

Ghost Cancellation Algorithms for MRI Images

AVIDEH ZAKHOR

Abstract—Reconstructed MRI images must ideally be real and positive, since they correspond to density distributions of objects, which by definition are real, positive quantities. In practice however, most reconstructed images become complex due to a shift in data from the nominal origin in k space. In some situations such as constant readout gradient, these phase shifts do not affect the magnitude of the reconstructed images, and can therefore be easily determined and eliminated. However, when the readout gradient is sinusoidal and the frequency plane is scanned with data reversal on alternate lines, time delays between the start of data acquisition and the start of the readout pulse become different for even and odd lines, and result in a ghost separated by half the image size. In this paper, we describe ghost cancellation algorithms for restoration of MRI images in medical applications. Our approach is to model the effect of the time delays and the asymmetry of the sinusoidal readout gradient for even and odd lines by two phase functions relating the actual object density to even and odd parts of the observed image. We then exploit *a priori* information about the phase functions in order to estimate the true object density. Examples of application of this ghost cancellation approach to liver and heart images will be presented.

I. INTRODUCTION

UNDER IDEAL conditions, reconstructed MRI images must be positive and real. This is because samples of the observed MRI signal in time correspond to the Fourier transform of the object density distribution, which by definition, is a real, positive quantity. In practice however, the shift in data from nominal origin in k space results in complex images even with the most basic scanning patterns in the Fourier domain such as row by row or column by column scanning. For instance, if the readout gradient is constant and the signal is sampled uniformly in time, delay in time data translates into a linear phase shift in the reconstructed image. These phase shifts can then be easily determined and eliminated since they do not effect the magnitude of the reconstructed image.

In recent years, an area of development in echo-planar MRI imaging has been the sampling of Fourier space along trajectories other than the parallel strips [1], [2]. This can be explained by considering that the signal read out at time t corresponds to a sample of the Fourier transform of the object density distribution at spatial frequency coordinates:

$$k_x = \frac{\gamma}{2\pi} \int_0^t G_x(t') dt' \quad (1)$$

Manuscript received July 17, 1989; revised January 20, 1990. This work was supported in part by Advanced NMR, Inc., Woburn, MA.

The author is with the Department of Electrical Engineering, Computer Science, University of California at Berkeley, Berkeley, CA 94720.

IEEE Log Number 9035902.

$$k_y = \frac{\gamma}{2\pi} \int_0^t G_y(t') dt' \quad (2)$$

where γ is the gyromagnetic ratio, and G_x and G_y are gradients along x and y directions, respectively. The flexibility provided by above equations imply that different gradient waveforms correspond to different sampling patterns in the Fourier domain. Examples of such patterns include spiral and square spiral scans [2].

Experimental results have shown that for scanning patterns with data reversal on alternate k_x lines, time delays between the start of data acquisition and the start of the readout pulse are different for even and odd lines. The effect of this on the image manifests itself as a ghost separated by half the image size along the y direction. An example of a scanning pattern resulting in ghosts in y direction is shown in Fig. 1(a). As seen, the x gradient, G_x , is a positive constant on even horizontal scan lines and negative constant on the odd ones. Under these conditions, if the MRI signal is sampled uniformly in time, the ghost image can be entirely removed by a first-order phase difference between odd and even lines [3]. However, when the readout gradient, G_x is sinusoidal, with the positive half of the sinusoid corresponding to even lines and the negative half corresponding to odd lines, the difference between even and odd line delays, together with asymmetry of the sinusoidal gradient result in ghosted images which cannot be corrected with a first order phase function. In this paper, we describe ghost correction algorithms for restoration of MRI images resulting from such gradient waveforms. Specifically, we consider a scanning pattern shown in Fig. 1(b) in which the sinusoidal gradients are applied along the x direction and constant gradients along the y direction. Optimal sampling and filtering of the signal resulting from these gradients are discussed in a separate paper [4].

Our approach is to model the imperfections such as sinusoidal asymmetry and time delay differences between even and odd lines by multiplying even and odd parts of the MRI image by two separate phase functions $\phi(n_1, n_2)$ and $\theta(n_1, n_2)$. Specifically, if $x(n_1, n_2)$ denotes the true density distribution of the object under consideration, and $Y(n_1, n_2)$ denotes the observed ghosted image, then symmetric and antisymmetric parts of the observed image,

¹The same argument can be made about k_y .

