

CORONAL MASS EJECTION RECONSTRUCTION FROM THREE VIEWPOINTS VIA SIMULATION MORPHING. I. THEORY AND EXAMPLES

RICHARD A. FRAZIN

Department of Atmospheric, Oceanic and Space Sciences, University of Michigan, Ann Arbor, MI 48109, USA; rfrazin@umich.edu
Received 2012 July 20; accepted 2012 October 18; published 2012 November 20

ABSTRACT

The problem of reconstructing the three-dimensional (3D) density distribution of a coronal mass ejection (CME) from three simultaneous coronagraph observations is timely in that the COR1 and COR2 coronagraphs on the dual-spacecraft *STEREO* mission complement the LASCO coronagraphs on the *SOHO* satellite and the Mk4 on Mauna Loa. While the separation angle between the *STEREO* spacecraft and the Earth depends on the time since the launch in 2006, the reconstruction problem is always severely underinformed. So far, all 3D reconstruction efforts have made use of relatively simple parameterized models in order to determine the 3D structure of the CME. Such approaches do not utilize the power of 3D MHD simulation to inform the reconstruction. This paper considers the situation in which a specific CME event observed in coronagraphs from three viewpoints is later simulated by solving MHD equations. The reconstruction is then subjected to an invertible morphological operator chosen so that morphed MHD simulation is most consistent with the three-viewpoint coronagraph data. The morphological operations are explained mathematically and synthetic examples are given. The practical application to reconstructing CMEs from *STEREO* and *SOHO* data is discussed.

Key words: Sun: coronal mass ejections (CMEs) – techniques: image processing

Online-only material: color figures

1. INTRODUCTION

Coronal mass ejections (CMEs) are important drivers of space weather and are of fundamental importance to understanding stellar winds and magnetic evolution (Schwenn 2006; Chen 2011). Predicting the effects of CMEs on the heliosphere and planetary magnetospheres is increasingly depending on MHD simulation and knowledge of the three-dimensional (3D) CME structure. Despite recent progress in MHD modeling of CMEs (Fan 2005; Manchester et al. 2008, 2012; Lynch et al. 2009; Roussev et al. 2012), agreement with observations leaves much to be desired.

On the other hand, line-of-sight (LOS) effects make empirical determination of 3D CME structure without strong geometrical assumptions impossible. Until the *STEREO* mission (Howard et al. 2008), CMEs could only be seen in coronagraph images from a single viewpoint, and with two *STEREO* spacecraft plus the Large Angle and Spectrometric Coronagraph Experiment on the *Solar and Heliospheric Observatory* and/or the ground-based Mk4, they can be seen from three viewpoints simultaneously. Although three viewpoints provide more information, they fall far short of providing the information necessary for tomographic reconstruction (Frazin et al. 2009). Thus, most investigators have elected to fit parametric geometrical models of CMEs to the coronagraph data, e.g., Lugaz et al. (2010) and Thernisien (2011). The literature on CME reconstruction is reviewed in Mierla et al. (2010), Thernisien et al. (2011), and Frazin et al. (2009). For a review on 3D reconstruction of prominences, see Bemporad (2011).

Thus far, no method has utilized the power of 3D MHD simulation of CMEs (Manchester et al. 2008; van der Holst et al. 2009; Lynch et al. 2009; Wu et al. 2011) for reconstruction, probably because finding a way to merge simulation and observational data is not obvious. One might consider data assimilation (Evensen 2007; Butala et al. 2008; D’Amato et al. 2011), but as the coronagraph data provide only line integrals

of the density (leaving the other MHD variables unconstrained), it seems unlikely to produce a reasonable solution. The novel idea introduced in this paper is to combine MHD simulation with non-rigid image registration techniques in order to create a spatially “morphed” simulation that agrees with the coronagraph data better than the initial (unmorphed) simulation. Image registration is a large topic in the medical literature that generally considers spatial transformations of an image in order to make its anatomical features “line-up” with the same features in a reference image. Image registration is used for a number of purposes, such as correcting for patient motion (e.g., breathing), and using images from many different patients for statistical studies (Crum et al. 2004). In this paper, the term “morphing” is synonymous with non-rigid image registration.

2. CME RECONSTRUCTION VIA MORPHOLOGICAL TRANSFORMATION

2.1. Mathematical Preliminaries

Consider triple-viewpoint coronagraph observations of a CME at a fixed time t . The CME has an unknown density $\rho(\mathbf{r})$, where $\mathbf{r} \in \mathbb{R}^3$ is the 3D coordinate vector. This density $\rho \in L^2$ can be thought of as a function that maps L^2 onto \mathbb{R} , where L^2 is the space of square integrable functions on \mathbb{R}^3 . The coronagraph images contain information about $\rho(\mathbf{r})$, as they detect the Thomson scattered solar disk light (van de Hulst 1950; Frazin & Janzen 2002; Frazin et al. 2010). All of the intensity values in the usable pixels of the three coronagraph images can be placed into a single vector y with N components, similarly to the procedure described in Frazin & Janzen (2002). The relationship between the k th component of y , y_k , and ρ is an integral over the k th LOS, given by the Thomson scattering function. Thus, the model for this LOS integration can be summarized as $y_k = A_k(\rho(\mathbf{r}))$, where A_k is a functional that is the Thomson scattering integral operator over the k th LOS. Considering all of the N components of y simultaneously leads

to the notation $y = A(\rho(\mathbf{r}))$, where A is a linear map from L^2 to \mathbb{R}^N . As discussed in Frazin et al. (2009), since three viewpoints are not nearly enough information to invert A with any type of classical methodology, it is necessary to seek another avenue.

