Gerchberg's extrapolation algorithm in two dimensions

Robert J. Marks II

Gerchberg's 1-D iterative extrapolation algorithm for bandlimited signals is generalized to two dimensions in two distinct ways. One generalization requires knowledge of the entire spectral pupil of the bandlimited image. The second requires only knowledge of two 1-D intervals formed by the vertical and horizontal projections of the pupil. For real bandlimited images of the low-pass type, this corresponds to knowing only the maximum $x$ and $y$ spatial frequencies of the image. The utilization of information of the known portion of the image in the extrapolation process is discussed for both algorithms. The second algorithm, reformulated discretely, is placed in closed form.

I. Introduction

Gerchberg has presented a 1-D extrapolation algorithm for bandlimited signals which, in iterative form, requires only the operations of Fourier transformation and truncation. This paper presents two distinctly different generalizations of Gerchberg's algorithm to two dimensions. The first generalization requires knowledge of the shape of the spectral pupil of the image to be extrapolated. The second requires only knowledge of the horizontal and vertical projections of the pupil. This latter algorithm, when discretely implemented, can be placed in closed form. In certain instances, the rectangular matrix formed by the sampled image is simply multiplied on both sides by an appropriately parameterized Sabri and Steenaart extrapolation matrix. Implementations are presented in Ref. 5.

Much has recently been written on the instability (or ill-posedness or incompletely posedness) of extrapolation and super resolution algorithms. Three observations are in order: (1) Analysis of algorithm stability, to date, deals with the error energy of the entire extrapolated signal. Extrapolation results should be better near to where the signal is known. No allowance has yet been made for this conjecture. (2) There have been successful extrapolations of elementary signals and images utilizing unstable algorithms. Required SNRs, however, are incredibly high. (3) There do exist completely posed extrapolation algorithms even according to present stability measures.

II. Gerchberg's Algorithm

Let $u(x)$ have a spectrum

$$U(f_x) = F_x u(x) = \int_{-\infty}^{\infty} u(x) \exp(-j2\pi f_x x) dx$$

that is nonzero only over some region $Q_x \subset f_x$. Defining the gate function

$$G_{Q_x}(f_x) = \begin{cases} 1, & f_x \in Q_x, \\ 0, & f_x \notin Q_x, \end{cases}$$

we say that $u$ is bandlimited if

$$\int_{-\infty}^{\infty} G_{Q_x} d f_x < \infty.$$  

Let $T_x$ denote an interval on $x$ and define the spatial gate by

$$G_x(x) = \begin{cases} 1, & x \in T_x, \\ 0, & x \notin T_x. \end{cases}$$

The extrapolation problem is: given $uG_x$ and $Q_x$, determine $u$. The uniqueness of the result in the absence of noise is assured by well-known analyticity arguments.

Gerchberg's algorithm, in iterative form, can be summarized as follows: (1) Fourier transform $uG_x$; (2) truncate the spectrum by multiplying by $G_{Q_x}$; (3) inverse transform; (4) discard that portion where the signal is known by multiplying by $(1 - G_x)$; (5) add in the known signal $uG_x$; and (6) Fourier transform, go to step 2, and repeat. From step 5, the $N$th estimate can be written as $u_N = uG_x + H_x u_{N-1}$, where $u_0 = uG_x, H_x = (1 - G_x)B_x$, and the bandlimiting operator is defined by $B_x = F_x^{-1}G_{Q_x}F_x$. The convergence of $u_N$ to $u$ as $N$ tends to infinity has been proven in three distinct ways.
Ill. Generalization to Two Dimensions

Let \( u(x,y) \) have a spectrum \( U(f_x, f_y) = \mathcal{F}_u(x, y) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u(x, y) \exp[-j2\pi(f_x x + f_y y)] dx dy. \)

Let \( \Omega \) denote that region where \( U \) is nonzero. With reference to Fig. 1(d), let \( Q_x \) and \( Q_y \) denote the respective horizontal and vertical projections of \( \Omega \). The image \( u \) is said to be bandlimited if

\[
\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G_{Q_x} G_{Q_y} df_x df_y < \infty,
\]

where \( G_{Q_x} \) is defined analogously to \( G_{Q_y} \) in Eq. (1).

