Forward-Looking Scanning Radar Superresolution Imaging Based on Second-Order Accelerated Iterative Shrinkage-Thresholding Algorithm

Wenchao Li, *Member, IEEE*, Meihua Niu, *Student Member, IEEE*, Yongchao Zhang, *Member, IEEE*, Yulin Huang, *Member, IEEE*, and Jianyu Yang, *Member, IEEE*

***Abstract*—Scanning radar can be used to obtain images of targets in forward-looking area, and has attracted much attention in many fields, such as ocean monitoring, air-to-ground attack, navigation, and so on. However, its azimuth resolution is extremely poor due to the limitation of the antenna size. In order to break through the limitation, many superresolution algorithms have been proposed, and iterative shrinkage-thresholding algorithm (ISTA) is one of the most famous methods because of its antinoise ability and simplicity. In the meantime, the slow convergence of iterative shrinkage-thresholding algorithm is also known to all. In this article, a second-order accelerated ISTA for scanning radar forward-looking superresolution imaging is proposed. In this algorithm, a prediction vector is constructed before each iteration by using the first and the second-order difference information of iteration vectors to reduce the number of iterations and get a faster convergence speed. In the end, simulations and experimental results are given to illustrate the effectiveness of the accelerated imaging algorithm.**

***Index Terms*—Accelerated imaging, azimuth resolution, iterative shrinkage-thresholding algorithm (ISTA), scanning radar, slow convergence.**

I. Introduction

MANY applications, such as terrain avoidance, autonomous landing, and precision guidance, have an urgent need for radar forward-looking imaging. However, conventional Doppler beam sharpening (DBS) and synthetic aperture radar (SAR) techniques cannot be used for forward-looking imaging due to the Doppler ambiguity and small changes in Doppler frequency [1].

In order to obtain high-resolution images in the forward-looking terrain, bistatic SAR, whose transmitter and receiver are mounted on separate platforms, is adopted in [2] and [3], and it is possible to avoid Doppler ambiguities and improve azimuth resolution. However, bistatic SAR imaging is confronted with complicated synchronization and motion compensation [4]–[6]. The array antenna can also be used to obtain high-resolution images of the forward-looking terrain [7], [8], but the azimuth resolution is greatly limited due to the size of the antenna.

Real beam scanning radar is another choice to obtain images of targets in forward-looking region. However, its azimuth resolution is determined by the slant range and the beamwidth of radar antenna. Even though the azimuth resolution can be enhanced by reducing the beamwidth, it is impossible to increase the antenna size infinitely to match the range resolution. Therefore, improving the azimuth resolution of scanning radar through signal processing is of great significance [9].

The monopulse method can be used to obtain the scattering information of the targets by monopulse angle measurement technology in azimuth. However, the method is not suitable for improving the azimuth resolution of extended targets [10].

Direction-of-arrival (DOA) estimation of signal source [11]–[13] can realize the azimuth superresolution using the spectrum estimation method. Nevertheless, this kind of methods needs a large number of snapshots [14]. Besides, these algorithms usually need to know the *a priori* information of the number of targets.

Moreover, since the radar echo in azimuth can be considered as convolution of scattering coefficients of targets and antenna pattern, deconvolution methods can be used to improve the azimuth resolution of scanning radar in theory [15]. However, deconvolution is an ill-conditioned problem due to the existence of noise.

Researchers have proposed many superresolution algorithms to solve the ill-posed problem, such as deconvolution method for multiantenna system [16], maximum entropy method [17], maximum *a posteriori* method [18], maximum likelihood method [19], iterative adaptive approach [20], [21], augmented Lagrangian method [22], deconvolution methods based on regularization [23], [24], etc. Among these algorithms, the method under the framework of regularized least squares which is called iterative shrinkage-thresholding algorithm (ISTA) is usually adopted due to its simplicity, robustness, and low sensitivity to noise. However, ISTA is recognized to have a slow convergence speed, which makes it difficult to meet the real-time imaging requirements of scanning radar.

In order to speed up its convergence, a two-step ISTA (TwISTA) [25] is proposed. It combines the advantages of ISTA and the iterative reweighted shrinkage algorithm (IRSA)

Manuscript received May 20, 2019; revised November 20, 2019; accepted December 24, 2019. Date of publication January 23, 2020; date of current version February 14, 2020. This work was supported by the National Natural Science Foundation of China under Grants 61671117 and 61901092. *(Corresponding author: Wenchao Li.)*

The authors are with the Department of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: liwenchao@uestc.edu.cn; sdniumeihua@163.com; zhang_yongchao1@163.com; yulinhuang@uestc.edu.cn; jyyang@uestc.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2020.2964589

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

together. Through constructing the next iteration vector by linear combination of two previous iteration vectors, the number of iterations can be reduced, and the convergence rate of the algorithm can be improved to a certain extent.

