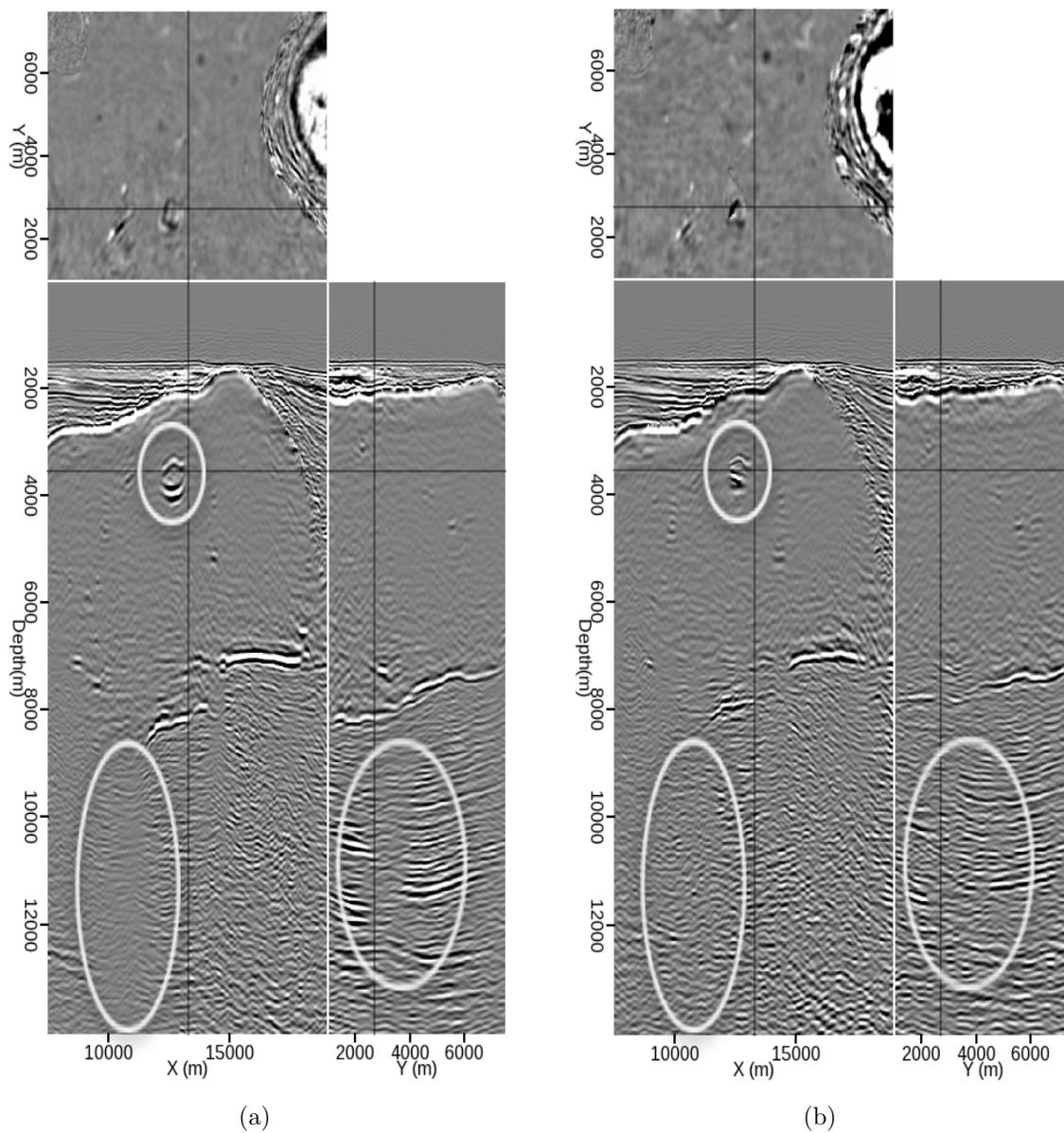


Figure 4.10: Same as Figure 4.9, but with areas of interest indicated. At this location, the subsalt reflectors show improved continuity and the salt inclusion is more accurately depicted on the image generated with the new velocity model (b). [NR] chap4/. imga-yz0b,imga-yz1b

ACKNOWLEDGMENTS

I thank Schlumberger Multiclient for providing the wide-azimuth dataset used in this chapter. In addition, I am grateful to Yang Zhang for his assistance with imaging the full dataset.

Bibliography

- Berthelot, A., A. H. S. Solberg, E. Morisbak, and L.-J. Gelius, 2012, 3D segmentation of salt using texture attributes: SEG Technical Program Expanded Abstracts 2012, 1–5.
- Biondi, B., 2005, 3-D seismic imaging: Stanford University.
- , 2006, Prestack exploding reflectors modeling for migration velocity analysis: SEG Technical Program Expanded Abstracts, 3056–3060.
- Chauris, H. and M. Benjema, 2010, Seismic wave-equation demigration/migration: GEOPHYSICS, **75**, S111–S119.
- Cheng, H., X. Jiang, Y. Sun, and J. Wang, 2001, Color image segmentation: advances and prospects: Pattern Recognition, **34**, 2259 – 2281.
- Claerbout, J., 2005, Basic earth imaging: Stanford University.
- Clapp, R., 2004, Reference velocity selection by a generalized lloyd method: SEG Technical Program Expanded Abstracts, 981–984.
- Dahlke, T., B. Biondi, and R. Clapp, 2014, Shape optimization using the FWI objective function for salt body segmentation : SEP-Report, **152**, 29–44.
- Felzenszwalb, P. F. and D. P. Huttenlocher, 2004, Efficient graph-based image segmentation: International Journal of Computer Vision, **59**, 167–181.
- Gao, F., P. Zhang, B. Wang, and V. Dirks, 2006, Fast beam migration a step toward interactive imaging: SEG Technical Program Expanded Abstracts, 2470–2474.
- Guerra, C., 2010, Migration-velocity analysis using image-space generalized wavefields: PhD thesis, Stanford University.
- Guerra, C. and B. Biondi, 2011, Fast 3D migration-velocity analysis by wavefield extrapolation using the prestack exploding-reflector model: GEOPHYSICS, **76**,

WB151–WB167.