Consider an MHD simulation of the CME event. One of the simulation outputs is the density $\rho_0(\mathbf{r})$, from which synthetic coronagraph images $y_0 \equiv A(\rho_0(\mathbf{r}))$ may be calculated. One is likely to find significant differences between y and y_0 , and it may be assumed that $\rho_0(\mathbf{r}) \neq \rho(\mathbf{r})$, and, thus, $\rho(\mathbf{r})$ remains unknown. However, if one can find a modified version of $\rho_0(\mathbf{r})$, say, $\rho'(\mathbf{r})$ such that $y' = A(\rho'(\mathbf{r}))$ is closer to y than is y_0 , there is potential to learn more about the CME. The aim of the present endeavor is to find a plausible “morphological transformation” of $\rho_0(\mathbf{r})$ so that y' is closer to y than is y_0 . Here the term “morphological transformation” refers to remapping the spatial domain of the simulation output. A morphological transformation operator \mathcal{M}_φ is defined as $\mathcal{M}_\varphi(\rho_0(\mathbf{r})) \equiv \rho_0(\mathbf{r}'_\varphi(\mathbf{r}))$, where $\mathbf{r}'_\varphi(\mathbf{r}) : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is a (generally nonlinear) remapping of the coordinate space and φ is a vector of parameters that controls the remapping. Defining $y_\varphi \equiv A(\rho_0(\mathbf{r}'_\varphi(\mathbf{r})))$, the aim is to find a plausible transformation \mathcal{M}_φ such that y_φ is a better match to y than y_0 , and the reconstruction is then taken to be $\rho_\varphi(\mathbf{r}) \equiv \mathcal{M}_\varphi(\rho_0(\mathbf{r}))$.

The 3×3 Jacobian matrix of the transformation is defined as $J_\varphi = \partial \mathbf{r}'_\varphi / \partial \mathbf{r}$ and its determinant is denoted as $|J_\varphi|$. Only a finite volume of \mathbb{R}^3 is occupied by the CME and it is assumed to be contained within the computational reconstruction domain $\mathcal{U} \subset \mathbb{R}^3$. It is assumed that \mathcal{U} is simply connected, such as the interior of a cube or a sphere. The compliment of \mathcal{U} (i.e., the portion of \mathbb{R}^3 outside of \mathcal{U}) will no longer be considered. Now, the transformation \mathbf{r}'_φ is invertible on \mathcal{U} if and only if $|J_\varphi|$ exists and is strictly positive or strictly negative everywhere on \mathcal{U} , meaning that sign changes or zeros in $|J_\varphi|$ correspond to lack of invertibility. Only transformations with differentiable inverses and positive values of $|J_\varphi|$ are sought (a negative determinant corresponds to reversal of orientation). Such transformations are called diffeomorphisms (Christensen et al. 1996), and the use of diffeomorphisms in image processing is reviewed in Chun & Fessler (2009), Le Guyader et al. (2012), Cordero-Grande et al. (2012), and references therein. In one dimension, diffeomorphisms are simply monotonic transformations. In two dimensions (2D), diffeomorphisms can be thought of as stretching and twisting a sheet of rubber without cutting or folding, so that a Cartesian coordinate grid line would never cross itself. A “plausible transformation” sought here is a diffeomorphism that improves agreement between y_φ and y .

2.2. Expansions of the Object and Deformation

Following Chun & Fessler (2009), the transformation can be written as

$$\mathbf{r}' = \mathbf{r} + \mathbf{q}(\mathbf{r}; \varphi), \quad (1)$$

where $\mathbf{q} = (q_x(\mathbf{r}; \varphi_x), q_y(\mathbf{r}; \varphi_y), q_z(\mathbf{r}; \varphi_z))$ is the deformation, which is controlled by the parameter vector $\varphi = (\varphi_x, \varphi_y, \varphi_z)$. The x -component of deformation is given by the basis expansion:

$$q_x(\mathbf{r}; \varphi_x) = \sum_{ijk} \varphi_x^{ijk} \beta\left(\frac{x}{s} - i\right) \beta\left(\frac{y}{s} - j\right) \beta\left(\frac{z}{s} - k\right), \quad (2)$$

where i, j , and k are spatial indices that refer to the spline knots, $\mathbf{r} = (x, y, z)$, s is the spacing of the spline knots (s is typically an integer number of pixels), and $\beta(t)$ is the third-order polynomial B-spline basis function (Unser 1999). Similar

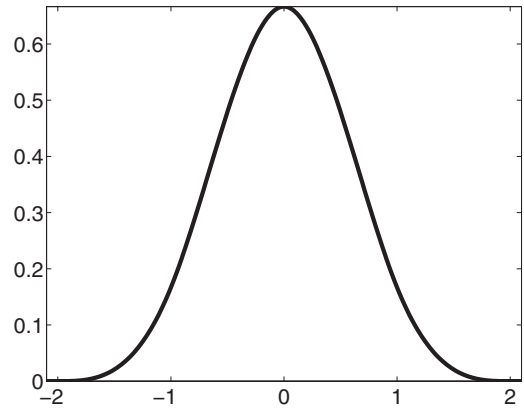


Figure 1. Cubic B-spline, $\beta(t)$, in Equation (3). Note that this function is only nonzero in the interval $(-2, 2)$.

equations hold for q_y and q_z . In this way, the deformation is a continuous function controlled by a finite number of parameters. Thus, if the computational volume is a cube with M^3 voxels and the spline knots are placed at every $s = p$ pixels in each dimension (meaning that a spline knot is in the center of p^3 voxels), then φ_x has $(M/p)^3$ components. Since the parameter vector $\varphi = (\varphi_x, \varphi_y, \varphi_z) = (\{\varphi_x^{ijk}\}, \{\varphi_y^{ijk}\}, \{\varphi_z^{ijk}\})$, φ has a total $3(M/p)^3$ components. Analytically, the polynomial B-spline of order q is given by the formula (Unser 1999)

$$\beta(t) = \frac{1}{q!} \sum_{k=0}^{q+1} \binom{q+1}{k} (-1)^k \left(t - k + \frac{q+1}{2}\right)_+^q, \quad (3)$$

where the first factor after the sum symbol is the binomial coefficient, q is the degree of the spline, and the $(\)_+^q$ operator indicates that if the value of the argument negative the result is 0, otherwise the q th power is taken. B-splines are a very useful basis because they are differentiable up to order q , have compact support and readily lend themselves to rapid processing (Unser 1999). The function $\beta(t)$ for $q = 3$ is shown in Figure 1.