A fast ISTA (FISTA) which has a higher convergence speed is proposed in [26]. This algorithm adopts Nesterov accelerated gradient method; before each iteration, it utilizes historical iteration information to construct iteration vectors, thus reducing the number of iterations and achieving acceleration.

By extending FISTA's acceleration idea to the vector extrapolation technique according to the Taylor series expansion formula, an accelerated iteration threshold shrinkage algorithm was proposed in [27], whose idea is constructing a prediction vector from extrapolating two historical iteration vectors before each iteration, aiming to reduce the iteration number to realize acceleration. However, only the first-order difference information of the iteration sequence is used when constructing the prediction vector.

In this article, an accelerated ISTA (AISTA) based on the principle of second-order vector extrapolation and Taylor series expansion is proposed. The main idea of the accelerated algorithm is that it conducts the iterations on a prediction vector constructed by the first- and the second- order difference information of the iteration vector, to improve the convergence speed further. The main contribution is that an AISTA is developed, and the algorithm is compared with other similar algorithms and applied to forward-looking scanning radar superresolution imaging successfully.

The rest of this article is organized as follows. In Section II, the signal model of scanning radar in azimuth is illustrated. In Section III, the principle of the proposed algorithm is introduced. In Section IV, simulation and experimental results are given to verify the effectiveness of the algorithm. Finally, Section V concludes this article.

II. Signal Model

In the real beam scanning radar system, the received echo can be considered as the convolution of the effective scattering coefficients and the convolution kernel consisting of the antenna pattern in azimuth and the pulse modulation function in range direction [18]. Mathematically, it can be expressed as follows:

$$s(\theta, R) = A_r x(\theta, R) \otimes \left[h(\theta) f_{pm}\left(\frac{2R}{c}\right)\right] \tag{1}$$

where θ means angle in azimuth, R denotes the range, $s(\theta, R)$ represents the output voltage of receiver, A_r is the amplitude constant related to the system, $x(\theta, \phi, R)$ denotes the scattering coefficients, $\otimes$ represents the convolution operation, $h(\theta)$ is the antenna pattern, $f_{pm}(\frac{2R}{c})$ denotes the pulse modulation function, and c is the light speed.

Considering the echo in one fixed range bin R_0, we have

$$s(\theta, R_0) = A_r x(\theta, R_0) \otimes \left[h(\theta) f_{pm}\left(\frac{2R_0}{c}\right)\right]. \tag{2}$$

For simplicity, assuming A_r equals one and ignoring the constant items, (2) can be rewritten as follows [28]:

$$s(\theta) = h(\theta) \otimes x(\theta). \tag{3}$$

Taking noise into account, the signal model is shown in Fig. 1. Mathematically, (3) can be expressed as follows:

$$s(\theta) = h(\theta) \otimes x(\theta) + n(\theta) \tag{4}$$

where $n(\theta)$ represents noise.

Suppose there are three point targets in a range bin as shown in Fig. 1, the targets in the same beam cannot be separated when they are densely distributed, as the echo energy of the point targets in the azimuth is broadened due to the weighting effect of antenna pattern.

Mathematically, (4) can be abbreviated in matrix-vector form to

$$\boldsymbol{s} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{5}$$

where $\boldsymbol{s}$ denotes echo vector, $\boldsymbol{x}$ denotes vector of scattering coefficient, $\boldsymbol{n}$ represents noise vector, $\boldsymbol{H}$ represents convolution matrix transformed from antenna pattern $\boldsymbol{h}$, $\boldsymbol{h} = [\boldsymbol{h}_0\ \boldsymbol{h}_1 \ldots \boldsymbol{h}_{l-1},]$, as shown in the following, where l is the length of h:

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{h}_0 & & & & \boldsymbol{h}_{l-1} & \cdots & \boldsymbol{h}_1 \\ \boldsymbol{h}_1 & \boldsymbol{h}_0 & & & & \ddots & \boldsymbol{h}_2 \\ \vdots & \boldsymbol{h}_1 & \ddots & & & & \vdots \\ \boldsymbol{h}_{l-1} & \vdots & & & & & \\ & \boldsymbol{h}_{l-1} & & & & & \\ & & \ddots & & & & \\ & & & \ddots & & & \\ & & & & \boldsymbol{h}_{l-1} & \cdots & \boldsymbol{h}_1\ \boldsymbol{h}_0 \end{bmatrix}_{N\times N} \tag{6}$$

where N is the number of sampling points in azimuth.

