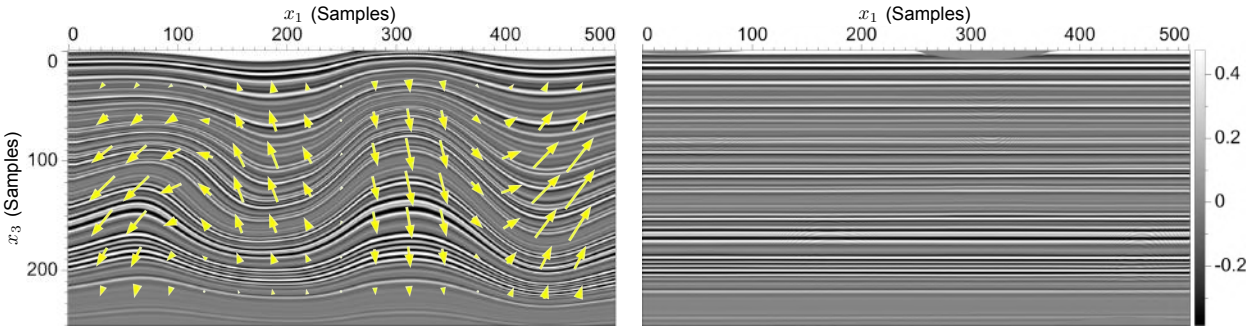
Non-vertical deformations for seismic image flattening

Simon Luo & Dave Hale
Center for Wave Phenomena, Colorado School of Mines, Golden, CO 80401, USA



(a) (b)

Figure 1. A synthetic seismic image (a) flattened (b) with a non-vertical shift vector field represented by the yellow arrows.

ABSTRACT

Seismic image flattening produces subsurface images in which sedimentary layering is horizontal. With flattened images, interpretation of stratigraphic features is straightforward, and horizon picking is trivial.

Most flattening methods are limited to vertical shearing and stretching of an image. Because of this limitation, these methods may have difficulty flattening seismic images that contain non-vertical deformations without significantly distorting image features. We propose a new image flattening method that uses a vector shift field, instead of a scalar field of vertical shifts, to represent deformation in an image. The method can flatten by vertically shearing or by rotating portions of an image, or by a combination of vertical shear and rotation. Because it is not limited to vertical shearing, the method can flatten in ways more consistent with geologic deformation.

Key words: flattening inversion structure tensor

1 INTRODUCTION

To interpret stratigraphic features in a seismic image, it is helpful to view an *isochron* image—an image of constant geologic time. However, because of structural deformation, the axes of seismic images are rarely aligned with geologic time. Thus, it is necessary to identify isochrons. This can be accomplished by manual picking of horizons, or alternatively, by automatic seismic image flattening.

Most methods for automatic flattening are limited to vertical shearing of images (Lomask et al., 2006; Parks, 2010). These methods can flatten well images in which the geologic deformation itself contains only vertical shifts, but for images with more complicated deformations, flattening by vertical shearing may significantly distort image features. For example, non-vertical shifts are necessary to flatten horizons that are faulted, folded, or overturned. A flattening method should be consistent with geologic deformation; but methods that are limited to vertical shearing of an image clearly cannot represent all manner of such deformation.

We propose a new method for automatic flattening of seismic images that represents deformation in an image using a shift vector field instead of a scalar field of vertical shifts, and then flattens the image by reversing the deformation. Figure 1 shows an example of flattening using non-vertical shifts. Notice that the shift vec-

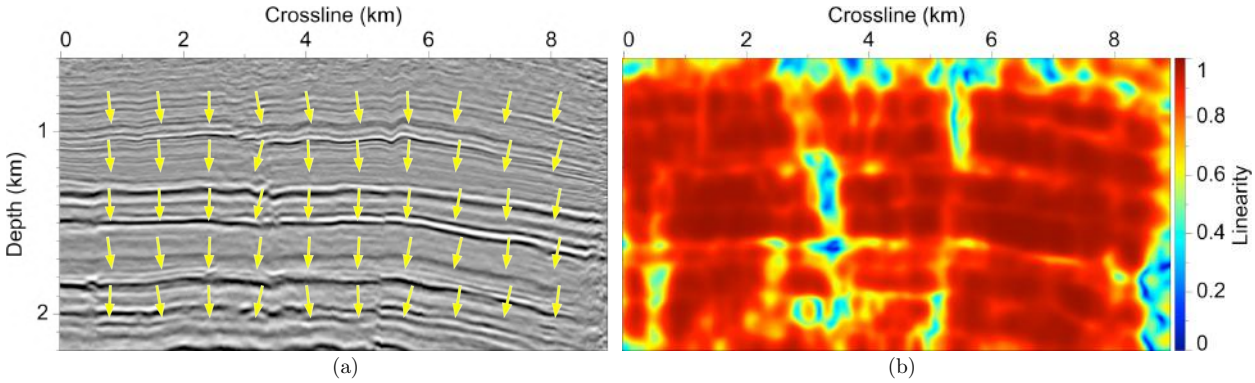


Figure 2. Normal vectors (a) and linearity (b) computed for a 2D seismic image.

tor field, depicted by the yellow arrows in Figure 1a, shifts image samples both vertically and horizontally. Because the method is not limited to vertical shearing, it can better represent actual geologic deformation, and can thereby minimize non-geologic distortions of image features.

1.1 Structure tensors

To flatten image features, we need a measure of their orientation. We use the *structure tensor* (van Vliet and Verbeek, 1995; Fehmers and Höcker, 2003) to compute normal vectors perpendicular to locally linear features in 2D images or locally planar features in 3D images.

The structure tensor, also called the gradient-squared tensor, is a smoothed outer product of image gradients. In 2D, the structure tensor $\mathbf{T}$ for a single image sample is a 2×2 symmetric positive-semidefinite matrix:

$$\mathbf{T} = \begin{bmatrix} t_{11} & t_{13} \\ t_{13} & t_{33} \end{bmatrix}, \tag{1}$$

where t_{ij} is a smoothed product of image derivatives in the x_i and x_j directions.

The eigendecomposition of $\mathbf{T}$ describes the orientation of features in an image (Fehmers and Höcker, 2003). For a 2D image, the eigendecomposition is

$$\mathbf{T} = \lambda_u \mathbf{u}\mathbf{u}^\mathsf{T} + \lambda_v \mathbf{v}\mathbf{v}^\mathsf{T}, \tag{2}$$

where λ_u and λ_v are the eigenvalues corresponding to eigenvectors $\mathbf{u}$ and $\mathbf{v}$, respectively. By convention, $\lambda_u \geq \lambda_v \geq 0$.

The eigenvector $\mathbf{u}$ corresponding to the largest eigenvalue describes the direction of the highest image derivative, and therefore is orthogonal to linear features in an image. In other words, eigenvector $\mathbf{u}$ is the normal vector. Figure 2a shows a subset of the normal vectors computed for a 2D seismic image.