Since $\rho_0(\mathbf{r})$ and $\mathcal{M}_\varphi(\rho_0(\mathbf{r})) \in L^2$, they need to be defined as (piecewise) continuous functions. For this, a third-order B-spline function expansion also suffices:

$$\rho_0(\mathbf{r}) = \sum_{ijk} \psi^{ijk} \beta(x - i) \beta(y - j) \beta(z - k), \quad (4)$$

where ψ is a vector of coefficients. As one is likely to perform most manipulations of this sort on a digital computer, it is useful to consider a sampling of \mathbf{r} , which is most easily done on a regular grid of voxels (3D; or pixels in 2D). The most convenient sampling grid is the set of integers (which become pixel/voxel numbers). Thus, the values of \mathbf{r} of interest are a range of integer values for the x, y , and z coordinates, of which there are M^3 triads of values, one for each of the M^3 voxels. For example, assume the integers i', j' , and k' are within the computation cube, then one may evaluate $\rho_0([i', j', k'])$ in Equation (4) by setting $x = i', y = j', z = k'$, which will result in a sum over several nonzero terms, corresponding to the nearest neighbors in each dimension. Let $\boldsymbol{\rho}_0$ represent the vector of values ρ_0 on this sampling grid of \mathbf{r} , i.e., $\boldsymbol{\rho}_0$ is a vector with M^3 components. Then, using Equation (4), $\boldsymbol{\rho}_0$ is given by

$$(\boldsymbol{\rho}_0)_m = \sum_n \psi^n \beta(i_n - i_m) \beta(j_n - j_m) \beta(k_n - k_m), \quad (5)$$

where $\{i_m\}$, $\{j_m\}$, and $\{k_m\}$ are index vectors with M^3 components so that as m goes from 0 to $M^3 - 1$, the vector $\mathbf{r}_m \equiv (i_m, j_m, k_m)$ samples all of the voxel centers, and $\psi^m \equiv \psi^{i_m j_m k_m}$. As Equation (5) is linear and finite dimensional, it can be written in matrix–vector form as

$$\boldsymbol{\rho}_0 = \mathbf{T}_0 \boldsymbol{\psi}, \quad (6)$$

where $\boldsymbol{\psi} \equiv \{\psi^m\}$ and $(\mathbf{T}_0)_{mn} \equiv \beta(i_n - i_m)\beta(j_n - j_m)\beta(k_n - k_m)$. Note that there is no significant dimension reduction in Equation (6), in fact, for a common choice of the spline representation $\boldsymbol{\psi}$ has the same number of components as $\boldsymbol{\rho}_0$, implying that \mathbf{T}_0 is a square matrix (Unser 1999).

Since an MHD simulation is unlikely to represent the density on a spline basis, determining the values of the vector $\boldsymbol{\psi}$ is a separate pre-processing step that can be carried out very rapidly (Unser 1999). Once $\boldsymbol{\psi}$ and φ have been determined, the reconstruction of the CME density is simply:

$$\rho_\varphi(\mathbf{r}) = \mathcal{M}_\varphi(\rho_0(\mathbf{r})) \quad (7)$$

$$= \rho_0(\mathbf{r} + \mathbf{q}_\varphi(\mathbf{r})) \quad (8)$$

$$= \sum_{ijk} \psi^{ijk} \beta(x + q_x(\mathbf{r}; \varphi_x) - i)\beta(y + q_y(\mathbf{r}; \varphi_y) - j) \\ \times \beta(z + q_z(\mathbf{r}; \varphi_z) - k), \quad (9)$$

where q_x is given by Equation (2). Similarly to Equation (6), Equation (9) can be written in matrix–vector notation as

$$\boldsymbol{\rho}_\varphi = \mathbf{T}_\varphi \boldsymbol{\psi}, \quad (10)$$

where $\boldsymbol{\rho}_\varphi$ is a sampled version of $\rho_\varphi(\mathbf{r})$ at points $\{\mathbf{r}_m\}$, and the deformation matrix is defined as

$$(\mathbf{T}_\varphi)_{mn} \equiv \beta(i_n + q_x(\mathbf{r}_n; \varphi_x) - i_m)\beta(j_n + q_y(\mathbf{r}_n; \varphi_y) - j_m) \\ \times \beta(k_n + q_z(\mathbf{r}_n; \varphi_z) - k_m). \quad (11)$$

Note that Equation (10) is linear in the expansion coefficients of the simulation $\boldsymbol{\psi}$, but it is more complex than Equation (6) because the q_x , q_y , and q_z terms in Equation (11) all involve expansions, as in Equation (2). Note that when $\varphi = 0$, which corresponds to zero deformation, $\mathbf{T}_\varphi = \mathbf{T}_0$ and $\boldsymbol{\rho}_\varphi = \boldsymbol{\rho}_0$.

2.3. Determining φ

Given a pixelized coronagraph observations y , the corresponding synthetic coronagraph images from the morphed simulation can be written as

$$y_\varphi = \mathbf{A}\mathbf{T}_\varphi\boldsymbol{\rho}_\varphi, \quad (12)$$

where \mathbf{A} is a discrete approximation of the Thomson scattering operator (Frazin & Janzen 2002). The values of φ are determined by minimizing a scalar difference metric $\mathcal{D}(y, y_\varphi)$ with respect to φ . Recall that for the case considered here, the vector y consists of several coronagraph images, therefore, $\mathcal{D}(y, y_\varphi)$ should be a useful measure of the similarity between image vectors y and y_φ . In case of real data, \mathcal{D} might be required to be robust to noise and various artifacts in the observed images such as stray light concentrations, diffraction rings, calibration differences between spacecraft, etc. Difficulties in some coronagraph images are discussed in the coronagraph intercalibration paper by Frazin et al. (2012). While finding

the best form of \mathcal{D} for a given set of instruments is a challenging problem, for the purposes of this paper it suffices to use a very simple form:

$$\mathcal{D}(y, y_\varphi) = \|y - y_\varphi\|^2 = \|y - \mathbf{A}\mathbf{T}_\varphi\boldsymbol{\psi}\|^2. \quad (13)$$

Unless $p \gg 1$ (which allows only coarse deformations), Equation (13) has roughly as many free parameters as the standard tomography problem. As reconstruction from only a few viewpoints is a vastly underdetermined problem (Frazin et al. 2009), at first glance it might not appear that much has been gained by taking this approach. However, requiring $|J_\varphi| > 0$ globally greatly restricts the available solution space, in effect, regularizing the problem.