Let \( u \) be known within a region \( T \). Define the corresponding aperture by

\[
G_T = \begin{cases} 
1; & (x, y) \in T, \\
0; & (x, y) \notin T.
\end{cases}
\]

Similarly, we define the spectral pupil as

\[
G_{f} = \begin{cases} 
1; & (f_x, f_y) \in \Omega, \\
0; & (f_x, f_y) \notin \Omega.
\end{cases}
\]

Then, in iterative form, Gerchberg’s extrapolation algorithm can be straightforwardly generalized to two dimensions as follows: (1) Fourier transform \( uG_T \); (2) multiply by spectral pupil \( G_{Q_y} \); (3) inverse transform; (4) discard the region where the image is known by multiplying by \((1 - G_T)\); (5) add in the known signal \( uG_T \); and (6) Fourier transform, go to step 2, and repeat.

From the above description, the \( N \)th estimate of \( u \) can be written from step 5 as

\[
u_N = uG_T + (1 - G_T)B_{Q_x} u_{N-1},\]

where \( u_0 = uG_T \), and

\[
H = (1 - G_T)B_{Q_x}. \tag{3}
\]

Inspired by Papoulis’s 1-D proof, proof of the convergence of \( u_N \) to \( u \) in Eq. (2) is offered in Appendix A for some nonseparable \( G_T \)’s. Implementation of this algorithm on a coherent processor has recently been proposed.\(^9\)

Note that Gerchberg’s algorithm in this form is sensitive to the spectral pupil \( \Omega \). If this area is overestimated, erroneous spectral data are introduced in each iteration. Underestimation results in deletion of spectral information. In the next section, we present an alternate extension of Gerchberg’s algorithm to two dimensions which requires knowledge of only \( \Omega_x \) and \( \Omega_y \).

IV. Alternate Generalization

Consider the case where \( T \) consists of one or more disjoint islands as pictured in Fig. 1(a). Consider the 1-D function corresponding to the horizontal slice of \( uG_T \) at \( y = y_0 \). The duration of this function is dictated by \( T \). To extrapolate the slice, however, we must also know its corresponding 1-D bandwidth interval. To determine this bandwidth interval, consider the spectrum of Fig. 1(d) and its inverse transform in \( y \) in Fig. 1(b). View the inverse transform from Fig. 1(d) to Fig. 1(b) as being performed along vertical slices. If the slice intersect \( \Omega \), we are inverse transforming a function with compact support. From the uncertainty principle of Fourier analysis, the result is a function which is bandlimited (in the 1-D sense) and is thus nowhere identically zero. If the slice does not intersect \( \Omega \), the inverse transform is, of course, zero. We thus conclude that the function in Fig. 1(b) is nonzero only within the shaded strip defined by the interval \( Q_y \). The bandwidth interval of the horizontal slice in Fig. 1(a) is, therefore, \( \Omega_x \), regardless of our choice of \( y_0 \). Generalizing, we conclude that two 1-D functions corresponding to two parallel slices of a bandlimited image have identical bandwidth intervals. Note, as shown in Fig. 1(c) for the vertical case, this interval can be disjoint.

With knowledge of the duration and bandwidth intervals of each horizontal slice, we can apply Gerchberg’s algorithm in one dimension to each horizontal slice in Fig. 1(a) and generate \( uG_y \). Then, using the bandwidth interval \( \Omega_y \), this result can be vertically extrapolated to yield \( u \) over the entire \((x, y)\) plane.

Note that, unlike the generalization in the previous section, this 2-D extrapolation scheme requires knowledge of only \( \Omega_x \) and \( \Omega_y \) (instead of \( \Omega \)). For real bandlimited images of the low-pass type, \( \Omega \) is a single centered symmetric area, and \( \Omega_x \) and \( \Omega_y \) can be determined from the maximum frequencies of the image in the \( x \) and \( y \) directions.