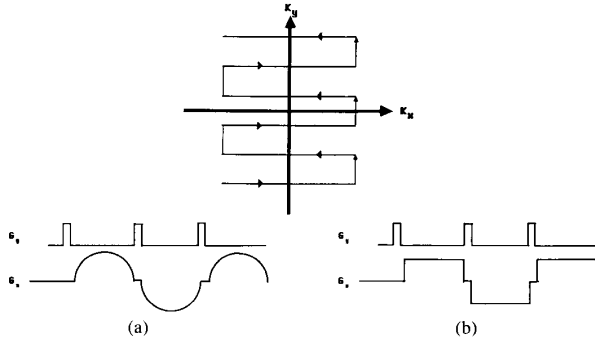


Fig. 1. k trajectory for constant positive G_x and data reversal on alternate k_x lines with a) constant positive G_x on even lines and constant negative G_x on odd lines; b) sinusoidal G_x with the positive half on even lines and negative half on odd lines.

$Y_{\text{even}}(n_1, n_2)$ and $Y_{\text{odd}}(n_1, n_2)$, can be modeled as:

$$\begin{aligned} Y_{\text{even}}(n_1, n_2) &\equiv Y(n_1, n_2) + Y\left(n_1, n_2 + \frac{N}{2}\right) \\ &= x(n_1, n_2)e^{j\phi(n_1, n_2)} + x\left(n_1, n_2 + \frac{N}{2}\right) \\ &\quad \cdot e^{j\phi(n_1, n_2 + N/2)} \\ & \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2} \quad (3) \end{aligned}$$

$$\begin{aligned} Y_{\text{odd}}(n_1, n_2) &\equiv Y(n_1, n_2) - Y\left(n_1, n_2 + \frac{N}{2}\right) \\ &= x(n_1, n_2)e^{j\theta(n_1, n_2)} - x\left(n_1, n_2 + \frac{N}{2}\right) \\ &\quad \cdot e^{j\theta(n_1, n_2 + N/2)} \\ & \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2} \quad (4) \end{aligned}$$

where N is the number of echoes and N_s is the number of samples in each echo. Having modeled the ghosted image, our objective can then be stated as estimation of the true object density distribution $x(n_1, n_2)$, from the observed ghosted image $Y(n_1, n_2)$. To this end, we exploit *a priori* information about the shape of θ and ϕ , together with a particular subset of pixels in the observed ghosted image in order to estimate θ and ϕ which are ultimately used in (3) and (4) to estimate the true object density.

The outline of the remaining part of this paper is as follows: We begin with the look up table approach to ghost correction in Section II. As we will see, although this approach does not result in complete ghost cancellation, it provides insight for derivation of the ghost cancellation algorithm of Section III. Section IV includes examples, and finally, Section V includes summary and conclusions.

II. "LOOK UP" TABLE APPROACH TO GHOST REMOVAL

In this section, we describe the look-up table approach to ghost removal. The basic idea behind this approach is to experimentally determine the difference between phase

functions $\phi(n_1, n_2)$ and $\theta(n_1, n_2)$ and use it in (3) and (4) to correct for ghosts. Specifically, from (3) and (4), it is clear that if the phase functions $\phi(n_1, n_2)$ and $\theta(n_1, n_2)$ are known for all values of n_1 and n_2 , then $x(n_1, n_2)$ and $x(n_1, n_2 + N/2)$ can be determined from $Y_{\text{even}}(n_1, n_2)$ and $Y_{\text{odd}}(n_1, n_2)$ by solving a 2×2 linear system of equations. In practice, the difference between $\phi(n_1, n_2)$ and $\theta(n_1, n_2)$ can be determined experimentally by placing a test object in the upper and lower half of the field of view (FOV) and measuring the phase difference between even and odd parts of the resulting images. When the object is in the upper half of the FOV, by definition we have:

$$x\left(n_1, n_2 + \frac{N}{2}\right) = 0 \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2}.$$

Substituting this into (3) and (4), we get:

$$\begin{aligned} Y_{\text{even}}(n_1, n_2) &= x(n_1, n_2)e^{j\phi(n_1, n_2)} \\ & \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2} \quad (5) \end{aligned}$$

$$\begin{aligned} Y_{\text{odd}}(n_1, n_2) &= x(n_1, n_2)e^{j\theta(n_1, n_2)} \\ & \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2}. \quad (6) \end{aligned}$$

The phase difference between $Y_{\text{even}}(n_1, n_2)$ and $Y_{\text{odd}}(n_1, n_2)$ can be used to obtain

$$\begin{aligned} \Delta(n_1, n_2) &\equiv \phi(n_1, n_2) - \theta(n_1, n_2) \\ &= \text{phase}(Y_{\text{even}}(n_1, n_2)) - \text{phase}(Y_{\text{odd}}(n_1, n_2)) \\ & \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2} \quad (7) \end{aligned}$$