The eigenvalues λ_u and λ_v provide a measure of the linearity of image features. The linearity λ_1 is computed as

$$\lambda_1 = (\lambda_u - \lambda_v)/\lambda_u. \tag{3}$$

Note that $0 \leq \lambda_1 \leq 1$. For coherent, linear features ($\lambda_u \gg \lambda_v$) linearity approaches unity, whereas for incoherent features ($\lambda_u = \lambda_v$) linearity approaches zero. Figure 2b shows linearity computed for the seismic image shown in Figure 2a. Notice that in noisy areas (e.g., the lower right corner of the image) and in areas where features are less coherent (e.g., near faults), linearity is closer to zero. When flattening, we use linearity to give more weight to coherent, linear image features.

In 3D, we compute image derivatives in three orthogonal directions. Accordingly, each structure tensor is a 3×3 symmetric positive-semidefinite matrix:

$$\mathbf{T} = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{12} & t_{22} & t_{23} \\ t_{13} & t_{23} & t_{33} \end{bmatrix}, \tag{4}$$

with eigendecomposition

$$\mathbf{T} = \lambda_u \mathbf{u}\mathbf{u}^\mathsf{T} + \lambda_v \mathbf{v}\mathbf{v}^\mathsf{T} + \lambda_w \mathbf{w}\mathbf{w}^\mathsf{T}. \tag{5}$$

For 3D image flattening, we are interested in planar, rather than linear, features. Planarity is given by

$$\lambda_2 = (\lambda_u - \lambda_v)/\lambda_u. \tag{6}$$

As for linearity in 2D, we use planarity to give more weight to coherent, planar image features when flattening in 3D.

2 IMAGE FLATTENING

Let $f(\mathbf{x})$ denote an input image and let $g(\mathbf{u})$ denote the flattened image, where $\mathbf{x}$ is a point in present-day coordinates and $\mathbf{u}$ a point in the flattened coordinates. In 2D, $\mathbf{x} = (x_1, x_3)$ and $\mathbf{u} = (u_1, u_3)$; in 3D, $\mathbf{x} = (x_1, x_2, x_3)$ and $\mathbf{u} = (u_1, u_2, u_3)$. Subscripts 1 and 2 denote horizontal axes, while subscript 3 denotes the vertical depth axis.

To flatten an image, we need a mapping $\mathbf{x}(\mathbf{u})$ that

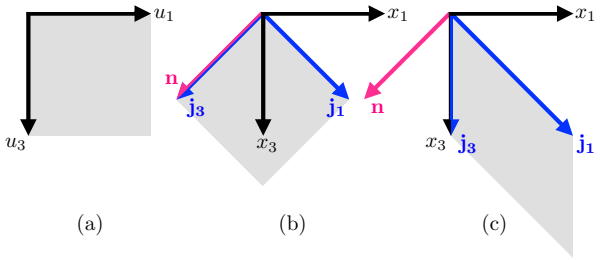


Figure 3. A unit area (a) in flattened space maps to a rotated unit area (b) or a sheared unit area (c) in present-day space.

specifies the present-day location $\mathbf{x}$ of any point $\mathbf{u}$ in the flattened space. Given $\mathbf{x}(\mathbf{u})$, we may compute the flattened image by

$$g(\mathbf{u}) = f(\mathbf{x}(\mathbf{u})). \tag{7}$$

In the flattened space, constant u_3 corresponds to constant geologic time. Thus, the corresponding surface $\mathbf{x}(\mathbf{u})$ for constant u_3 is a geologic horizon.

Consider a point $\mathbf{x} = \mathbf{x}(\mathbf{u})$ located on a geologic horizon, located in the infinitesimal neighborhood of a point $\mathbf{x}_0 = \mathbf{x}(\mathbf{u_0})$ on the same horizon. The first-order Taylor series approximation for $\mathbf{x}$ is

$$\mathbf{x} = \mathbf{x}_0 + \mathbf{J}\,(\mathbf{u} - \mathbf{u}_0), \tag{8}$$

where $\mathbf{J} = \partial\mathbf{x}/\partial\mathbf{u}$ is the Jacobian of the transformation $\mathbf{x}(\mathbf{u})$. Let $\mathbf{n}$ denote the unit normal vector at $\mathbf{x}_0$, and let $\mathbf{m}$ denote the corresponding normal vector at $\mathbf{u}_0$. Then, because

$$\mathbf{n}^\mathsf{T}(\mathbf{x} - \mathbf{x}_0) = \mathbf{n}^\mathsf{T}\mathbf{J}\,(\mathbf{u} - \mathbf{u}_0) = \mathbf{m}^\mathsf{T}(\mathbf{u} - \mathbf{u}_0) = 0, \tag{9}$$

we have

$$\mathbf{J}^\mathsf{T}\mathbf{n} = \mathbf{m}. \tag{10}$$

Because the normal vector in the flattened space must point downward for the image to be flat, all components of $\mathbf{m}$ except for m_3 must be zero. Note that we compute normal vectors $\mathbf{n}$ from the input image, so that $\mathbf{n} = \mathbf{n}(\mathbf{x})$, not $\mathbf{n}(\mathbf{u})$.

2.1 Flattening in 2D

The mapping $\mathbf{x}(\mathbf{u})$ may be written in terms of a shift vector field $\mathbf{r}(\mathbf{u})$:

$$\mathbf{x}(\mathbf{u}) = \mathbf{u} - \mathbf{r}(\mathbf{u}). \tag{11}$$

The Jacobian for this mapping in 2D is

$$\mathbf{J} = \begin{bmatrix} \partial x_1/\partial u_1 & \partial x_1/\partial u_3 \\ \partial x_3/\partial u_1 & \partial x_3/\partial u_3 \end{bmatrix}, \tag{12}$$

which, in terms of the shift field $\mathbf{r}(\mathbf{u})$, is

$$\mathbf{J} = \begin{bmatrix} 1 - r_{11} & -r_{13} \\ -r_{31} & 1 - r_{33} \end{bmatrix}, \tag{13}$$

where $r_{ij} \equiv \partial r_i/\partial u_j$. From equation 10, we have

$$n_1(1 - r_{11}) - n_3 r_{31} = 0 \tag{14}$$

$$n_3(1 - r_{33}) - n_1 r_{13} = m_3 \tag{15}$$

where m_3 is the vertical component of the normal vector after flattening, and is related to the type of deformation in the shift vector field $\mathbf{r}(\mathbf{u})$. Equation 14 can be considered the flattening equation—if the shift vector field $\mathbf{r}(\mathbf{u})$ satisfies this equation for n_1 and n_3 computed from an image, then applying the shifts $\mathbf{r}(\mathbf{u})$ will flatten the image. We write equation 13 as

$$\mathbf{J} = \mathbf{I} - \mathbf{R}, \tag{16}$$

where $\mathbf{I}$ is a 2×2 identity matrix, and $\mathbf{R}$ is a matrix of partial derivatives of the shift vector field $\mathbf{r}(\mathbf{u})$:

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{13} \\ r_{31} & r_{33} \end{bmatrix}. \tag{17}$$

Solving equation 16 for $\mathbf{R}$ gives

$$\mathbf{R} = \mathbf{I} - \mathbf{J}. \tag{18}$$

To find $\mathbf{R}$, we need an appropriate Jacobian that satisfies equation 10.