Mathematically, we can write the horizontal extrapolation within \( T_y \) as

\[
u_G y = \sum_{m=0}^{N} H_{m} uG_T,
\]

where

\[
H_{m} = (1 - G_T)B_{y}.
\]

Vertical extrapolation follows as

\[
B_x = F^{-1} G_{Q_x} F, \quad \eta = x, y.
\]
\[
\begin{align*}
\sum_{n=0}^{\infty} H_n^2 u G_y = \sum_{n=0}^{\infty} H_n^2 \sum_{m=0}^{\infty} H_m^2 u G_T,
\end{align*}
\]

where

\[
H_y = (1 - G_y) B_y,
\]

and \( G_y \) is the gate corresponding to \( T_y \), the vertical projection of \( T \) [see Fig. 1(a)].

V. Algorithm Comparison

A contrast between the algorithms in Secs. III and IV is desirable. Algorithm 1, presented in Sec. III, requires knowledge of the entire spectral pupil region \( \Omega \). Algorithm 2 (Sec. IV) requires only knowledge of two projections of \( \Omega \): \( \Omega_x \) and \( \Omega_y \). We are thus utilizing less information in this case and, as might be expected, will in some sense diminish algorithm effectiveness.

Consider Fig. 2 in which we wish to extrapolate \( u G_T \). Using algorithm 2, the value of the extrapolation at point \( P_1 \), which lies within \( T_y \), is determined solely from information gained from the intersection of \( u G_T \) with line \( L_1 \). Point \( P_2 \) lies in the area where we have extrapolated the horizontal extrapolation. Point \( P_3 \) is a hybrid case—formed both from information from \( u G_T \) and the horizontal extrapolation.

Thus we conclude that algorithm 2 extrapolates to a point using only 1-D slices of the original signal and/or previous extrapolations. Every point exterior to \( T \), however, is related to every point within \( T \). This observation is made clear upon inspection of point \( P_4 \) in Fig. 2. The extrapolated value at \( P_4 \) can, in principle, be determined from the intersection of any line through \( P_4 \) that intersects \( T \).

Algorithm 1, on the other hand, clearly relates each interior point to each exterior point with the price that the entire spectral region, \( \Omega \), must be known.

VI. Algorithm 2 in Iterative Form

The algorithm in Sec. IV can be placed in iterative form by rewriting Eq. (6) as

\[
\begin{align*}
u = \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} H_n^2 H_m^2 u G_T,
\end{align*}
\]

and, in the spirit of iteration, define

\[
\begin{align*}
u_N = \sum_{n=0}^{N} \sum_{m=0}^{\infty} H_n^2 H_m^2 u G_T.
\end{align*}
\]

Illustrations of the convergence of \( u_N \) to \( u \) are given in Appendix B. Note that

\[
\begin{align*}
u_N = \nu_{N-1} + v_N,
\end{align*}
\]

where

\[
\begin{align*}
u_N = \sum_{n=0}^{N} \sum_{m=0}^{\infty} H_n^2 H_m^2 u G_T.
\end{align*}
\]

Furthermore,

\[
\begin{align*}
u_N = H_n^2 u \nu_{N-1} + w_N,
\end{align*}
\]

where

\[
\begin{align*}w_N = H_n^2 u G_T.\end{align*}
\]

Equations (9)–(11) define an iterative form of Eq. (8) with initialization \( u_0 = v_0 = 0 = u G_T \).
The second form of the extrapolation matrix can be found from matrix inversion. Assuming convergence, we have

\[ E_n = \lim_{L \to \infty} E_L = (I - H_n)^{-1}. \]

Then Eq. (14) becomes

\[ u = E_n u_{GT} + \sum_{n=1}^{\infty} H_n u_{GT}, \]

Some example implementations of this closed form 2-D extrapolation scheme are given in Ref. 5.