Given $\boldsymbol{H}$, the aim of scanning radar superresolution is to recover $\boldsymbol{x}$ from $\boldsymbol{s}$, which is known as solving inverse problem. It is worth noting that even if the noise could be ignored and the matrix $\boldsymbol{H}$ is invertible, the error of the calculation result $\hat{\boldsymbol{x}} = \boldsymbol{H}^{-1}\boldsymbol{s}$ is always great, as the solution is sensitive to small changes of the data due to the ill-condition of matrix $\boldsymbol{H}$.

The existence of observation noise will inevitably lead to numerical instability and unacceptable results. In order to overcome the difficulty, regularization is introduced, and it is a kind of least squares method with a penalty term. As for the penalty term which describes the confidence level of prior information, since l_1 norm is usually adopted to induce sparsity in the optimal solution and the targets we are interested in are always sparse, the l_1 norm is used here [26], [29]. Then, the solution formula to the problem can be described as

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}}\{F(\boldsymbol{x}) \equiv \|\boldsymbol{s} - \boldsymbol{H}\boldsymbol{x}\|^2 + \lambda\|\boldsymbol{x}\|_1\} \tag{7}$$

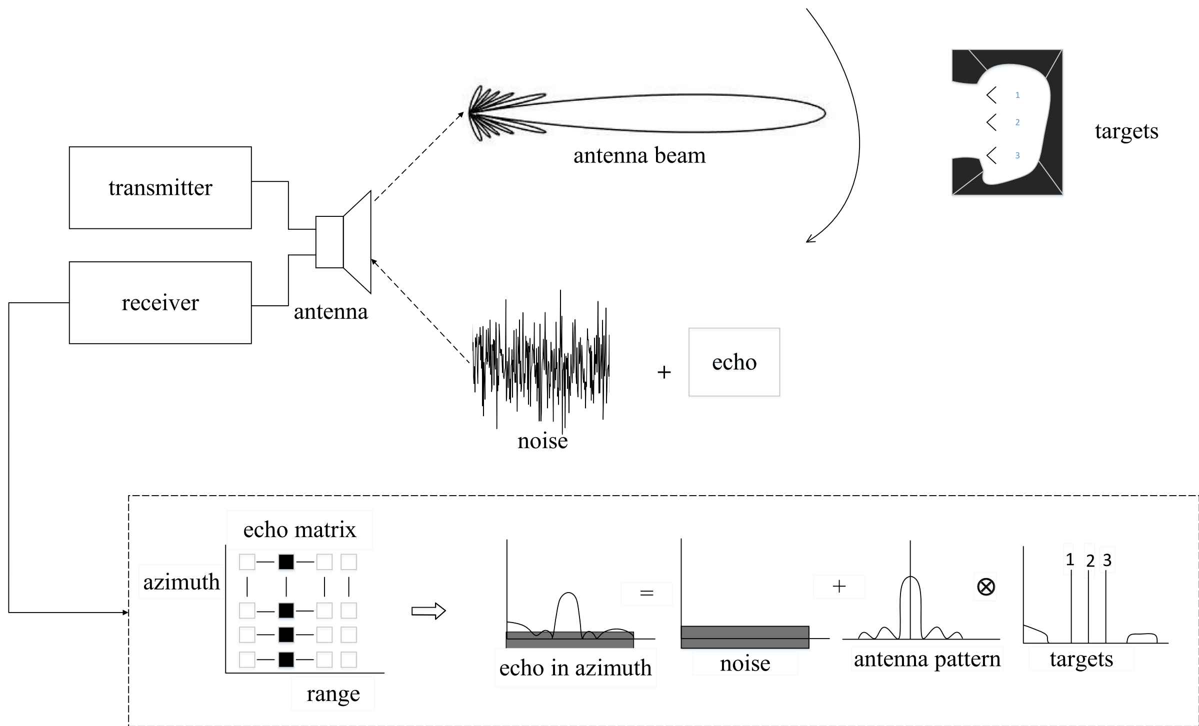


Fig. 1. Signal model.

where $F(\boldsymbol{x})$ is objective function, $\|\boldsymbol{x}\|_1$ denotes l_1 norm. λ is the regularization parameter used to make a balance between data fidelity items and constraint items.

III. Principle of Proposed Algorithm

In this section, the principles of ISTA and AISTA are illustrated, then an accelerated scheme of ISTA based on second-order vector extrapolation technique is proposed. In the end, the computation complexity of these algorithms are analyzed.