Minimizing $\mathcal{D}(y, y_\varphi)$ with respect to φ is unlikely to produce to result in a deformation with the property $|J_\varphi| > 0$, so, instead, it is useful to perform the following optimization:

$$\hat{\varphi} = \underset{\varphi}{\operatorname{argmin}} \{\mathcal{D}(y, y_\varphi) + \lambda\mathcal{P}(\varphi)\}, \quad (14)$$

where $\hat{\varphi}$ is the estimate of φ based on the data y , $\mathcal{P}(\varphi)$ is a penalty function that enforces desirable properties of the deformation (e.g., $|J_\varphi| > 0$) globally, and λ is a parameter that controls the relative importance of the penalty term. Chun & Fessler (2009) review the literature on the restrictions of spline coefficients that result in diffeomorphic maps, and introduce a new penalty function that allows fairly large deformations and depends only on the differences between spline coefficients at neighboring knot positions. We use the Chun & Fessler (2009) penalty, also given in Appendix C, for the examples given here.

The optimization problem posed in Equation (14) can be difficult depending on the choice of the functionals \mathcal{D} and \mathcal{P} . The simple form of \mathcal{D} in Equation (13) results in maximum tractability for the optimization problem; however, the problem is still non-quadratic in φ and is likely to have many local minima, as is common with image registration problems. These issues are discussed in, e.g., Wachowiak et al. (2004). Remarkably enough, a straightforward implementation of a multi-scale conjugate gradient worked well in all of our numerical experiments, including the results presented here. The multi-scale conjugate gradient algorithm is given in Appendix B.

3. EXAMPLES

Section 2.2 gives the expansions for the case of three spatial dimensions (3D), but for simplicity, only 2D examples are given here. To reduce the expressions in Section 2.2, one only needs to ignore the factors corresponding to the z dimension.

These examples utilize the 2D simulations calculated with the Center for Shock Radiation Hydrodynamics (CRASH) simulation code (van der Holst et al. 2011). The CRASH code simulates shock tube experiments carried out by the experimentalists at CRASH. In the CRASH experiments, a 20 μm thick beryllium foil disk is irradiated with a high-power laser. In the simulation, the interaction between the laser and the Be disk is simplified by assuming that the driving radiation is thermalized. This drives a shock wave down a cylindrical plastic tube about 4 mm in length that is filled with xenon gas (Myra et al. 2012). The CRASH simulations shown here were calculated in (adaptive mesh) cylindrical coordinates under the assumption of azimuthal symmetry. The code allows various beryllium target thicknesses and laser power settings. Here we

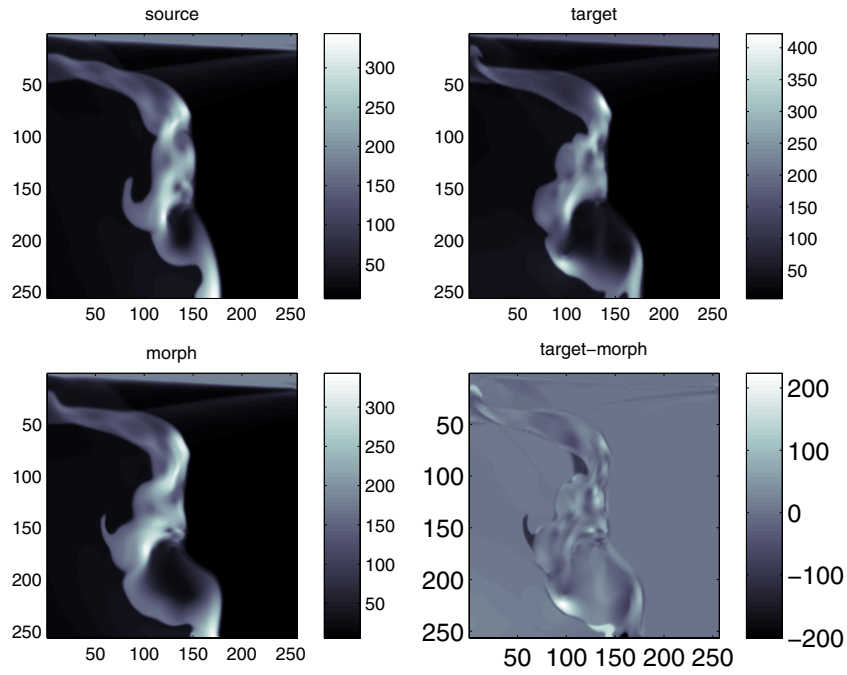


Figure 2. Results from Experiment 0. Upper left: the shock density in run A (source). Upper right: the shock density in run B. Lower right: density difference of target–morphed source. Lower left: density of morphed source. (A color version of this figure is available in the online journal.)

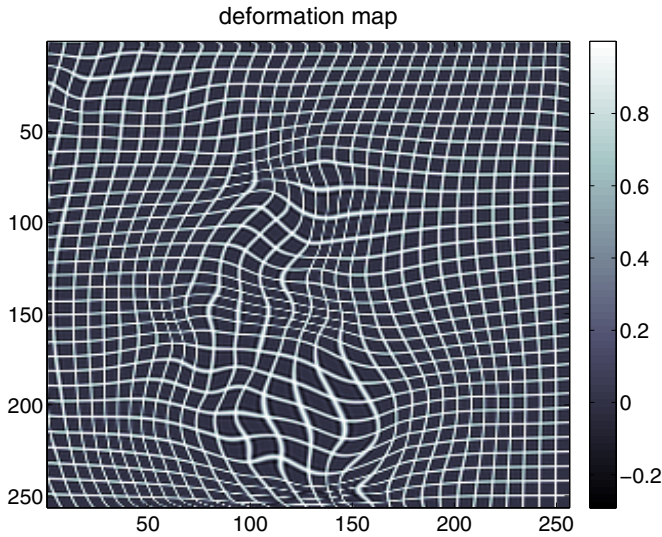


Figure 3. Deformation grid found for Experiment 0. (A color version of this figure is available in the online journal.)

take two runs of the simulation with different parameters, taken at the same instant in time after the laser firing. We will call these runs A and B. The upper left panel of Figure 2 shows the density for run A and the upper left shows that of run B. The densities from each run were binned to square arrays of 256^2 pixels. The density values in the figure (given by the color scales of the images) are in kg m^{-3} , and each pixel is about $1.35 \mu\text{m}$. Each panel shows the x – y plane in which the shock is propagating in the $+x$ direction. Due to the azimuthal symmetry, the bottom edge of each panel corresponds to the axis of the cylindrical shock tube, and the upper edge corresponds to the wall of the tube.