We consider separately two different methods: flattening by vertical shearing, and flattening by rotation. The Jacobian for vertical shearing in 2D is

$$\mathbf{J}_v = \begin{bmatrix} 1 & 0 \\ -n_1/n_3 & 1 \end{bmatrix}, \tag{19}$$

where n_1 and n_3 are the components of the unit normal vector $\mathbf{n}$, and the subscript v indicates vertical shearing. For rotation, the Jacobian is

$$\mathbf{J}_r = \begin{bmatrix} n_3 & n_1 \\ -n_1 & n_3 \end{bmatrix}, \tag{20}$$

where the subscript r indicates rotation. Figure 3 shows the mapping produced by these Jacobian matrices. In the figure, $\mathbf{n}$ is the normal vector in present-day space, and $\mathbf{j_1}$ and $\mathbf{j_3}$ are, respectively, the first and second column of the Jacobian. Figure 3b shows the rotation Jacobian $\mathbf{J}_r$, while Figure 3c shows the vertical shear Jacobian $\mathbf{J}_v$. Both rotation and vertical shear are area-preserving transformations; thus, a unit area in flattened space maps to either a rotated or sheared unit area in present-day space. Note that thickness measured in the direction of the normal vector is preserved only in the case of rotation.

From equation 18 with $\mathbf{J} = \mathbf{J}_v$, we obtain the matrix of partial derivatives of the shift vector field for vertical shearing:

$$\mathbf{R}_v = \begin{bmatrix} 0 & 0 \\ n_1/n_3 & 0 \end{bmatrix}. \tag{21}$$

Similarly, equation 18 with $\mathbf{J} = \mathbf{J}_r$ gives the partial derivative matrix for rotation:

$$\mathbf{R}_r = \begin{bmatrix} 1 - n_3 & -n_1 \\ n_1 & 1 - n_3 \end{bmatrix}. \tag{22}$$

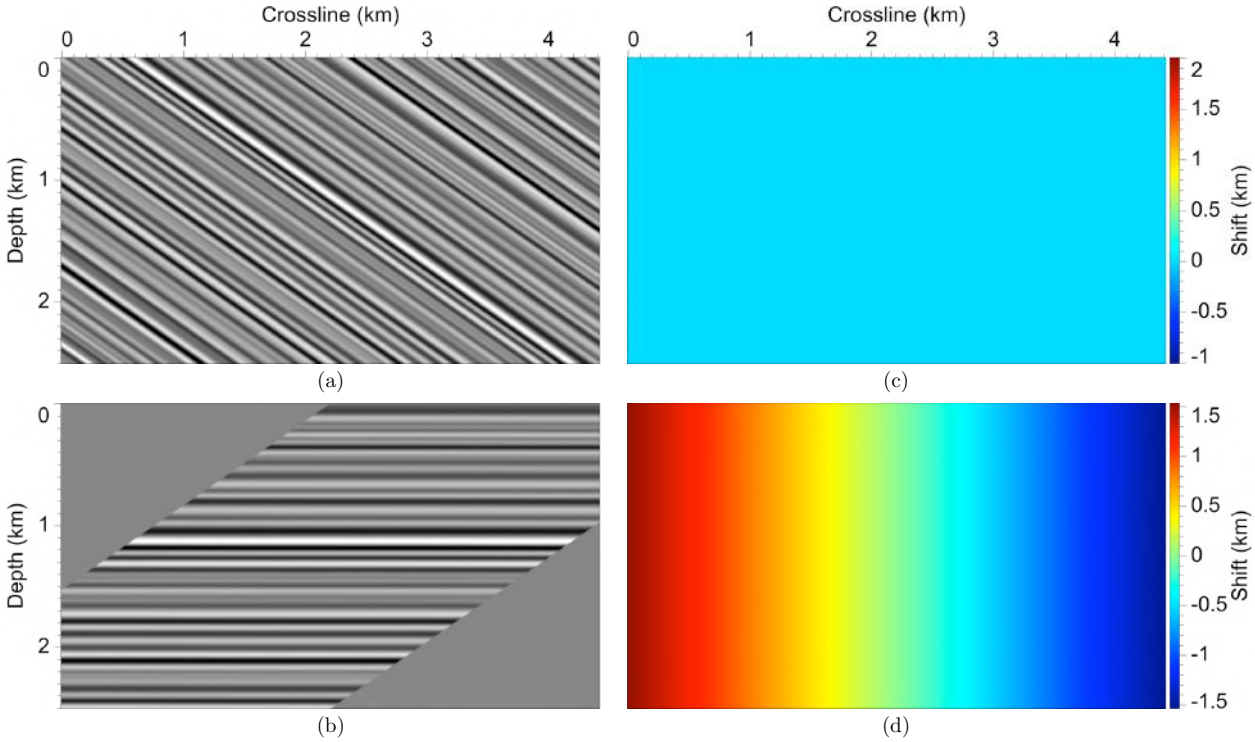


Figure 4. A synthetic image (a) flattened by vertical shear (b) with the horizontal (c) and vertical (d) components of the shift vector field $\mathbf{r}(\mathbf{u})$.