Similarly, by placing the object in the lower half of the FOV, we have:

$$\begin{aligned} x(n_1, n_2) &= 0 \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2} \\ & \quad (8) \end{aligned}$$

$$\begin{aligned} Y_{\text{even}}(n_1, n_2) &= x\left(n_1, n_2 + \frac{N}{2}\right)e^{j\phi(n_1, n_2 + N/2)} \\ & \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2} \\ & \quad (9) \end{aligned}$$

$$\begin{aligned} Y_{\text{odd}}(n_1, n_2) &= -x\left(n_1, n_2 + \frac{N}{2}\right)e^{j\theta(n_1, n_2 + N/2)} \\ & \quad 0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2}. \end{aligned}$$

The phase difference between $Y_{\text{even}}(n_1, n_2)$ and $Y_{\text{odd}}(n_1, n_2)$ can be used to obtain

$$\begin{aligned} \Delta\left(n_1, n_2 + \frac{N}{2}\right) &\equiv \phi\left(n_1, n_2 + \frac{N}{2}\right) - \theta\left(n_1, n_2 + \frac{N}{2}\right) \\ &= \text{phase}(Y_{\text{even}}(n_1, n_2)) - \text{phase} \\ &\quad \cdot (Y_{\text{odd}}(n_1, n_2)) + \pi \\ &0 \leq n_1 < N_s \quad 0 \leq n_2 < \frac{N}{2}. \end{aligned} \quad (10)$$

Letting

$$A(n_1, n_2) = x(n_1, n_2)e^{j\theta(n_1, n_2)} \quad (11)$$

$$B(n_1, n_2) = x\left(n_1, n_2 + \frac{N}{2}\right)e^{j\theta(n_1, n_2 + N/2)} \quad (12)$$

in (3) and (4), we get:

$$\begin{aligned} Y_{\text{even}}(n_1, n_2) &= A(n_1, n_2)e^{j\Delta(n_1, n_2)} + B(n_1, n_2) \\ &\quad \cdot e^{j\Delta(n_1, n_2 + N/2)} \\ Y_{\text{odd}}(n_1, n_2) &= A(n_1, n_2) - B(n_1, n_2) \end{aligned} \quad (13)$$

Thus, experimental values of $\Delta(n_1, n_2)$ and $\Delta(n_1, n_2 + N/2)$ can be used to determine $A(n_1, n_2)$ and $B(n_1, n_2)$ by solving the linear system of equations in (13). Once A and B are determined, their magnitudes can be used to find $x(n_1, n_2)$ and $x(n_1, n_2 + N/2)$, respectively.

We have found experimentally, that there are two major drawbacks with the above approach. The first drawback has to do with the fact that the phase difference function depends on physical parameters for the MRI experiment. Some of these parameters include strength of x , y , and z gradients, static magnetic field, or RF. Therefore, to be able to apply this method successfully, we need to have a different look up table for different experimental set ups. The second drawback has to do with the fact that phase difference function, $\Delta(n_1, n_2)$, is somewhat object dependent. More specifically, although the general shape of $\Delta(n_1, n_2)$ does not vary drastically from one object to the next, the change is large enough to introduce considerable ghosting. The third drawback of the above approach has to do with the fact that obtaining the phase difference function $\Delta(n_1, n_2)$ of a test object for all values of n_1 and n_2 is a nontrivial task from an experimental point of view. This has to do with factors such as susceptibility effects.

In short, we have found the performance of above scheme to be inadequate for most medical images. However, as we will see, the approximate shape of the phase difference function obtained via the above strategy provides useful insight for development of the ghost correction algorithm of the next section.

III. AUTOMATIC GHOST CORRECTION (AGC) ALGORITHM

In deriving the AGC algorithm, we take advantage of the approximate shape of the phase difference function, which as we saw in the previous section, can be obtained experimentally for test objects. Experiments indicate that variation of the function $\Delta(n_1, n_2)$ is considerably smaller along n_2 than n_1 direction. In fact for fixed n_1 , variation along n_2 is symmetric about $n_2 = N/2$, and $\Delta(n_1, n_2)$ can be closely approximated by a piecewise linear function of the form:

$$\begin{aligned} \Delta(n_1, n_2) &= \begin{cases} \alpha(n_1) + \beta(n_1)n_2 & 0 \leq n_2 \leq \frac{N}{2} \\ \alpha(n_1) + \beta(n_1)N - \beta(n_1)n_2 & \frac{N}{2} \leq n_2 < N. \end{cases} \end{aligned} \quad (14)$$

From a physics point of view, (14) can be justified by considering the two main reasons behind the ghost formation process. One is misadjustment of the center of k space, which for the constant gradient case can be corrected by first order phase correction as long as all echoes have the same origin offset. The other is different amounts of magnet inhomogeneity phase errors between echoes which is qualitatively parabolic before shimming, and unpredictable afterwards. Thus, polynomial approximation of the phase difference function as shown in (14) is reasonable.