A. Iterative Shrinkage-Thresholding Algorithm

Equation (7) in Section II is the least square optimization problem. Let the least square term be $f(\boldsymbol{x})$ (smooth convex function)

$$f(\boldsymbol{x}) = \|\boldsymbol{s} - \boldsymbol{H}\boldsymbol{x}\|^2. \tag{8}$$

Then, (7) can be written as

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}}\{F(\boldsymbol{x}) \equiv f(\boldsymbol{x}) + \lambda\|\boldsymbol{x}\|_1\}. \tag{9}$$

One of the simplest ways to obtain $\hat{\boldsymbol{x}}$ is using gradient descent algorithm, and the iteration sequence $\boldsymbol{x}_k$ is generated via

$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} - t\nabla F(\boldsymbol{x}_{k-1}) \tag{10}$$

where $\boldsymbol{x}_k$ is the kth iteration vector, and t is step size which is restricted to $(0, \frac{1}{\|\boldsymbol{H}^T\boldsymbol{H}\|})$ to guarantee the convergence of the algorithm.

However, $\lambda\|\boldsymbol{x}\|_1$ is nondifferential, and $\nabla F(\boldsymbol{x}_{k-1})$ does not exist. Therefore, the gradient descent algorithm cannot be used. As the gradient descent algorithm can be regarded as a special form of proximal regularization method, the solution formula (10) is equivalent to

$$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}}\Big\{f(\boldsymbol{x}_{k-1}) + \langle\boldsymbol{x} - \boldsymbol{x}_{k-1}, \nabla f(\boldsymbol{x}_{k-1})\rangle + \frac{1}{2t}\left\|\boldsymbol{x} - \boldsymbol{x}_{k-1}\right\|^2 + \lambda\left\|\boldsymbol{x}\right\|_1\Big\} \tag{11}$$

where $\langle\cdot\rangle$ denotes inner product, and $\nabla f(\boldsymbol{x}_k)$ indicates the gradient of the least square term $f(\boldsymbol{x})$ at $\boldsymbol{x}_k$

$$\nabla f(\boldsymbol{x}_k) = 2\boldsymbol{H}^T\left(\boldsymbol{H}\boldsymbol{x}_k - \boldsymbol{s}\right). \tag{12}$$

Ignoring the constant terms, we can reexpress (11) as

$$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}}\Big\{\frac{1}{2t}\left\|\boldsymbol{x} - (\boldsymbol{x}_{k-1} - t\nabla f(\boldsymbol{x}_{k-1}))\right\|^2 + \lambda\left\|\boldsymbol{x}\right\|_1\Big\}. \tag{13}$$

Due to the separable property of l_1 norm, (13) can be reduced to a simple computation procedure [26]

$$\boldsymbol{x}_k = T_{\lambda t}\left(\boldsymbol{x}_{k-1} - t\nabla f(\boldsymbol{x}_{k-1})\right) \tag{14}$$

where λt is the product of t and λ, and it denotes the threshold of the thresholding shrinkage operator $T_{\lambda t}(\cdot)$ and

$$T_{\lambda t}(\boldsymbol{x})_i = \left(|(\boldsymbol{x})_i| - \lambda t\right)_+ \operatorname{sgn}((\boldsymbol{x})_i) \tag{15}$$

where $(\cdot)_+$ denotes positive operator and $\operatorname{sgn}(\cdot)$ is symbolic function.

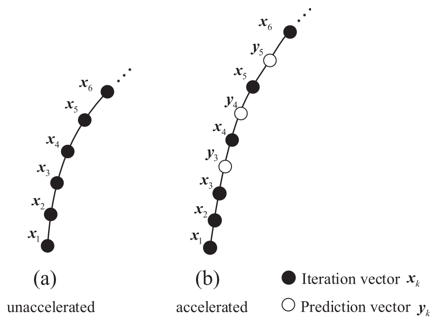


Fig. 2. Schematic diagram of the acceleration process.

Equation (14) is the basic iteration of ISTA through which $\hat{\boldsymbol{x}}$ can be obtained. Obviously, ISTA is derived as an extension of gradient algorithms, the slow convergence speed $O(1/k)$ of gradient algorithms is also maintained [30].

B. Vector Extrapolation Acceleration Technique

Formally, the vector extrapolation method is similar to the line search method [31], but unlike the traditional line search acceleration technique, its search direction vector is calculated based on the difference vector between the current iteration and the historical iteration.