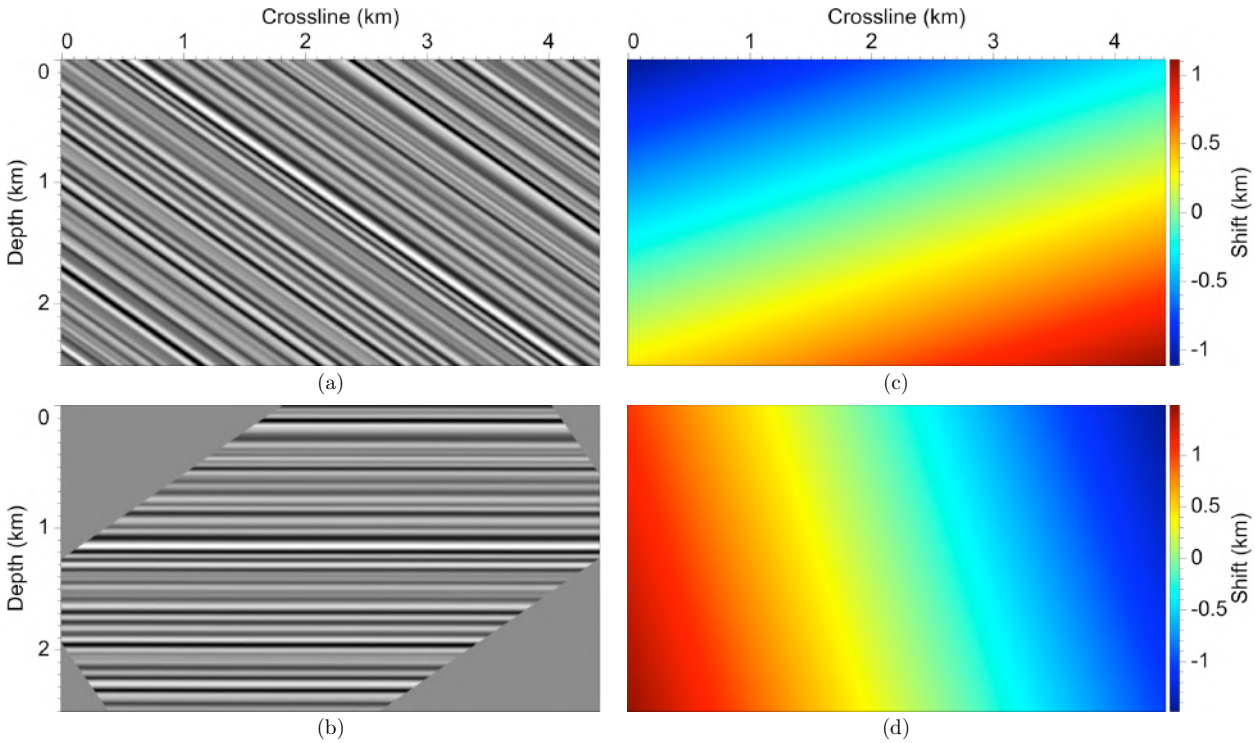


Figure 5. A synthetic image (a) flattened by rotation (b) with the horizontal (c) and vertical (d) components of the shift vector field $\mathbf{r}(\mathbf{u})$.

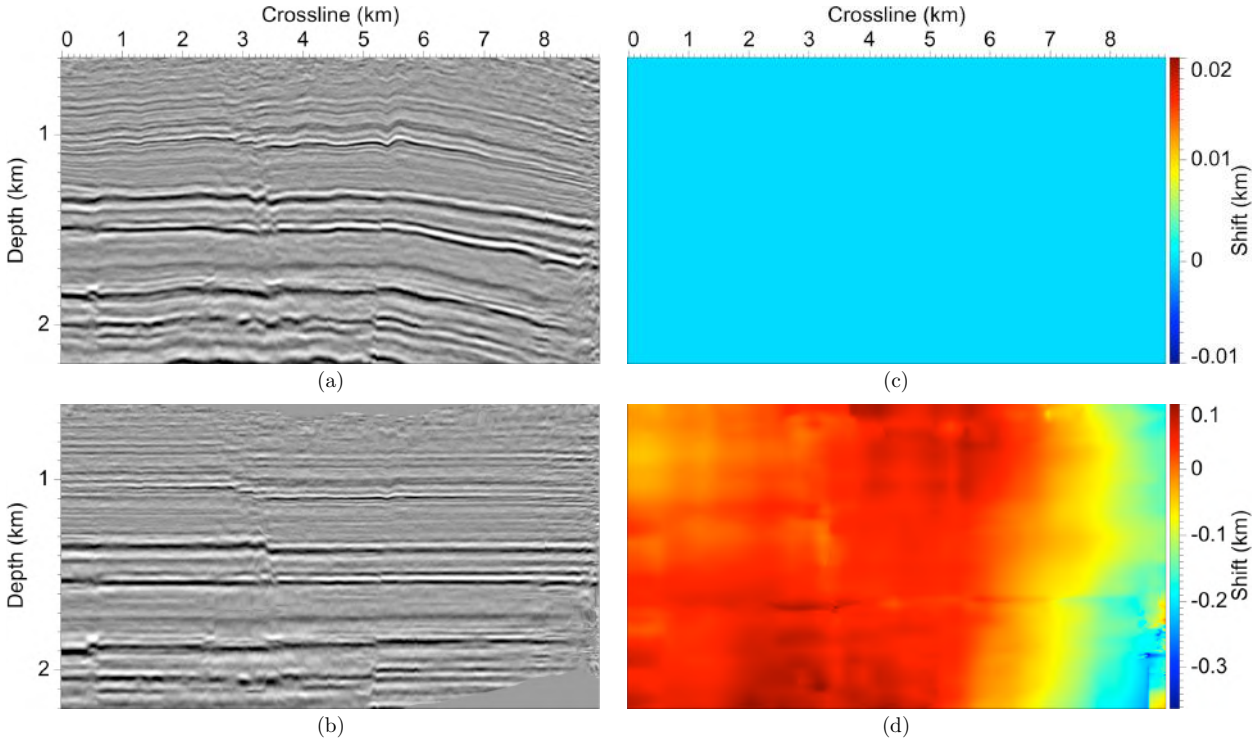


Figure 6. A seismic image (a) flattened by vertical shear (b) with the horizontal (c) and vertical (d) components of the shift vector field $\mathbf{r}(\mathbf{u})$.

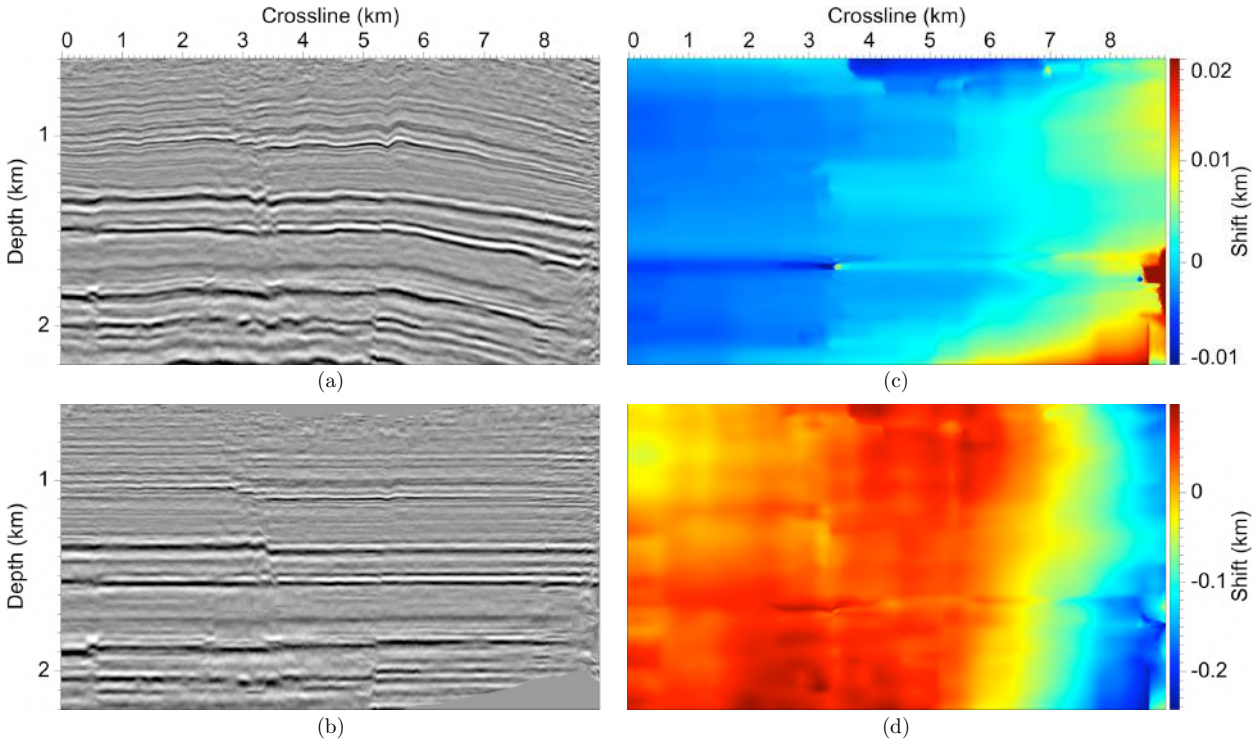


Figure 7. A seismic image (a) flattened by rotation (b) with the horizontal (c) and vertical (d) components of the shift vector field $\mathbf{r}(\mathbf{u})$.