Note that the extrapolation matrices in both Eqs. (14) and (15) are parametrized by \( T_\eta \) and \( \Omega_\eta \), \( \eta = x,y \). Since each horizontal slice of \( u_{GT} \) is known over the same interval \( T_x \) and has the same bandwidth, \( \Omega_x \), the same extrapolation matrix is used for each slice. The same is true for the vertical extrapolation. Note again that, if \( T_x = T_y \) and \( \Omega_x = \Omega_y \), then \( E_x = E_y \).

A straightforward generalization holds when \( G_T \) has a finite \( T_x \) and \( T_y \) but is not separable. For a given \( y \) within \( T_y \), we can extrapolate each horizontal slice of \( u_{GT} \) using an extrapolation matrix parametrized by the same bandwidth interval, \( \Omega_x \), and the interval corresponding to the intersection of the horizontal line at our

chosen \( y \) with \( G_T \). Once \( u_{GT} \) is extrapolated into a horizontal strip within \( T_y \), vertical extrapolation can be performed with a single extrapolation matrix parametrized by \( T_y \) and \( \Omega_y \).

The algorithm, in fact, is applicable to all \( T \) such that a single vertical or horizontal extrapolation does not fill the entire plane. Consider, for example, Fig. 3 and let \( T = T_4 \). The horizontal extrapolation would fill \( T_2 \). The vertical extrapolation would then fill \( T_1 \) and \( T_2 \).

Proof of convergence of Eq. (8) for the case where \( T = T_2 \) is contained in Appendix B.

VIII. Augmentation of Algorithm 2

Algorithm 2 is not applicable to the case where \( T \) is chosen such that a single vertical or horizontal extrapolation fills the plane. Such a case is when \( T = T_1 UT_4 \) in Fig. 3. An alternate approach is thus necessary.

Consider first the case where \( u \) is known outside a finite region as illustrated in Fig. 4. Denote the region where \( u \) is not known by \( T_c \). Let \( t_x \) and \( t_y \) be the corresponding horizontal and vertical projections.

One method of extrapolation for this case is simply to extrapolate vertically within the \( t_x \) interval. Specifically, for \( x \in t_x \),

\[ u = u_{GT} + \sum_{n=1}^{\infty} H_n u_{GT}, \]

with operator definitions in Eqs. (5) and (16). This relation can be straightforwardly placed in iterative form:

\[ u_N = u_{N-1} + \frac{1}{2}(1 - G_T) \sum_{n=1}^{\infty} H_n u_{GT}, \]

with initializations

\[ u_0 = G_t u_{GT}, \]

\[ v_0 = G_y u_{GT}, \]

\[ w_0 = G_t u_{GT}, \]

where

\[ G_t(\eta) = \begin{cases} 1 & \eta \in t_x; \ \eta = (x,y), \\ 0 & \eta \in t_x \end{cases} \]

One might conclude that better extrapolation can be gained by averaging the contributions of a larger and...
larger number of radial strips. Radial extrapolation could be similarly applied to the algorithm in Sec. VI. Each radial strip, however, requires knowledge of another projection of \( \Omega \). Thus, in the limit, we would require complete knowledge of \( \Omega \).

This work was generously supported by the National Science Foundation under grant ENG-79-08009.

Appendix A

Our purpose here is to prove the convergence of \( u_N \) to \( u \) in Eq. (2) as \( N \) tends to infinity. In all cases, we will assume \( \Omega \) to be a \( 2W_x \times 2W_y \) rectangle centered about the origin. Define

\[
G_\epsilon = \begin{cases} 1; & |\eta| \leq c/2, \\ 0; & |\eta| > c/2, \end{cases}
\]

where \((c, \eta) = (a, x)\) or \((b, \eta)\). In this case, the familiar integral equation,

\[
\lambda_r \varphi_r(\eta) = 2W_\eta \int_{-c/2}^{c/2} \varphi_r(\xi) \frac{\sin c \pi \xi}{\pi \xi} d\xi,
\]

where \((r, c, \eta) = (p, a, x)\) or \((q, b, y)\) and \( \sin c \pi \xi = \sin c x/\pi x \), has as a solution the set of prolate spheroidal wave functions parametrized by the space–bandwidth product \( 2W_\eta C \). The eigenvalues have the property that

\[
0 < \lambda_r < 1.
\]