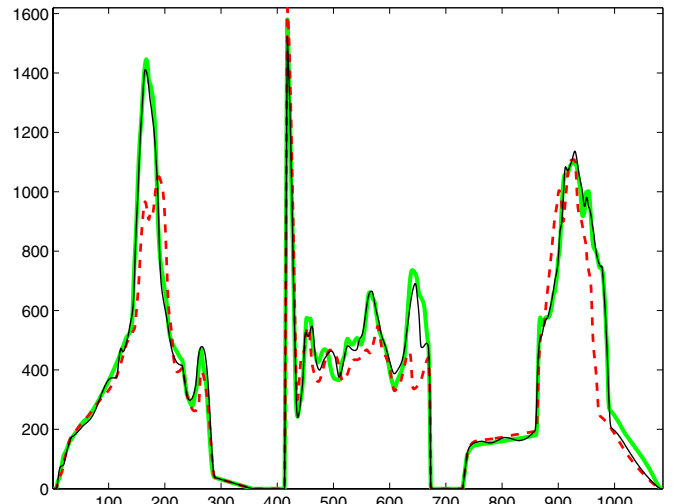


Figure 4. Projection data in Experiment 1. The thick green line is the data vector used for the inversion, the thin black line is the simulated data vector made from the morphed source, and the red dashed line is the simulated data from the source before morphing. (A color version of this figure is available in the online journal.)

Experiment 0 shows the ability of the algorithm to morph the source into the target given all of the information about the target. In Experiment 0, the run A’s 256^2 density values (put into a single column vector) played the role of ρ_0 in Section 2.2. This density vector is called the “source.” The data vector y was similarly given by run B’s 256^2 density values, thus, requiring A to be the $256^2 \times 256^2$ identity matrix, i.e., $y = \rho_{\text{target}}$. Run B’s density vector is called the “target.” We used the multi-scale conjugate gradient method described in Appendix B to determine φ and thereby morphed the source to better match the target. The image of the morphed source,

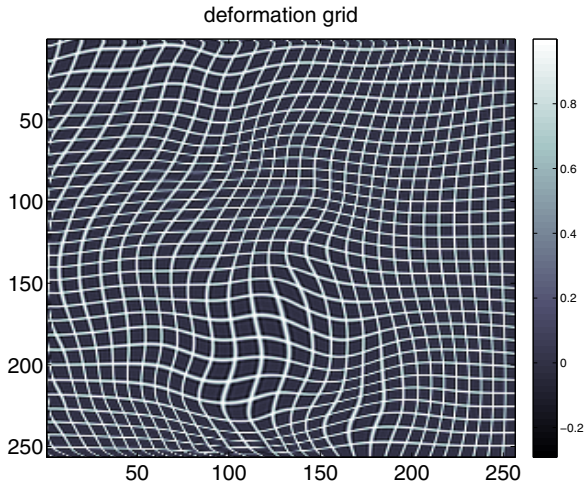


Figure 5. Deformation grid found for Experiment 1.

(A color version of this figure is available in the online journal.)

i.e., $\mathbf{T}_\varphi \psi$, is shown in the lower left panel of Figure 2. The lower right panel shows an image of the difference in density between the target and morphed source. Before the morphing, the sum squared difference between the source and the target was about 1.9×10^8 , and the corresponding value for the difference between the morphed source and the target was about 2.8×10^7 , a reduction of about 85%. Figure 3 shows the corresponding deformation applied to set of grid lines in the x - y plane.

Experiment 1 is more analogous to the problem of CME reconstruction from only three points of view. The thick solid green line in Figure 4 shows a single vector containing projections (parallel line integrals) of the run B (target) density at three different angles (29° , 89° , and 149° from the vertical direction). Each projection has 362 elements (which adequately samples projections of 256×256 pixel object), thus, the x -axis in the figure shows goes from 1 to 1086. This list of

1086 numbers forms the vector y . The dashed red line shows the same projections of run A's density (source), $\mathbf{A}\rho_0$. The thin black line shows the projections of the morphed source, i.e., $\mathbf{A}\rho_\varphi = \mathbf{A}\mathbf{T}_\varphi\psi$. Before morphing, the initial data misfit was, $\sqrt{(\|y - \mathbf{A}\rho_0\|^2)/(\|y\|^2)} \approx 0.26$, and after morphing the misfit value improved to $\sqrt{(\|y - \mathbf{A}\rho_\varphi\|^2)/(\|y\|^2)} \approx 0.06$. Before the morphing, the sum squared difference between the source and the target was again about 1.9×10^8 , and the corresponding value for the difference between the morphed source and the target was about 9.3×10^7 , a reduction of over 50%. Figure 5 shows the corresponding deformation applied to set of grid lines in the x - y plane, which are not as distorted as that shown in Figure 3. This is because the deformation was determined with much less information.

Figure 6 is similar to Figure 2, but it shows the effect of vastly reduced information in Experiment 1. Unlike in Figure 2, the hook-like feature on the left part of the source, which is not present in the target, is not greatly reduced in size. Other more subtle features that were adjusted in Experiment 0 were not adjusted in Experiment 1. However, similarly to Experiment 0, the morphing greatly increases the size of the cavity-type structure to better match the target.

Experiment 2 was the same as Experiment 1, except λ in Equation (14) was set to 0, which removes the Jacobian constraint. This allowed the optimization algorithm to find a solution φ that made the value of $\mathcal{D}(y, y_\varphi)$ close to 0. The resulting deformation grid is shown in Figure 7, where many obvious manifestations of $|J_\varphi| < 0$ (making the deformation non-diffeomorphic) can be seen where grid lines cross upon themselves and so on. Figure 8 shows the resulting morphed source, and it manifests a considerable amount of undesirable structure including several islands and a large plume in the lower left of the image. Compared with Experiment 1, this example helps to illustrate the power of the simple geometrical constraint $|J_\varphi| > 0$ to regularize the inversion problem.