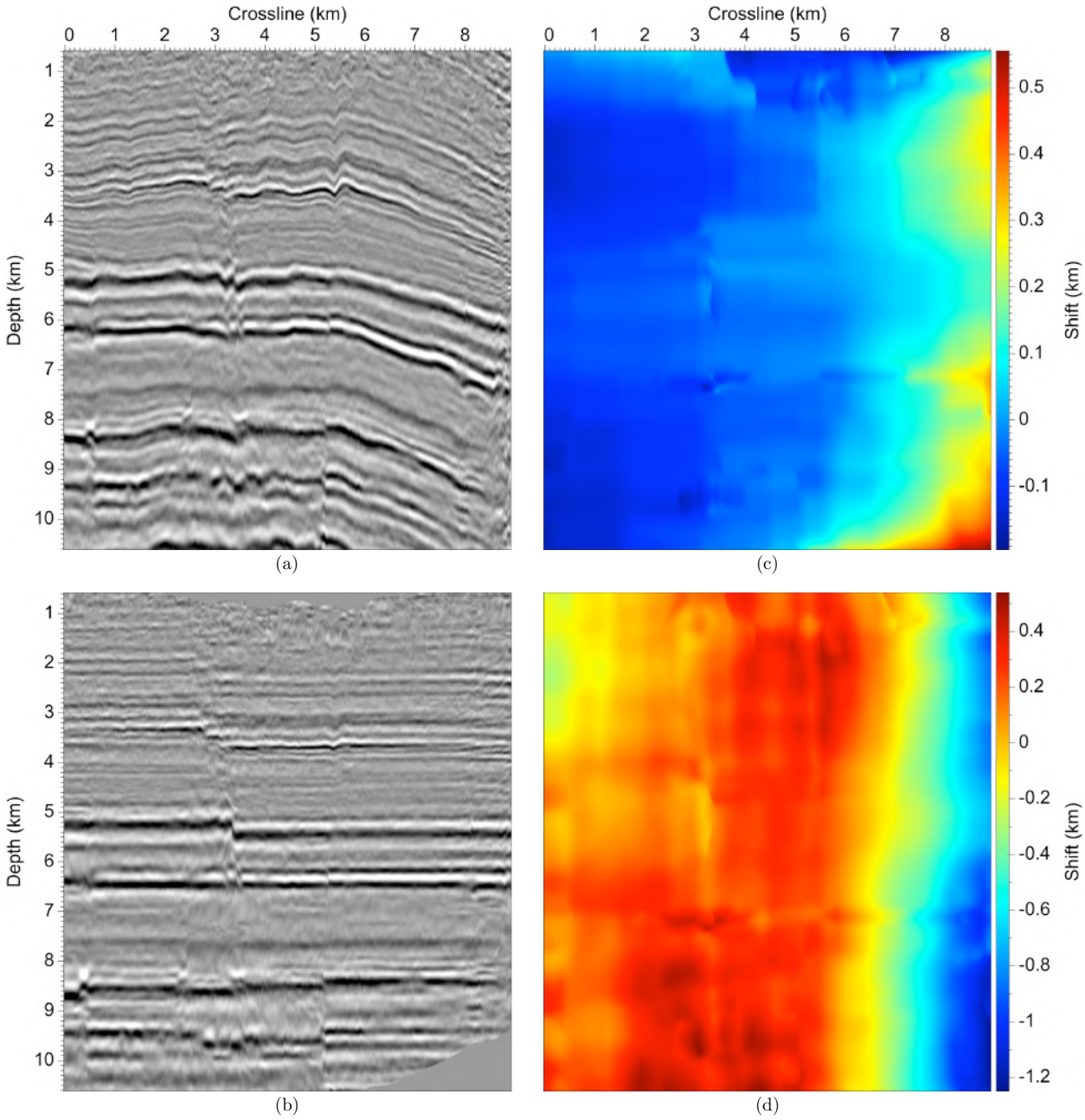


Figure 8. A seismic image (a) flattened by rotation (b) with the horizontal (c) and vertical (d) components of the shift vector field $\mathbf{r}(\mathbf{u})$.

Matrices $\mathbf{R}_v$ and $\mathbf{R}_r$ describe the shift vector fields for pure vertical shear or for pure rotation; but to represent more complex geologic deformations, we must combine both methods. One simple combination is

$$\mathbf{R}_c = (1-\alpha)\,\mathbf{R}_v + \alpha\,\mathbf{R}_r. \tag{23}$$

In terms of the components of the normal vector and the parameter α,

$$\mathbf{R}_c = \begin{bmatrix} \alpha(1-n_3) & -\alpha n_1 \\ \alpha n_1 + (1-\alpha)n_1/n_3 & \alpha(1-n_3) \end{bmatrix}. \tag{24}$$

If we know (or assume) the type of geologic deformation that occurred, then we may choose the corresponding $\alpha \in [0,1]$. Note that $\alpha = 0$ produces vertical shearing, while $\alpha = 1$ produces rotation. We equate equation 17 and equation 24

$$\mathbf{R} = \mathbf{R}_c, \tag{25}$$

to obtain four equations for the partial derivatives of the shift vector field

$$r_{11} = \alpha(1-n_3) \tag{26}$$
$$r_{13} = -\alpha n_1 \tag{27}$$
$$r_{31} = \alpha n_1 + (1-\alpha)n_1/n_3 \tag{28}$$
$$r_{33} = \alpha(1-n_3), \tag{29}$$

which we solve for $\mathbf{r}(\mathbf{u})$ by weighted least-squares. Note from equation 14 that r_{11} and r_{31} are related to flattening. The remaining two equations for r_{13} and r_{33} are related to area preservation, as they determine the length of normal vector $\mathbf{m}$ in the flattened space. If we weight all four equations equally, the method will attempt to flatten an image while scaling locally the area of the image by the determinant of the Jacobian. For most seismic images, it is not possible to flatten everywhere if we constrain the area. Thus, when solving for $\mathbf{r}(\mathbf{u})$, we give more weight to equations 26 and 28 to ensure flattening.