Hale, D. and J. U. Emanuel, 2002, Atomic meshing of seismic images, *in* Expanded Abstracts, 2126–2129, SEG.

———, 2003, Seismic interpretation using global image segmentation, *in* Expanded Abstracts, 2410–2413, SEG.

Halpert, A., R. Clapp, and B. Biondi, 2009, Seismic image segmentation with multiple attributes: SEG Technical Program Expanded Abstracts, 3700–3704.

Haukas, J., O. R. Ravndal, B. Fotland, A. Bounaim, and L. Sonneland, 2013, Automated salt body extraction from seismic data using the level set method: First Break, **31**, 35 – 42.

Hill, N., 2001, Prestack gaussian beam depth migration: GEOPHYSICS, **66**, 1240–1250.

Hill, N. R., 1990, Gaussian beam migration: GEOPHYSICS, **55**, 1416–1428.

Hudec, M. R. and M. P. Jackson, 2007, Terra infirma: Understanding salt tectonics: Earth-Science Reviews, **82**, 1 – 28.

Lewis, W., B. Starr, and D. Vigh, 2012, A level set approach to salt geometry inversion in full-waveform inversion: SEG Technical Program Expanded Abstracts, 1–5.

Lloyd, S., 1982, Least squares quantization in pcm: Information Theory, IEEE Transactions on, **28**, 129–137.

Lomask, J., 2007, Seismic volumetric flattening and segmentation: PhD thesis, Stanford University.

Lomask, J., R. G. Clapp, and B. Biondi, 2007, Application of image segmentation to tracking 3D salt boundaries: GEOPHYSICS, **72**, P47–P56.

Mosher, C., E. Keskula, J. Malloy, R. Keys, H. Zhang, and S. Jin, 2007, Iterative imaging for subsalt interpretation and model building: The Leading Edge, **26**, 1424–1428.

Osher, S. and J. A. Sethian, 1988, Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations: Journal of Computational Physics, **79**, 12 – 49.

Pham, D. L., C. Xu, and J. L. Prince, 2000, Current methods in medical image segmentation: Annual Review of Biomedical Engineering, **2**, 315–337. (PMID:

- 11701515).
- Pilcher, R. S., B. Kilsdonk, and J. Trude, 2011, Primary basins and their boundaries in the deep-water northern gulf of mexico; origin, trap types, and petroleum system implications: *AAPG Bulletin*, **95**, 219–240.
- Romero, L. A., D. C. Ghiglia, C. C. Ober, and S. A. Morton, 2000, Phase encoding of shot records in prestack migration: *GEOPHYSICS*, **65**, 426–436.
- Shi, J. and J. Malik, 2000, Normalized cuts and image segmentation: *Institute of Electrical and Electronics Engineers Transactions on Pattern Analysis and Machine Intelligence*, **22**, 838–905.
- Stoffa, P., J. Fokkema, R. de Luna Freire, and W. Kessinger, 1990, Split-step fourier migration: *GEOPHYSICS*, **55**, 410–421.
- Stolt, R. H. and A. Benson, 1986, *Seismic migration: Theory and practice*: Geophysical Press.
- Taner, M. T., F. Koehler, and R. E. Sheriff, 1979, Complex seismic trace analysis: *GEOPHYSICS*, **44**, 1041–1063.
- Tang, Y., 2009, Target-oriented wave-equation least-squares migration/inversion with phase-encoded hessian: *GEOPHYSICS*, **74**, WCA95–WCA107.
- , 2011, Imaging and velocity analysis by target-oriented wavefield inversion: PhD thesis, Stanford University.
- Tang, Y. and B. Biondi, 2011, Target-oriented wavefield tomography using synthesized born data: *GEOPHYSICS*, **76**, WB191–WB207.
- Valet, L., G. Mauris, P. Bolon, and N. Keskes, 2001, Seismic image segmentation by fuzzy fusion of attributes: *IEEE Transactions on Instrumentation and Measurement*, **50**, 1014–1018.
- Wang, B., J. Ji, C. Mason, S. Gajawada, and Y. Kim, 2008, Beam-based interactive imaging for salt interpretation and salt model building: *SEG Technical Program Expanded Abstracts*, **27**, 3073–3077.
- Wang, B., C. Mason, K. Yoon, J. Ji, J. Cai, S. Suh, and Z. Li, 2011, Complex salt model building using a combination of interactive imaging and layer-stripping RTM: *First Break*, **29**, 47–54.
- Zahedi, F. and R. Thomas, 1993, A maximum homogeneity based median filter: *IEEE*

Colloquium on morphological and nonlinear image processing techniques, 7/1 – 7/5.
Zahn, C. T., 1971, Graph-theoretical methods for detecting and describing Gestalt clusters: IEEE Transactions on Computers, **C-20**, 68–86.