The AGC algorithm takes advantage of this approximation by estimating $\alpha(n_1)$ and $\beta(n_1)$ for each column of the data, i.e., $n_1 = 0, \dots, N_s - 1$. Specifically, the algorithm consists of the following steps:

1) Take 2-D inverse Fourier transform of the time data to obtain the ghosted image $Y(n_1, n_2)$.

2) Determine signal energy for each column of the data by computing

$$E(n_1) = \sum_{n_2=0}^{N-1} |Y(n_1, n_2)|^2. \quad (15)$$

3) Discard columns whose signal energy level is below a fixed threshold, snr .

4) Let S denote the set of indices of the columns whose signal energy level is larger than the threshold snr . Estimate $\alpha(n_1)$ and $\beta(n_1)$ for all $n_1 \in S$. The estimation procedure will be discussed at length later.

5) Use $\alpha(n_1)$ and $\beta(n_1)$ in (14) to find $\Delta(n_1, n_2)$ for $0 \leq n_2 < N$.

6) Use $\Delta(n_1, n_2)$ in (13) to find $A(n_1, n_2)$ and $B(n_1, n_2)$ for $0 \leq n_2 < N$. From (11) and (12), the true object density distribution at (n_1, n_2) and $(n_1, n_2 + N/2)$ are found by taking magnitudes of $A(n_1, n_2)$ and $B(n_1, n_2)$, respectively.

We will now describe the forth step of the algorithm in more detail.

A. Parametric Estimation of the Phase Difference Function

Recall from Section II that the phase difference function $\Delta(n_1, n_2)$, can be found experimentally by placing a test object in the lower or upper half of the FOV. Specifically, the ghost of a test object at location (n_1, n_2) with $0 \leq n_2 < N/2$ appears at $(n_1, n_2 + N/2)$, and vice versa. Thus, the location of the ghost of a pixel at (n_1, n_2) is $(n_1, (n_2 + N/2) \bmod N)$.

In general, if an object only fills out half of the FOV, its ghost does not overlap with itself in the reconstructed image. Under these conditions, the phase different function associated with the object and the particular experimental set up can be determined empirically for regions of the reconstructed image which correspond to the object rather than its ghost.² On the other hand, when an object fills out more than half of the FOV, its phase difference function can only be determined for pixels whose ghosts are not superimposed on pixels corresponding to other parts of the object.

We can exploit this information about $\Delta(n_1, n_2)$ to find the parameters α and β as defined by (14). We define pixels which correspond to bright parts of the object³ (rather than low energy part of the object or empty space in the FOV), and whose ghost locations correspond to either empty space in the FOV or parts of the object with very little or no energy, to be "ghosting pixels." Specifically, if the pixel at location (n_1, n_2) is a ghosting one, then by definition,

1) It corresponds to a high energy point in the object. Therefore:

$$|x(n_1, n_2)| \gg 0.$$

2) The pixel at location $[n_1, (n_2 + N/2) \bmod N]$ corresponds to a low energy part of an object or empty space in the FOV. That is

$$\left| x\left(n_1, \left(n_2 + \frac{N}{2}\right) \bmod N\right) \right| \approx 0.$$

The forth step of the AGC algorithm, derives the parameters associated with the n_1 th column in two steps. Specifically, it first finds the value of the phase difference function for all the "ghosting pixels" of the column, and then solves an overdetermined linear system of equations to find linear least square estimates of $\alpha(n_1)$ and $\beta(n_1)$. At this point, the key question which remains to be answered is the way ghosting pixels are detected, or "rec-

ognized." Our approach has been to use two criteria for classifying pixels as ghosting ones. The first criterion is a direct consequence of (13). It takes advantage of the fact that the magnitude of the even and parts of a ghosting pixel at location (n_1, n_2) are identical. Specifically, if $Y(n_1, n_2)$ is a ghosting pixel, from (13) we get:

$$\begin{aligned} |Y_{\text{even}}(n_1, n_2)| &= |Y_{\text{odd}}(n_1, n_2)| \\ &= |A(n_1, n_2)| = x(n_1, n_2). \end{aligned} \quad (16)$$

Thus, if the ratio between the magnitude of even and odd parts of the pixel at location (n_1, n_2) are equal or somewhat close to each other, then $Y(n_1, n_2)$ can be classified as a ghosting pixel. In the actual implementation of the AGC algorithm, if the ratio between the magnitudes of even and odd parts of a pixel are in the range $[1/\text{eoratio}, \text{eoratio}]$, then the pixel is classified as a ghosting one, where eoratio stands for "even or odd ratio." It is important to note that under this criterion, the conditions for $Y(n_1, n_2)$ and $Y(n_1, n_2 + N/2)$ to qualify as ghosting pixels are identical. Thus, if (16) is exactly or approximately satisfied, then there is an ambiguity as to which pixel is the ghosting one.⁴ As we will see, the second criteria for ghosting pixel detection will resolve this ambiguity.