It is simple to verify that the Jacobian for $\mathbf{R}_c$ satisfies equation 10. The composite Jacobian is

$$\mathbf{J}_c = \mathbf{I} - \mathbf{R}_c, \tag{30}$$

and for $\mathbf{J}_c^{\mathsf{T}}\mathbf{n}$, we have

$$\begin{bmatrix} 1-\alpha\,(1-n_3) & -\alpha\,n_1 - (1-\alpha)\,n_1/n_3 \\ \alpha\,n_1 & 1-\alpha\,(1-n_3) \end{bmatrix} \begin{bmatrix} n_1 \\ n_3 \end{bmatrix} = \begin{bmatrix} 0 \\ \alpha + n_3\,(1-\alpha) \end{bmatrix}. \tag{31}$$

As expected, the horizontal component of the normal vector $\mathbf{m}$ in the flattened space is zero. Note that for $\alpha = 1$, $\mathbf{m}$ is also a unit vector, which indicates that flattening by rotation preserves the thickness (measured perpendicular to bedding) of sedimentary layers. This is not the case for flattening by vertical shear ($\alpha = 0$).

One problem with equation 25 is that that shifts $\mathbf{r}(\mathbf{u})$ are a function of $\mathbf{u}$, so that $\mathbf{R} = \mathbf{R}(\mathbf{u})$; but the normal vectors $\mathbf{n}(\mathbf{x})$ are functions of $\mathbf{x}$, so that $\mathbf{R}_c = \mathbf{R}_c(\mathbf{x})$. Therefore, the equation we must solve is

$$\mathbf{R}(\mathbf{u}) = \mathbf{R}_c(\mathbf{x}) = \mathbf{R}_c(\mathbf{u} - \mathbf{r}(\mathbf{u})). \tag{32}$$

This equation is nonlinear, because the right-hand-side depends on the solution $\mathbf{r}(\mathbf{u})$. In practice, we use a simple fixed-point iteration to handle this (generally weak) nonlinearity. We begin with $\mathbf{r}(\mathbf{u}) = 0$, compute $\mathbf{R}_c(\mathbf{u})$, solve for $\mathbf{r}(\mathbf{u})$, compute $\mathbf{R}_c(\mathbf{u} - \mathbf{r}(\mathbf{u}))$, solve for $\mathbf{r}(\mathbf{u})$, and so on until converged. Convergence is fast when the normal vectors vary slowly, i.e., when $\mathbf{n}(\mathbf{u}) \approx \mathbf{n}(\mathbf{x})$.

Figures 4 and 5 show, respectively, examples of flattening by vertical shear and flattening by rotation for a synthetic seismic image. Notice that for the vertical shear case, the horizontal component of the shift vector field $\mathbf{r}(\mathbf{u})$ shown in Figure 4c is zero, which indicates traces are being shifted only vertically. Notice also that thickness measured in the direction normal to the layering is preserved only in the case of rotation (Figure 5b).

Figures 6 and 7 show examples of flattening by vertical shear and flattening by rotation for a 2D slice from a 3D seismic image of Teapot Dome. Note that the images for these examples are stretched vertically, and that if we were to plot the input seismic image with a 1:1 aspect ratio, we would see that the image is nearly flat. Because flattening by vertical shear and flattening by rotation produce identical results for an already flat image, it is not surprising that Figure 6 and Figure 7 are similar.

We may exaggerate the structure in a seismic image by increasing the sampling interval in the depth direction. Figure 8 shows an example of flattening by rotation for the same 2D seismic image shown in Figures 6a and 7a, but with the depth sampling interval increased from 4 m to 25 m. Notice that the horizontal components $r_1(\mathbf{u})$ of the shift vector field $\mathbf{r}(\mathbf{u})$ shown in Figure 8c are larger than those shown in Figure 7c, which are nearly zero.

2.2 Flattening in 3D

The extension of our flattening method to 3D is straightforward. In 3D, the Jacobian matrix for a mapping $\mathbf{x}(\mathbf{u})$ is

$$\mathbf{J} = \begin{bmatrix} \partial x_1/\partial u_1 & \partial x_1/\partial u_2 & \partial x_1/\partial u_3 \\ \partial x_2/\partial u_1 & \partial x_2/\partial u_2 & \partial x_2/\partial u_3 \\ \partial x_3/\partial u_1 & \partial x_3/\partial u_2 & \partial x_3/\partial u_3 \end{bmatrix}, \tag{33}$$

which, from equation 11, is equivalent to

$$\mathbf{J} = \begin{bmatrix} 1-r_{11} & -r_{12} & -r_{13} \\ -r_{21} & 1-r_{22} & -r_{23} \\ -r_{31} & -r_{32} & 1-r_{33} \end{bmatrix}. \tag{34}$$

where $r_{ij} \equiv \partial r_i/\partial u_j$. From equation 10, we have

$$n_1(1-r_{11}) - n_2 r_{21} - n_3 r_{31} = 0 \tag{35}$$
$$n_2(1-r_{22}) - n_1 r_{12} - n_3 r_{32} = 0 \tag{36}$$
$$n_3(1-r_{33}) - n_1 r_{13} - n_2 r_{23} = m_3. \tag{37}$$

Equations 35 and 36 are the flattening equations; given normal vectors computed from an image, applying any shift vector field $\mathbf{r}(\mathbf{u})$ that satisfies both equations will flatten the image. Equation 34 may be written as

$$\mathbf{J} = \mathbf{I} - \mathbf{R}, \tag{38}$$

where $\mathbf{I}$ is a 3×3 identity matrix, and $\mathbf{R}$ is a matrix of partial derivatives of the shift vector field:

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}. \tag{39}$$

Solving equation 38 for $\mathbf{R}$, we obtain

$$\mathbf{R} = \mathbf{I} - \mathbf{J}. \tag{40}$$

To find $\mathbf{R}$, we need a Jacobian matrix that satisfies equation 10.

Again, we consider separately the Jacobian matrix for flattening by vertical shear and flattening by rotation. For vertical shear in 3D, the Jacobian is

$$\mathbf{J}_v = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -n_1/n_3 & -n_2/n_3 & 1 \end{bmatrix}, \tag{41}$$

and for rotation

$$\mathbf{J}_r = \begin{bmatrix} n_3 + n_2^2/(1+n_3) & -n_1 n_2/(1+n_3) & n_1 \\ -n_1 n_2/(1+n_3) & n_3 + n_1^2/(1+n_3) & n_2 \\ -n_1 & -n_2 & n_3 \end{bmatrix}. \tag{42}$$

From equation 18 we obtain the corresponding matrix of partial derivatives for vertical shear

$$\mathbf{R}_v = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ n_1/n_3 & n_2/n_3 & 0 \end{bmatrix}, \tag{43}$$

and for rotation

$$\mathbf{R}_r = \begin{bmatrix} n_1^2/(1+n_3) & n_1 n_2/(1+n_3) & -n_1 \\ n_1 n_2/(1+n_3) & n_2^2/(1+n_3) & -n_2 \\ n_1 & n_2 & 1-n_3 \end{bmatrix}. \tag{44}$$