The eigenfunctions \( \varphi_r \) in Eq. (A2) are obviously bandlimited and are thus not altered by filtering:

\[
\varphi_r(\eta) = 2W_\eta \int_{-\infty}^{\infty} \varphi_r(\xi) \frac{\sin c \pi \xi}{\pi \xi} d\xi.
\]

Equations (A2) and (A4) can be equivalently stated in operator form as

\[
B_q \varphi_r G_\epsilon = \lambda_r \varphi_r,
\]

\[
B_q \varphi_r = \varphi_r.
\]

Any bandlimited image with the specified bandwidth region, \( \Omega \), can be written as

\[
u = \sum_{pq} u_{pq} \varphi_p \varphi_q.
\]

The determination of \( u_{pq} \) from \( uG_T \) will be specified in each example to follow.

Case 1: Consider first the nonseparable case when \( u \) is known outside of an \( a \times b \) rectangle. That is, \( G_T = 1 - G_a G_b \). The expansion coefficients can then be found from

\[
u_{pq} = \frac{1}{1 - \lambda_p \lambda_q} \int_{-\infty}^{\infty} \int u G_T \varphi_p \varphi_q dxdy.
\]

Using Eqs. (A5), (A6), and (2), we can inductively show that, for \( N \geq 1 \),

\[
H^n \varphi_p \varphi_q G_T = (1 - G_T) (\lambda_p \lambda_q)^{n-1} (1 - \lambda_p \lambda_q) \varphi_p \varphi_q.
\]

Substituting Eq. (A7) into Eq. (2) thus yields

\[
u_N = uG_T + (1 - G_T) \sum_{pq} u_{pq} \varphi_p \varphi_q (1 - \lambda_p \lambda_q) \sum_{n=1}^{N} (\lambda_p \lambda_q)^{n-1}
\]

\[
= uG_T + (1 - G_T) \sum_{pq} u_{pq} \varphi_p \varphi_q [1 - (\lambda_p \lambda_q)^N].
\]

From Eq. (A3) the term \((\lambda_p \lambda_q)^N \to 0 \) as \( N \to \infty \), and Eq. (A9) becomes Eq. (A7).

Case 2: Here we choose \( G_T = 1 - G_a (1 - G_b) \). Thus, \( u \) is known everywhere but within two semi-infinite vertical strips \( T = T_1UT_3UT_4 \) in Fig. 3. The proof here is the same as above, except, instead of Eq. (A8), the expansion coefficients are determined from

\[
u_{pq} = \frac{1}{1 - \lambda_p (1 - \lambda_q)} \int_{-\infty}^{\infty} \int u G_T \varphi_p \varphi_q dxdy.
\]

Equation (A9) remains identical, except that, instead of \([1 - (\lambda_p \lambda_q)^N]\), we obtain the term \(1 - [\lambda_p (1 - \lambda_q)]^N\), which, due to Eq. (A3), also converges to unity.

Similar proof can be generated for the cases where \( G_T = G_a G_b, 1 - (1 - G_a) (1 - G_b), G_a (1 - G_b), \) etc.

Appendix B

Case 1: In this section the alternate form of Gerchberg’s iterative extrapolation algorithm in two dimensions [Eq. (8)] is proved first for the case where \( G_T = G_a G_b \). Thus, \( u = G_a G_b \). Consequently, the results of Appendix A, we can express \( u \) as in Eq. (A7), and

\[
u = uG_T + \sum_{n=1}^{N} H^n_2 uG_T + \sum_{n=1}^{N} H^n_3 uG_T + \sum_{n=1}^{N} \sum_{m=1}^{N} H^n_m H^m_2 uG_T.
\]

Equations (A9) remains identical, except that, instead of \([1 - (\lambda_p \lambda_q)^N]\), we obtain the term \(1 - [\lambda_p (1 - \lambda_q)]^N\), which, due to Eq. (A3), also converges to unity.