The second criteria for detecting ghosting pixels takes advantage of its definition. To describe this condition, let us rewrite (3) and (4) in the following way:

$$\begin{aligned} Y(n_1, n_2) &\equiv \frac{1}{2} [Y_{\text{even}}(n_1, n_2) + Y_{\text{odd}}(n_1, n_2)] \\ &\cdot \frac{x(n_1, n_2)}{2} [e^{j\phi(n_1, n_2)} + e^{j\theta(n_1, n_2)}] \\ &+ \frac{x\left(n_1, n_2 + \frac{N}{2}\right)}{2} \\ &\cdot [e^{j\phi(n_1, n_2 + N/2)} - e^{j\theta(n_1, n_2 + N/2)}] \end{aligned} \quad (17)$$

$$\begin{aligned} Y\left(n_1, n_2 + \frac{N}{2}\right) &\equiv \frac{1}{2} [Y_{\text{even}}(n_1, n_2) + Y_{\text{odd}}(n_1, n_2)] \\ &\cdot \frac{x(n_1, n_2)}{2} [e^{j\phi(n_1, n_2)} + e^{j\theta(n_1, n_2)}] \\ &+ \frac{x\left(n_1, n_2 + \frac{N}{2}\right)}{2} \\ &\cdot [e^{j\phi(n_1, n_2 + N/2)} - e^{j\theta(n_1, n_2 + N/2)}]. \end{aligned} \quad (18)$$

²Note that two special cases of this were discussed in Section II. These cases correspond to the object being in the lower or upper half of the FOV.

³Bright parts of an object are defined to be parts of the object whose image has large values of intensity. Similarly, the image of low energy parts of an object have small values of intensity.

⁴Note that by definition, $Y(n_1, n_2)$ and $Y(n_1, (n_2 + N/2) \bmod N)$ cannot both be ghosting pixels.

As we expect, if

$$\theta(n_1, n_2) = \phi(n_1, n_2)$$

or equivalently

$$\Delta(n_1, n_2) = 0$$

the observed image, $Y(n_1, n_2)$, becomes ghost free and is identical to the true object density function $x(n_1, n_2)$. Recall that if the pixel at location (n_1, n_2) is a ghosting one, then by definition, the magnitude of $x(n_1, n_2)$ must be large (i.e., not at the noise level) and the magnitude of $x(n_1, n_2 + N/2)$ must be very small (i.e., at the noise level). From (17) and (18) we conclude that if the difference between $\theta(n_1, n_2)$ and $\phi(n_1, n_2)$ is small (or equivalently $|\Delta(n_1, n_2)|$ is small), then a ghosting pixel at (n_1, n_2) results in a large value of the following ratio:

$$\text{Ratio}(n_1, n_2) = \frac{|Y(n_1, n_2)|}{\left|Y\left(n_1, n_2 + \frac{N}{2}\right)\right|}. \quad (19)$$

Specifically as $|\Delta(n_1, n_2)|$ changes from 0 to π , the above ratio changes from ∞ to 0. We have found experimentally, that $|\Delta(n_1, n_2)|$ is smaller for columns closer to the center of the magnet (i.e., around $n_1 = N_s/2$). This implies that, the ratio of (19) becomes larger as the index of the column under investigation becomes closer to $N_s/2$. Thus, the second criterion for detecting ghosting pixels of the n_1 th column consists of

- 1) Computing the quantity shown in (19) for each pixel.
- 2) Comparing this ratio to a fixed threshold associated with that column.

Clearly, this threshold is column dependent and becomes larger as the indices of the columns become closer to $N_s/2$. In the AGC algorithm, we have chosen the threshold for the n_1 th column:

- to be a parameter denoted by the variable "threshold," for $n_1 = N_s/2$, i.e., the column corresponding to the center of the magnet.
- to decrease linearly with n_1 for $|n_1 - N_s/2| < 15^5$, i.e., for columns in the neighborhood of the center of the magnet.
- to be constant for $|n_1 - N_s/2| > 15$, i.e., for columns far away from the center of the magnet.

To summarize, the first ghost detection criterion checks the ratio between the magnitudes of even and odd parts of the pixel at location (n_1, n_2) . If this ratio is close to one, then either $Y(n_1, n_2)$ or $Y(n_1, (n_2 + N/2) \bmod N)$ are classified as ghosting pixels. To resolve this ambiguity

⁵Clearly, 15 can be replaced by another input parameter for greater flexibility.

and to improve the detection procedure, we use a second criterion which computes the ratio shown in (19). For columns close to the center of the magnet, large values of this ratio imply a ghosting pixel at location (n_1, n_2) . However, for columns further away from the center, the second criterion becomes more or less inconclusive, and we must find other ways to overcome the ambiguity problem of the first criterion. Our approach has been to use the *a priori* knowledge about the approximate shape of the phase difference function in order to resolve this ambiguity. Detailed experimental procedures for obtaining phase difference function was described in Section II. Note that unlike the "look up" table approach of Section II exact values of the phase difference function are not needed; In fact, the algorithm only needs to know as much about $\Delta(n_1, n_2)$ as to make binary decisions.