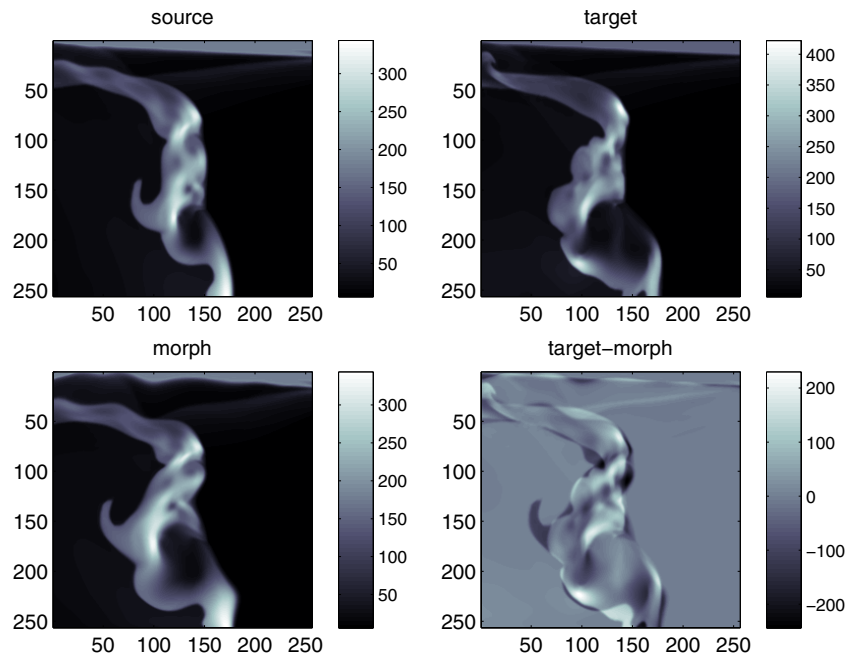


Figure 6. Results from Experiment 1. Upper left: the shock density in run A (source). Upper right: the shock density in run B. Lower right: density difference of target–morphed source. Lower left: density of morphed source.

(A color version of this figure is available in the online journal.)

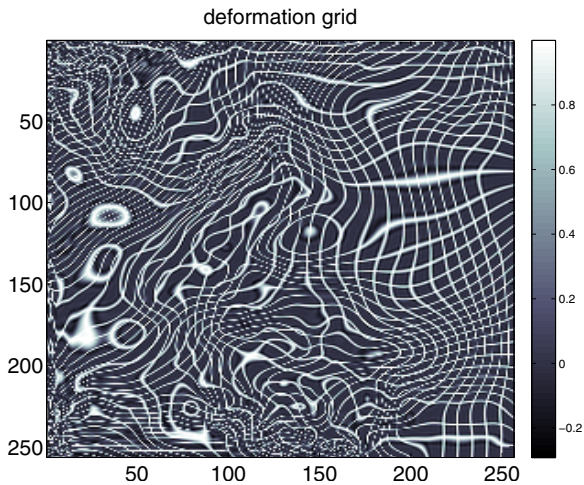


Figure 7. Deformation grid found for Experiment 2.

(A color version of this figure is available in the online journal.)

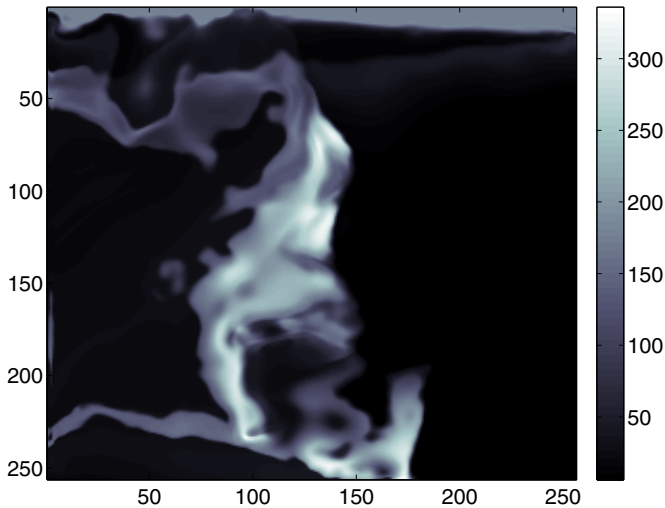


Figure 8. Morphed source for Experiment 2.

(A color version of this figure is available in the online journal.)

4. DISCUSSION

This paper has addressed the challenging problem of CME reconstruction from coronagraph images at three viewpoints. The challenge arises from the fact that three viewpoints are far too little information to perform classical tomographic inversion, in which one must solve for the value of a high-dimensional model that specifies the density at every point. The other option that presently exists for CME reconstruction is to create a simple geometrical model that is a function of less than about a dozen free parameters, and to use the coronagraph observations to determine those parameters (Thernisien 2011). On the other hand, the latest generation of MHD simulations of CMEs produce realistic 3D structures that are far too complex to be captured a parametric model (Manchester et al. 2008). While data assimilation is the classical avenue for combining numerical hydrodynamics and physical measurements of a system, the problem of having only line integrals of the density and no measurements about other state variables seems rather daunting and has not been addressed. The method suggested here is an attempt to utilize image registration technology to morph the numerical hydrodynamics simulation to improve

agreement with the data. The power of the method comes from the fact that enforcing a simple differentiability constraint is highly constraining regularizer that compensates for the lack of information in the data.

The application of this method to multi-coronagraph observations of CMEs has a number of open questions.

1. What are good choices for the data misfit function $\mathcal{D}(y, y_\varphi)$?
2. What are the effects of utilizing a CME simulation that is significantly incorrect?
3. Will the constraint $|J_\varphi| > 0$ work as well in 3D as it seems to in 2D? If not, are there other useful constraints that can be placed on the transformation?
4. How can the method be extended to four dimensions (4D), i.e., three spatial dimensions plus time?
5. Once the reconstructions have been achieved, how can one make scientific use of them?

The extension to 4D is discussed in Appendix A. In a partial answer to the last question, one could imagine creating several CME simulations with different initial configurations and look for commonality between the morphings. Or perhaps one might look at the deformation field itself to get clues as to how to improve the simulation. Clearly, it would be important to perform this exercise systematically over a number of CMEs in order to gain physical insight.