We form a composite matrix as a linear combination of $\mathbf{R}_v$ and $\mathbf{R}_r$:

$$\mathbf{R}_c = (1-\alpha)\,\mathbf{R}_v + \alpha\,\mathbf{R}_r. \tag{45}$$

Equating equation 39 and 45 gives nine equations for the partial derivatives of the shift field:

$$r_{11} = \alpha n_1^2/(1+n_3) \tag{46}$$
$$r_{12} = \alpha n_1 n_2/(1+n_3) \tag{47}$$
$$r_{13} = -\alpha n_1 \tag{48}$$
$$r_{21} = \alpha n_1 n_2/(1+n_3) \tag{49}$$
$$r_{22} = \alpha n_2^2/(1+n_3) \tag{50}$$
$$r_{23} = -\alpha n_2 \tag{51}$$
$$r_{31} = \alpha n_1 + (1-\alpha) n_1/n_3 \tag{52}$$
$$r_{32} = \alpha n_2 + (1-\alpha) n_2/n_3 \tag{53}$$
$$r_{33} = \alpha(1-n_3) \tag{54}$$

The flattening equations 35 and 36 contain terms r_{11}, r_{12}, r_{21}, r_{22}, r_{31}, and r_{32}. Thus, when solving for $\mathbf{r}(\mathbf{u})$, we give more weight to the corresponding equations 46, 47, 49, 50, 52, and 53. The remaining equations for r_{13}, r_{23} and r_{33} determine the length of the normal vector $\mathbf{m}$, and therefore are related to volume preservation. These equations are usually given less weight in order to ensure flattening. Figure 9 shows an example of flattening by rotation ($\alpha = 1$) for an image of Teapot Dome, with the depth sampling interval increased from 4 m to 25 m to exaggerate the structure in the image.

3 CONCLUSION

The flattening method described in this report is versatile, in that it can flatten an image by vertically shearing the image, by rotating portions of the image, or by a combination of vertical shear and rotation. Because the method solves for a shift vector field, rather than a scalar field of vertical shifts, it is not constrained to flatten by vertical shearing only. A shift vector field can more accurately represent geologic deformation, and can be used to flatten images while minimizing non-geologic distortions of image features.

However, in order to flatten an image according to the true geologic deformation, our method currently requires that we specify the type of deformation that occurred. In practice, it may be difficult to determine the true deformation, even when provided with relevant geologic information. But this shortcoming suggests an interesting research question: can we estimate the type of deformation from the image itself?

One potentially useful measure of the correctness (relative to the true geologic deformation) of a flattening method is the determinant of the Jacobian matrix. In 2D, the determinant gives the ratio between an infinitesimal area in present-day space and the corresponding area in flattened space. In 3D, the determinant gives the ratio between infinitesimal volumes. Thus, if the determinant is unity, then area or volume has been preserved while flattening. Assuming that the true deformation is area- or volume-preserving, we might use the determinant of the Jacobian to quantify how well a flattening method reflects true geologic deformation.

For now we must either assume, or we must use geologic information to estimate, the type of deformation. In any case, for most images, geologic deformation cannot be described with only a scalar field of vertical shifts. Solving for a vector shift field as described in this report is a simple way to flatten images with non-vertical deformations.

ACKNOWLEDGEMENTS

Thanks to the Rocky Mountain Oilfield Test Center for providing the 3D seismic image of Teapot Dome, and

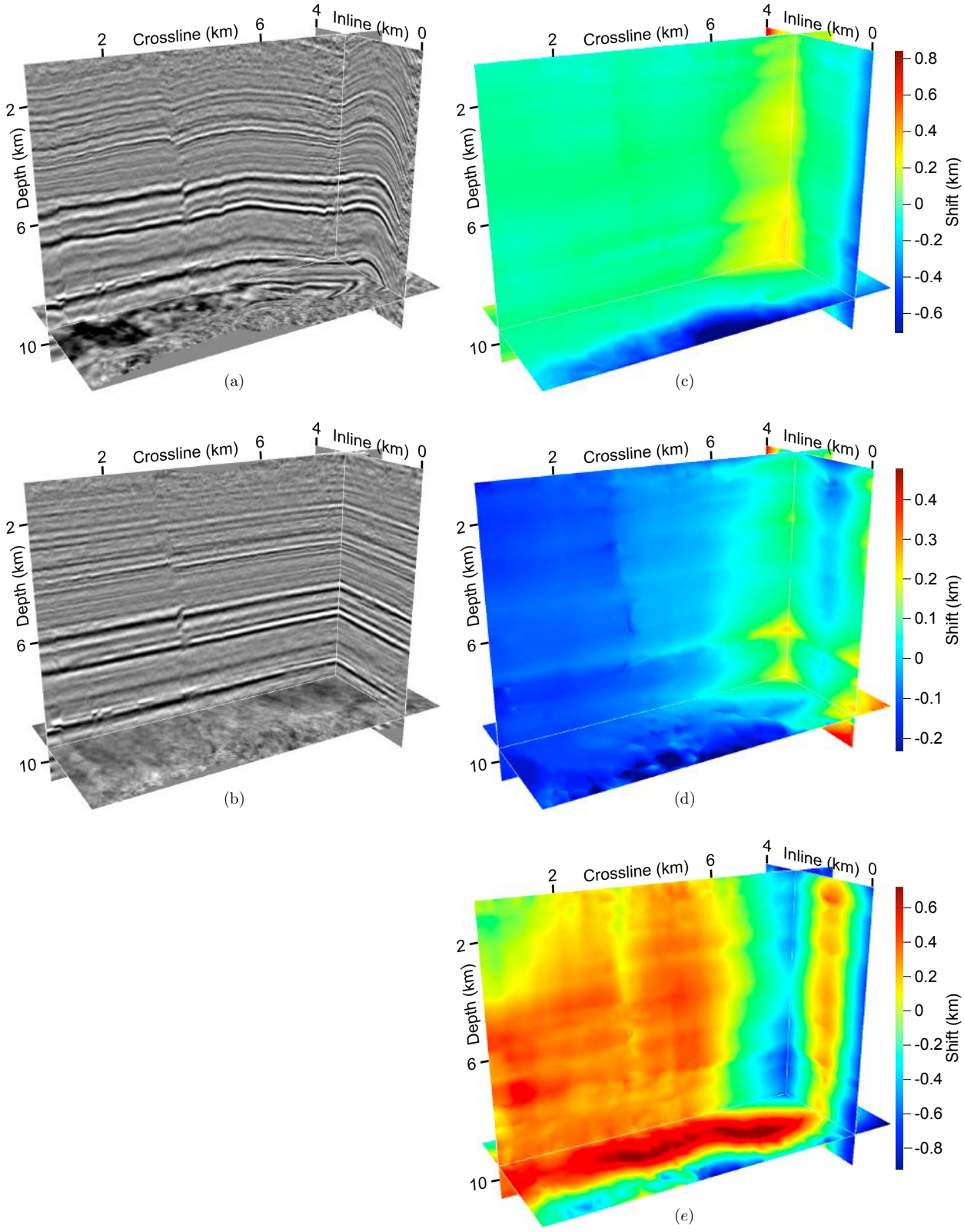


Figure 9. A 3D seismic image (a) flattened by rotation (b) with the inline shift r_1 (c), crossline shift r_2 (d), and vertical shift r_3 (e) of the shift vector field $\mathbf{r}(\mathbf{u})$.

thanks to Transform Software for providing the time-to-depth conversion of that image. Thanks also to Derek Parks and Ran Xuan, whose prior work on seismic image flattening was a helpful starting point for this research. This research was supported by the sponsors of the Center for Wave Phenomena at the Colorado School of Mines.