Case as we shall demonstrate, each of these four terms corresponds to the disjoint regions on the \( x \times y \) plane illustrated in Fig. 3. The first term, \( uG_T \), is our original image and thus exists only in region \( T = T_1 \). The second term corresponds to the horizontal extrapolation result and exists only in region \( T_2 \). The third term similarly exists only in \( T_3 \). The final term corresponds to the extrapolated extrapolation and exists only in region \( T_4 \).

We will now examine the terms individually. Using the results of Appendix A, we can express \( u \) as in Eq. (A7), where

\[
u_{pq} = \frac{1}{1 - \lambda_p \lambda_q} \int_{-\infty}^{\infty} \int u G_T \varphi_p \varphi_q dxdy.
\]

Consider first the second term in Eq. (B1). From Eqs. (A5), (A6), and (A7), we can inductively show that, for \( n \geq 1 \),

\[
H^n_2 uG_T = G_b (1 - G_a) \sum_{pq} u_{pq} (1 - \lambda_p)^{n-1} \lambda_p \varphi_p \varphi_q.
\]

The second term thus becomes

\[
\sum_{pq} H^n_2 uG_T = G_b (1 - G_a) \sum_{pq} u_{pq} \varphi_p \varphi_q (1 - (\lambda_p \lambda_q)^N).
\]

Using Eq. (A3), this expression converges to \( u \) via Eq. (A7) over region \( T_2 \) as \( N \) tends to \( \infty \). A similar analysis can be applied to show the convergence of the third term to \( u \) in region \( T_3 \).

To analyze the fourth term in Eq. (B1), we first use Eq. (B2) with \( m \) replaced by \( m \) to obtain \( H^n_2 uG \). Then applying \( H^n_2 \) we can inductively show that, for \( n > m > 0 \),
where
\[ s_{pq}(n,m) = \lambda_p \lambda_q (1 - \lambda_p)^{m-i} (1 - \lambda_q)^{n-m-1}. \]

If we can show that
\[ S_N = \sum_{n=1}^{N} \sum_{m=1}^{n} s_{pq}(n,m) \quad \text{(B4)} \]

tends to unity as \( N \to \infty \), convergence to \( u \) in \( T_4 \) is assured.

If \( \lambda_p \neq \lambda_q \), we can apply the geometric series formula to Eq. (B4) twice and obtain
\[ S_N = \frac{1}{\lambda_p - \lambda_q} \left[ (1 - \lambda_q)^N - (1 - \lambda_p)^N \right]. \]

Using Eq. (A3), we conclude that
\[ \lim_{N \to \infty} S_N = 1. \]

The occurrence of the relation \( \lambda_p = \lambda_q \) will appear when \( a = b \). Then
\[ S_N = (N + 1) (1 - \lambda_p)^N - N(1 - \lambda_p)^{N-1} + 1. \]

Due to Eq. (A3), this relation also approaches unity.

**Case 2:** A similar proof can be generated for \( G_T = G_a(1 - G_b) \). Consider again Eq. (B1). For our given \( G_T \), the first term exists only in region \( T = T_3 \) in Fig. 3. The second term will converge in region \( T_4 \), the third in \( T_1 \), and the last in \( T_2 \). The proof is identical to Case 1, except that \( 1 - G_b \) replaces \( G_b \) and \( 1 - \lambda_q \) replaces \( \lambda_q \).

Similar proofs can also be generated when a single vertical or horizontal extrapolation does not fill the entire plane. These include \( G_T = (1 - G_a)(1 - G_b) \) and \( (G_a - G_b)^2 \).