The author thanks Alfred H. Hero, Eric Myra, Ward B. Manchester IV, Paul Shearer, Jeff Fessler, and most especially Se Young Chun for their help. This research was supported by the NSF CDI program, award 1027192.

APPENDIX A

EXTENSION TO FOUR DIMENSIONS

Thus far, this paper has only considered the deformation field $\mathbf{q}(\mathbf{r}; \varphi)$ at a given instant in time t . There may significant advantage to considering a time series of data $y(t)$, and a time-dependent deformation field, $\mathbf{q}(\mathbf{r}, t; \varphi)$ (with a corresponding redefinition of φ), since additional constraints become available. This section discusses several possibilities for spatio-temporal constraints.

One straightforward way to create the time-dependent deformation field would be to add a temporal spline factor to Equation (2). For example, the x -component of deformation may be given by the basis expansion:

$$q_x(\mathbf{r}, t; \varphi_x) = \sum_{ijkl} \varphi_x^{ijkl} \beta\left(\frac{t}{\tau} - l\right) \beta\left(\frac{x}{s} - i\right) \times \beta\left(\frac{y}{s} - j\right) \beta\left(\frac{z}{s} - k\right), \quad (\text{A1})$$

where τ is the temporal grid spacing. Of course, the spatial Jacobian constraints, discussed in Section 2.1, must apply at each time t , and therefore are independent of the index l in Equation (A1). At the most simplistic level, one expects the deformation field to vary smoothly in time, so one can restrict the solution space (regularize) by penalizing its time derivatives in a cost function.

Assuming that at any time t' the deformation $\mathbf{q}(\mathbf{r}, t'; \varphi)$ is a spatial diffeomorphism, enforcing smoothness (e.g., with the temporal spline in Equation (A1)) in the temporal dimension will also lead to a 4D diffeomorphism, as the transformation will be smooth and bijective.

Another way to restrict the solution space by requiring the deformation to be consistent with the principle of mass conservation. Note that one can require that the deformation does not alter the mass locally simply by requiring $|J_\varphi| = 1$ (volume preserving deformation), but this would lead to very restrictive solution space, indeed. However, less restrictive is to assume that $\partial \mathbf{q}(\mathbf{r}, t; \varphi)/\partial t$ can be interpreted as an empirical correction to the velocity from the MHD simulation $\mathbf{v}_0(\mathbf{r}, t)$. Thus, the corrected velocity is

$$\mathbf{v}(\mathbf{r}, t) \equiv \mathbf{v}_0(\mathbf{r}, t) + \delta \mathbf{v}(\mathbf{r}, t), \quad (\text{A2})$$

where $\delta \mathbf{v}(\mathbf{r}, t) \equiv \partial \mathbf{q}(\mathbf{r}, t; \varphi)/\partial t$. Now, away from the computational boundary zones (where a source is presumably ‘‘feeding’’ the CME eruption or letting it escape the computational domain), one can assume mass conservation:

$$0 = \frac{\partial \rho_0}{\partial t} + \nabla \cdot \left[\rho_0 \left(\mathbf{v}_0 + \frac{\partial \mathbf{q}(\mathbf{r}, t; \varphi)}{\partial t} \right) \right], \quad (\text{A3})$$

where Equation (A3) uses the fact that the morphological operation does not alter the density values, hence only the simulation density ρ_0 enters into the equation. Since the MHD simulation already satisfies the condition: $\partial \rho_0/\partial t + \nabla \cdot (\rho_0 \mathbf{v}_0) = 0$, Equation (A3) reduces to

$$0 = \nabla \cdot \left(\rho_0 \frac{\partial \mathbf{q}(\mathbf{r}, t; \varphi)}{\partial t} \right). \quad (\text{A4})$$

Equation (A4) provides a set of constraints on the time derivative of the deformation and it utilizes the simulation density ρ_0 . These derivatives can be evaluated using Equation (A1) and one could place them into an additional penalty term of the cost function. One undesirable property of enforcing constraints (A4) is the fact that the simulation density ρ_0 is only an approximation of the true (unknown) density ρ . One may consider schemes in which one also uses an improved estimate of ρ instead of ρ_0 .

APPENDIX B

MULTI-SCALE OPTIMIZATION

Multi-scale optimization is commonly used for image processing. The idea is to perform the optimization at the coarsest scale first and to perform the optimization again at successively finer scales. In the case of image registration problems, this avoids many local minima in the original problem because only the coarsest scales are calculated first. In addition, multi-scale optimization provides a ‘‘warm start’’ to each successive optimization at a finer scale, greatly speeding convergence.

Before starting the optimization process, one must first calculate the ‘‘image pyramid’’ corresponding to a multi-resolution representation of the object to be deformed, in this case, ρ_0 . In case of B-spline representations, image pyramids can be calculated very quickly (Unser 1999). The image pyramid of ρ_0 is the collection of images $\{\rho_0^J\}$, where ρ_0^J is defined as J th reduction of ρ_0 (also, $\rho_0^0 \equiv \rho_0$). Consider a 2D example in which ρ_0 is a 256×256 pixel image and reductions are by a factor of two in each dimension, then ρ_0^J is an image that is $(256/2^J) \times (256/2^J)$. In accordance with Equation (4), accompanying each image ρ_0^J is its B-spline representation ψ^J . The optimization problem in Equation (14) is to be done at a scale J . Thus, the operators \mathbf{A}^J and \mathbf{T}_φ^J are needed as well. Note that φ^J has $(3M)/(2^J p)$ components, so that the size of the optimization problem decreases

quickly with J . The \mathbf{A}^J operators need to be constructed for the specific problem so that each maps an image at scale J to \mathbb{R}^N , the data space where y is defined.

For the difference functional in Equation (13), the multi-scale optimization proceeds as follows.

1. Start at the coarsest scale $J = J_{\max}$, with initial guess $\varphi_\star^{J_{\max}} \equiv 0$.
2. Using the nonlinear conjugate gradient algorithm, solve for $\hat{\varphi}^J$, with initial guess $\varphi_{\text{start}} = \hat{\varphi}_\star^J$:

$$\hat{\varphi}^J = \underset{\varphi}{\operatorname{argmin}} \|y - \mathbf{A}^J \mathbf{T}_\varphi^J \psi^J\|^2 + \lambda \mathcal{P}(\varphi). \quad (\text{B1})$$

3. Interpolate $\hat{\varphi}^J$ to the next finer scale, call the result $\hat{\varphi}_\star^{J-1}$. This is the ‘‘warm start.’’
4. If $J > 0$, set $J \rightarrow J - 1$, and go back to step 2.