From the description of the AGC algorithm, it is clear that the ghosting pixel detection part is rather heuristic. To decrease the sensitivity of the algorithm to this part, and to improve the estimation of the phase difference function, we have taken few measures.

B. Improving Estimation Robustness

From classical results in estimation theory, we know that the error in estimating parameters of $\Delta(n_1, n_2)$, i.e., α and β , is reduced as the number of observations is increased [7]. In our case, the observations are the empirical value of the phase difference function for the ghosting pixels. Experimental results seem to indicate that for columns whose number of observations (or equivalently the number of ghosting pixels) is small, the error in estimating β in (14) becomes rather large. This error manifests itself as large magnitude for β , resulting in unrealistic values of the phase difference function. Since experimental results indicate that the magnitude of β is small for most columns, our strategy to overcome this problem has been to set $\beta = 0$ for columns whose number of ghosting pixels is less than a fixed integer. To make the estimation part of the algorithm more robust, we also set $\beta = 0$ when the magnitude of its estimated value exceeds a certain threshold. In effect, this corresponds to modifying the model for the phase difference function from a first order one to a zeroth order one. Model modification is a standard practice in regression analysis and is commonly used in empirical model building [5].

The second measure we have taken to improve the robustness of the algorithm has been to discard ghosting pixels whose least-square residue is too large. This is also a standard practice in regression analysis, and is routinely used to remove the effect of "bad" data, also called "rogue" observations [5]. If $i_1^{(n-1)}, i_2^{(n-1)}, \dots, i_{\text{upper}}^{(n-1)} < N/2$ and $j_1^{(n-1)}, j_2^{(n-1)}, \dots, j_{\text{lower}}^{(n-1)} \geq N/2$ denote the indices of the ghosting pixels of the n_1 th column, taking into account (14), the linear least-squares estimation of α and β consists of solving the following overdeter-



(a)



(b)

Fig. 2. (a) Unprocessed ghosted heart image; (b) Processed, ghost-free heart image obtained by the AGC algorithm with the following parameters: $snr = 5$, $eoratio = 1.5$, $threshold = 100$, $mse = 3$.



(a)



(b)

Fig. 3. (a) Unprocessed ghosted heart image; (b) Processed, ghost-free heart image obtained by the AGC algorithm with the following parameters: $snr = 5$, $eoratio = 1.5$, $threshold = 100$, $mse = 2$.

mined linear system of equations:

$$\begin{bmatrix} 1 & i_1^{(n-1)} \\ 1 & i_2^{(n-1)} \\ \vdots & \vdots \\ 1 & i_{\text{upper}}^{(n-1)} \\ 1 & N - j_1^{(n-1)} \\ 1 & N - j_2^{(n-1)} \\ \vdots & \vdots \\ 1 & N - j_{\text{lower}}^{(n-1)} \end{bmatrix} \begin{bmatrix} \alpha(n_1) \\ \beta(n_1) \end{bmatrix} = \begin{bmatrix} \Delta(n_1, i_1) \\ \Delta(n_1, i_2) \\ \vdots \\ \Delta(n_1, i_{\text{upper}}^{(n-1)}) \\ \Delta(n_1, j_1^{(n-1)}) \\ \Delta(n_1, j_2^{(n-1)}) \\ \vdots \\ \Delta(n_1, j_{\text{lower}}^{(n-1)}) \end{bmatrix} \quad (20)$$

If $\alpha(n_1)$ and $\beta(n_1)$ denote the solution of the above linear least-squares problem, the residue of the ghosting pixels at $(n_1, i_l^{(n-1)})$ and $(n_1, j_l^{(n-1)})$ are defined to be:

$$\begin{aligned} \text{res}(n_1, i_l^{(n-1)}) &\equiv \Delta(n_1, i_l^{(n-1)}) \\ &\quad - (\hat{\alpha}(n_1) + \hat{\beta}(n_1)i_l^{(n-1)}) \end{aligned}$$

$$\begin{aligned} \text{res}(n_1, j_l^{(n-1)}) &\equiv \Delta(n_1, j_l^{(n-1)}) \\ &\quad - (\hat{\alpha}(n_1) + \hat{\beta}(n_1)N - \hat{\beta}(n_1)j_l^{(n-1)}) \end{aligned} \quad (21)$$

and the mean square error at n_1 is defined to be mean squared error (n_1)

$$\begin{aligned} &\equiv \frac{1}{\text{upper} + \text{lower}} \\ &\quad \cdot \left[\sum_{l=0}^{\text{upper}} \text{res}^2(n_1, i_l^{(n-1)}) + \sum_{l=0}^{\text{lower}} \text{res}^2(n_1, j_l^{(n-1)}) \right]. \end{aligned}$$

To reduce the likelihood of false alarm (i.e., declaring nonghosting pixels as ghosting ones), our strategy has been to discard pixels for which the ratio between their residue and the mean squared error exceeds a fixed threshold. We denote this threshold by the variable "mse."