REFERENCES

Fehmers, G. C., and C. F. W. Höcker, 2003, Fast structural interpretations with structure-oriented filtering: Geophysics, **68**, 1286–1293.

Hale, D., 2006, Recursive gaussian filters: Technical Report CWP-546, Center for Wave Phenomena, Colorado School of Mines.

Lomask, J., A. Guitton, S. Fomel, J. F. Claerbout, and A. Valenciano, 2006, Fast structural interpretations with structure-oriented filtering: Geophysics, **68**, 1286–1293.

Parks, D., 2010, Seismic image flattening as a linear inverse problem: M.Sc. thesis CWP-643, Center for Wave Phenomena, Colorado School of Mines.

van Vliet, L. J., and P. W. Verbeek, 1995, Estimators for orientation and anisotropy in digitized images: Proceedings of the first annual conference of the Advanced School for Computing and Imaging ASCI95, Heijen (The Netherlands), 442–450.

APPENDIX A: INVERSE SHIFT VECTORS

We considered in this report the mapping

$$\mathbf{x}(\mathbf{u}) = \mathbf{u} - \mathbf{r}(\mathbf{u}), \tag{A1}$$

which describes the deformation of a point $\mathbf{u}$ in the flattened space. Alternatively, we may consider the inverse mapping

$$\mathbf{u}(\mathbf{x}) = \mathbf{x} + \mathbf{s}(\mathbf{x}), \tag{A2}$$

where $\mathbf{s}(\mathbf{x})$ is a shift vector field that satisfies

$$\mathbf{s}(\mathbf{x}) = \mathbf{r}(\mathbf{u}) = \mathbf{r}(\mathbf{x} + \mathbf{s}(\mathbf{x})). \tag{A3}$$

One advantage of using $\mathbf{s}(\mathbf{x})$ rather than $\mathbf{r}(\mathbf{u})$ is that the normal vectors $\mathbf{n}$ are functions of $\mathbf{x}$. Thus, it may be possible to obtain a linear set of equations for $\mathbf{s}(\mathbf{x})$ that can be solved without the fixed-point iterations required when solving for $\mathbf{r}(\mathbf{u})$. In fact, this is the approach used by Parks (2010) to solve for a scalar shift field for vertical shearing only.

Unfortunately, it is difficult to flatten while preserving this linearity when solving for a more general vector shift field. To see this, consider flattening in 3D using the inverse shift vectors $\mathbf{s}(\mathbf{x})$. Analogous to equation 8, we write the first-order Taylor series approximation for $\mathbf{u}$ as

$$\mathbf{u} = \mathbf{u}_0 + \mathbf{K}\,(\mathbf{x} - \mathbf{x}_0), \tag{A4}$$

where $\mathbf{K}$ is the Jacobian of the transformation $\mathbf{u}(\mathbf{x})$. From equation A2, we have

$$\mathbf{K} = \begin{bmatrix} 1+s_{11} & s_{12} & s_{13} \\ s_{21} & 1+s_{22} & s_{23} \\ s_{31} & s_{32} & 1+s_{33} \end{bmatrix} = \mathbf{I} + \mathbf{S}. \tag{A5}$$

where $s_{ij} \equiv {}^{\partial s_i}/_{\partial x_j}$. The normal vectors before and after flattening are related as

$$\mathbf{n} = \mathbf{K}^{\mathsf{T}}\mathbf{m}, \tag{A6}$$

which gives

$$\mathbf{K}^{-\mathsf{T}}\mathbf{n} = \mathbf{m}. \tag{A7}$$

Note that the inverse of the Jacobian of the transformation $\mathbf{u}(\mathbf{x})$ is equal to the inverse of the Jacobian of $\mathbf{x}(\mathbf{u})$:

$$\mathbf{K}^{-1} = \mathbf{J}, \tag{A8}$$

and that equation A7 is equivalent to equation 10, except that it is written in terms of the shift vectors $\mathbf{s}(\mathbf{x})$ instead of $\mathbf{r}(\mathbf{u})$. For a flattened image, all components of the vector $\mathbf{m}$ except for m_3 must be zero. Thus to form flattening equations, we need the inverse of the Jacobian $\mathbf{K}$:

$$\mathbf{K}^{-1} = \frac{1}{\det \mathbf{K}} \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}, \tag{A9}$$

where

$$k_{11} = (1+s_{22})(1+s_{33}) - s_{23}s_{32} \tag{A10}$$
$$k_{12} = s_{13}s_{32} - s_{12}(1+s_{33}) \tag{A11}$$
$$k_{13} = s_{12}s_{23} - s_{13}(1+s_{22}) \tag{A12}$$
$$k_{21} = s_{23}s_{31} - s_{21}(1+s_{33}) \tag{A13}$$
$$k_{22} = (1+s_{11})(1+s_{33}) - s_{13}s_{31} \tag{A14}$$
$$k_{23} = s_{13}s_{21} - s_{23}(1+s_{11}) \tag{A15}$$
$$k_{31} = s_{21}s_{32} - s_{31}(1+s_{22}) \tag{A16}$$
$$k_{32} = s_{12}s_{31} - s_{32}(1+s_{11}) \tag{A17}$$
$$k_{33} = (1+s_{11})(1+s_{22}) - s_{12}s_{21}. \tag{A18}$$

From equation A7, the flattening equations corresponding to the zero-valued horizontal components of $\mathbf{m}$ are

$$k_{11}n_1 + k_{12}n_2 + k_{13}n_3 = 0 \tag{A19}$$
$$k_{21}n_1 + k_{22}n_2 + k_{23}n_3 = 0. \tag{A20}$$

Unlike flattening equations 35 and 36 for $\mathbf{r}(\mathbf{u})$, which contained only six elements of the matrix $\mathbf{R}$, these equations contain all nine elements of the matrix $\mathbf{S}$. So to flatten, we need to satisfy all nine equations for the partial derivatives s_{ij}. Recall that if we weight equally all nine equations, then the method will attempt to both

flatten and preserve volume. For most images, it is not possible to flatten and preserve volume everywhere.

By solving for shifts $\mathbf{r}(\mathbf{u})$, we avoided this problem by simply weighting the flattening equations. For the inverse shifts $\mathbf{s}(\mathbf{x})$, we cannot separate the flattening equations from the volume preserving equations. For this reason, we solve for shifts $\mathbf{r}(\mathbf{u})$ instead.