APPENDIX C

CHUN’S PENALTY

The diffeomorphism constraint is enforced via a penalty the penalty function proposed in Chun & Fessler (2009). For completeness, their formulation is presented here. First we define the function:

$$p(t; \zeta_1, \zeta_2) = \begin{cases} \frac{1}{2}(1 - \zeta_1)^2, & t < \zeta_1 \\ 0, & \zeta_1 \leq t \leq \zeta_2 \\ \frac{1}{2}(1 - \zeta_2)^2, & t > \zeta_2 \end{cases}. \quad (\text{C1})$$

Incorporating this new function into the penalty function (see Equation (2)), we have

$$\begin{aligned} \mathcal{P}(\varphi) = & \sum_{\gamma \in \{x, y, z\}} \sum_{i, j, k} [p(\varphi_\gamma^{i+1, j, k} - \varphi_\gamma^{i, j, k}; \zeta_1^{\gamma, x}, \zeta_2^{\gamma, x}) \\ & + p(\varphi_\gamma^{i, j+1, k} - \varphi_\gamma^{i, j, k}; \zeta_1^{\gamma, y}, \zeta_2^{\gamma, y}) \\ & + p(\varphi_\gamma^{i, j, k+1} - \varphi_\gamma^{i, j, k}; \zeta_1^{\gamma, z}, \zeta_2^{\gamma, z})], \end{aligned} \quad (\text{C2})$$

where $\zeta_1^{\gamma, \delta} = -mk$, in which $\delta \in \{x, y, z\}$, and $\zeta_2^{\gamma, \delta} = mk$ when $\delta \neq \gamma$, and $\zeta_2^{\gamma, \delta} = mK$ when $\delta = \gamma$. Recall that m was the spline knot spacing, the code in the examples presented here used $k = 0.495m$ and $K = 9m$.

REFERENCES

- Bemporad, A. 2011, *J. Atmos. Sol.-Terr. Phys.*, **73**, 1117
 Butala, M., Kamalabadi, F., Frazin, R., & Chen, Y. 2008, *IEEE J. Sel. Top. Signal Process.*, **2**, 755
 Chen, P. F. 2011, *Living Rev. Sol. Phys.*, **8**, 1
 Christensen, G., Rabbitt, R., & Miller, M. 1996, *IEEE Trans. Image Process.*, **5**, 1435
 Chun, S. Y., & Fessler, J. A. 2009, *IEEE J. Sel. Top. Signal Process.*, **3**, 159
 Cordero-Grande, L., Vegas-Sanchez-Ferrero, G., Casaseca-de-la Higuera, P., & Alberola-Lopez, C. 2012, *IEEE Trans. Image Process.*, **21**, 2047
 Crum, W. R., Hartkens, T., & Hill, D. L. G. 2004, *Br. J. Radiol.*, **77**, S140
 D’Amato, A. M., Ridley, A. J., & Bernstein, D. S. 2011, *Stat. Anal. Data Min.*, **4**, 446
 Evensen, G. 2007, *Data Assimilation: The Ensemble Kalman Filter* (Berlin: Springer)
 Fan, Y. 2005, *ApJ*, **630**, 543
 Frazin, R. A., Jacob, M., Manchester, W. B., IV, Morgan, H., & Wakin, M. B. 2009, *ApJ*, **695**, 636
 Frazin, R. A., & Janzen, P. 2002, *ApJ*, **570**, 408
 Frazin, R. A., Lamy, P., Llebaria, A., & Vásquez, A. M. 2010, *Sol. Phys.*, **265**, 19
 Frazin, R. A., Vásquez, A. M., Thompson, W. T., et al. 2012, *Sol. Phys.*, **280**, 273

- Howard, R. A., Moses, J. D., Vourlidas, A., et al. 2008, *Space Sci. Rev.*, **136**, 67
- Le Guyader, C., Apprato, D., & Gout, C. 2012, *IEEE Trans. Image Process.*, **21**, 1587
- Lugaz, N., Hernandez-Charpak, J. N., Roussev, I. I., et al. 2010, *ApJ*, **715**, 493
- Lynch, B. J., Antiochos, S. K., Li, Y., Luhmann, J. G., & DeVore, C. R. 2009, *ApJ*, **697**, 1918
- Manchester, W. B., IV, van der Holst, B., Tóth, G., & Gombosi, T. I. 2012, *ApJ*, **756**, 81
- Manchester, W. B., IV, Vourlidas, A., Tóth, G., et al. 2008, *ApJ*, **684**, 1448
- Mierla, M., Inhester, B., Antunes, A., et al. 2010, *Ann. Geophys.*, **28**, 203
- Myra, E. S., Fryxell, B., Grosskopf, M. J., et al. 2012, High Energy Density Phys., submitted
- Roussev, I. I., Gaalgaard, K., Downs, C., et al. 2012, *Nat. Phys.*, **8**, 845
- Schwenn, R. 2006, *Living Rev. Sol. Phys.*, **3**, 2
- Thernisien, A. 2011, *ApJS*, **194**, 33
- Thernisien, A., Vourlidas, A., & Howard, R. A. 2011, *J. Atmos. Sol.-Terr. Phys.*, **73**, 1156
- Unser, M. 1999, *IEEE Signal Process. Mag.*, **16**, 22
- van de Hulst, H. C. 1950, *Bull. Astron. Inst. Neth.*, **11**, 135
- van der Holst, B., Manchester, W., IV, Sokolov, I. V., et al. 2009, *ApJ*, **693**, 1178
- van der Holst, B., Tóth, G., Sokolov, I. V., et al. 2011, *ApJS*, **194**, 23
- Wachowiak, M., Smolikova, R., Zheng, Y., Zurada, J., & Elmaghraby, A. 2004, *IEEE Trans. Evol. Comput.*, **8**, 289
- Wu, C.-C., Dryer, M., Wu, S. T., et al. 2011, *J. Geophys. Res. (Space Phys.)*, **116**, 12103