IV. EXAMPLES OF THE AGC ALGORITHM

In this section, we show a few examples of the AGC algorithm. For each example, the ghosted image, the pro-



(a)



(b)

Fig. 4. (a) Unprocessed ghosted heart image; (b) Processed, ghost-free heart image obtained by the AGC algorithm with the following parameters: $snr = 5$, $eoratio = 1.5$, $threshold = 100$, $mse = 2$.



(a)



(b)

Fig. 5. (a) Unprocessed ghosted body image; (b) Processed ghost-free body image obtained by the AGC algorithm with the following parameters: $snr = 5$, $eoratio = 1.5$, $threshold = 100$, $mse = 2$.

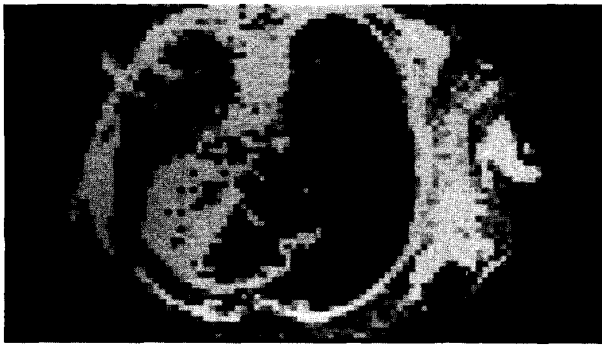
cessed ghost-free image, and the parameters used with the algorithm is specified. It is important to emphasize that none of the parameters used in the algorithm need to be measured experimentally. As described in the previous section, the essence of the algorithm is to convert a ghost cancellation problem into a parameter estimation one. In effect, the only observed quantity is the degraded, ghosted image, and the parameters which need to be estimated for the n_1 th column, i.e., $\alpha(n_1)$ and $\beta(n_1)$ are determined only on the basis of the observed ghosted image.

Similar to most image restoration applications, quantifying the performance of the proposed algorithm is not an easy task [7]. Nor is it easy to quantify the artifact generated by the algorithm. Therefore, we must rely on qualitative, subjective comparison of the degraded and processed image by human observers to illustrate the performance of the algorithm.

Figs. 2, 3, 4, and 5 show ghosted and processed heart and liver images obtained via the ghost cancellation algorithm. The parameters used for each image is shown in its caption. Clearly, in all the examples the algorithm is capable of removing the ghost successfully. In Fig. 2(a), the main ghost is to the upper left of the image and there is a small amount of ghost to the lower right. As seen, there is a great deal of correlation between the ghosts and

the ghosting pixels, in the sense that they are half the image size away from each other. The ghost in Fig. 3(a) is to the upper right of the image, and its intensity is larger than those of other examples presented here. The ghosts in Fig. 4(a) are on the upper and lower right, and upper left of the image. As seen, although the algorithm cancels most of the ghosts, it also introduces some artifacts in the lower left part of the image. Finally, Fig. 5 shows a body image, in which the outer contour of the body is causing a ghost in the upper part of the image.

As seen, the variation among the input parameters of the various examples is not large. For instance parameters snr , $threshold$, and $eoratio$ are identical for all the examples, and mse is either 2 or 3. To examine the effect of each parameter, we perturb the input parameters for the heart image shown in Fig. 3(a). Fig. 6 shows processed versions of Fig. 3(a) for different values of the parameters mse with the remaining parameters fixed at $snr = 5$, $eoratio = 1.5$, and $threshold = 100$. As seen in Fig. 6(a), for $mse = 0.1$, the ghost essentially remains uncorrected. This is because too many of the candidate ghosting pixels are discarded in the estimation procedure. On the other hand, the image corresponding to $mse = 4$ shown in Fig. 6(b) is relatively ghost free, but suffers from artifacts around the upper left side. This can be attributed to the



(a)



(b)

Fig. 6. Processed version of Fig. 3(a) with the same parameters as Fig. 3(b), except for (a) $mse = 0.1$; (b) $mse = 4$.

fact that large values of mse result in "bad" data or "rogue" observations to be included in the estimation process.

Fig. 7 shows processed versions of Fig. 3(a) for $eoratio = 1, 2, 4$ with snr , $threshold$, and mse fixed at 5, 100, and 2, respectively. As seen, for $eoratio = 1$, the ghost essentially remains uncorrected because the $eoratio$ conditions for detection of ghosting pixels becomes ineffective. In this case, the necessary condition for a pixel to be detected as a ghosting one is that the magnitude of $Y_{even}(n_1, n_2)$ becomes *exactly* identical to magnitude of $Y_{odd}(n_1, n_2)$. Since this is a stringent condition, few pixels are likely to be classified as ghosting ones. For $eoratio = 2$ and 4 on the other hand, the processed images look almost identical to the one shown in Fig. 3(b). Thus, the processed image is fairly robust with respect to $eoratio$ as long as it is not chosen too close to 1.

Fig. 8 shows the processed versions of Fig. 3(a) for $threshold = 1, 10, \text{ and } 1000$. As seen, the processed images are almost indistinguishable from one another and from Fig. 3(b). Thus, the algorithm does not seem to be too sensitive to the variable threshold as long as it is greater than or equal to 1.

Finally, we examine the effect of setting $\beta = 0$ which was discussed in the beginning of section III-B as a model modification measure. Fig. 9 shows the processed version



(a)



(b)



(c)

Fig. 7. Processed version of Fig. 3(a) with the same parameters as Fig. 3(b), except for (a) $eoratio = 1$; (b) $eoratio = 2$; (c) $eoratio = 4$.

of Fig. 3(a) with the same parameters as that of Fig. 3(b) except that β is not set to zero when the number of ghosting pixels is small, or the estimated value of β is outside its expected range. As seen, even though the ghosts are removed, there is a great deal of artifacts in the image. This has to do with the fact that the *a priori* information about range of β has not been utilized, and that small number of observations, i.e., ghosting pixels, result in noisy estimates of β .



(a)



(b)



(c)

Fig. 8. Processed version of Fig. 3(a) with the same parameters as Fig. 3(b), except for (a) $threshold = 1$; (b) $threshold = 10$; (c) $threshold = 1000$.

V. SUMMARY AND CONCLUSIONS

In this paper, we described ghost correction algorithms for MRI images obtained by scanning k space as shown in Fig. 1(b). The readout gradient, G_x , is assumed to be sinusoidal, with the positive half of the sinusoid corresponding to even horizontal lines and the negative half corresponding to odd horizontal lines in the k space.



Fig. 9. Processed version of Fig. 3(a) with the same parameters as Fig. 3(b), except that β is not set to zero under the conditions described in the first part of Section III, B.

Ghosting is a result of the difference between even and odd line delays, asymmetry of the sinusoidal gradients, and magnet inhomogeneity.

We modeled these imperfections by multiplying even and odd parts of the image by two phase functions, estimated the phase difference function, and solved a 2×2 linear system of equations to determine the true object density. In Section II, we investigated the plausibility of using the phase difference function of test objects for ghost correction of real medical objects. We found that object dependency of the phase difference function makes this an inadequate approach in most practical situations. However, approximate shape of the phase difference function obtained via this strategy was exploited to derive the ghost cancellation algorithm of Section III. This algorithm generates the ghost free version of a ghosted image by first detecting "ghosting pixels" and then using their intensities to estimate the phase difference function. As it was seen in Section IV, the algorithm results in excellent ghost cancellation for a variety of heart and liver images. Future work in this area involves applying the AGC algorithm of this paper to other scanning patterns such as square spirals.

REFERENCES

- [1] S. J. Riederer, "Recent advances in magnetic resonance imaging," *Proc. IEEE*, vol. 76, no. 9, pp. 1095-1105, Sept. 1988.
- [2] D. B. Twieg, "The k -trajectory formulation of the *nmr* imaging process with applications in analysis and synthesis of imaging methods," *Med. Phys.*, vol. 10, no. 5, pp. 610-621, Sept./Oct. 1983.
- [3] S. Gull, "Compensating for system imperfections in the ANMR experiment," Technical Report, Advanced NMR Inc., January 1987.
- [4] A. Zakhor, "Optimal sampling and reconstruction of MRI signals with time varying gradients," in *SPIE Proc. Visual Commun. and Image Process. IV*, vol. 1199, Nov. 1989, pp. 1541-1555.
- [5] Harry L. Van Trees, *Detection, Estimation, and Modulation Theory*, Vol. 1. New York: Wiley, 1968.
- [6] G. E. P. Box, *Empirical Model-Building and Response Surfaces*. New York, Wiley, 1987.
- [7] J. S. Lim, *Two-Dimensional Signal and Image Processing*. Englewood Cliffs, NJ: Prentice Hall, 1990.