The vector extrapolation schematic diagram is shown in Fig. 2, where Fig. 2(a) is the iteration path of unaccelerated algorithm and Fig. 2(b) denotes the iterative path of the vector extrapolation acceleration algorithm. Before each iteration, a predicted vector is constructed by extrapolating the previous iterative vector along the direction of the difference vector. Through vector extrapolation, the iterative path can be extended without iterative operation; then, the acceleration could be realized.

C. Accelerated Iterative Shrinkage-Thresholding Algorithm

Aiming at the inherent problem of ISTA, AISTA was proposed based on the vector extrapolation technique [27]. Mathematically, the vector extrapolation process can be expressed as follows:

$$\boldsymbol{y}_k = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k \tag{16}$$

where

$$\boldsymbol{d}_k = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}. \tag{17}$$

Then, the next iterative vector can be obtained by $\boldsymbol{x}_{k+1} = \psi(\boldsymbol{y}_k)$. $\psi(\cdot)$ denotes the ISTA operation.

The step size α_k is the key factor that affects the convergence speed and robustness of the algorithm. In order to ensure the convergence speed and stability of the algorithm, α_k is constructed according to the geometric convergence of iterative sequences and the similarity of the adjacent gradient vectors as follows:

$$\alpha_k = \frac{\sum \boldsymbol{g}_k \cdot \boldsymbol{g}_{k-1}}{\sum \boldsymbol{g}_{k-1} \cdot \boldsymbol{g}_{k-1}}, 0 < \alpha_k < 1 \tag{18}$$

where

$$\boldsymbol{g}_k = \boldsymbol{x}_k - \boldsymbol{y}_{k-1}. \tag{19}$$

Algorithm 1: AISTA.

Input: parameters needed for ISTA iteration λ, t
Intial step: set $\boldsymbol{x}_0$ to **0**,
calculate $\boldsymbol{x}_1$ according to (14)
calculate $\boldsymbol{x}_2$ according to (14)

FOR: $k = 2$ to n repeat
Compute the prediction step α_k
$d_k = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$
$g_k = \boldsymbol{x}_k - \boldsymbol{y}_{k-1}$
$\alpha_k = \frac{\sum_{gk \cdot gk-1}}{\sum_{gk-1 \cdot gk-2}}$
If $\alpha_k < 0$, set the value of α_k to 0
If $\alpha_k > 1$, set the value of α_k to 1
Construct the prediction vector y_k
$\boldsymbol{y}_k = \boldsymbol{x}_k + \alpha_k(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$
Compute the next iteration vector $\boldsymbol{x}_k + 1$ through (14) from $\boldsymbol{y}_k$

The procedure of AISTA is shown in Algorithm 1.

D. Second-Order AISTA (SO-AISTA)

The Taylor series expansion formula of iteration vector $\boldsymbol{y}_k$ at $\boldsymbol{x}_k$ is

$$\boldsymbol{y}_k = \boldsymbol{x}_k + \alpha\Delta\boldsymbol{x}_k + \frac{1}{2!}\alpha^2\Delta^2\boldsymbol{x}_k + \frac{1}{3!}\alpha^3\Delta^3\boldsymbol{x}_k + \cdots + \frac{1}{n!}\alpha^n\Delta^n\boldsymbol{x}_k + \cdots \tag{20}$$

where $\Delta^n\boldsymbol{x}_k$ is nth difference at $\boldsymbol{x}_k$.

According to the description in Section III-C, the acceleration technique of AISTA uses only the first two terms in the Taylor series expansion formula, which is called first-order vector extrapolation here. Although the convergence speed can be accelerated, it may not be enough to obtain satisfying acceleration as there is still some distance between the predicted iteration vector and the true iteration vector.

Moreover, we can find that the more previous information is used, the more accurate the approximation will be. Hence, we can use high-order terms to make the difference between the prediction vector and the true iteration vector as small as possible.

In the meantime, it is worthy to mention that the information decreases greatly as the order increases. In the case where each prediction is pushed forward one step, retaining higher order information may lead to an insignificant acceleration at the cost of storing resources.

Take the simulations in Section IV as an example, the second-order difference information $\boldsymbol{E}_k$ can be expressed as follows:

$$\boldsymbol{E}_k = \frac{1}{2}\alpha^2\Delta^2\boldsymbol{x}_k \tag{21}$$

where

$$\Delta^2\boldsymbol{x}_k = (\boldsymbol{d}_k - \boldsymbol{d}_{k-1}) \tag{22}$$

TABLE I
MODULUS OF DIFFERENCE INFORMATION OF EACH ORDER

Item	Iteration number	Modulus
$\boldsymbol{d}_k$	10	5.8853e-02
$\boldsymbol{e}_k$	10	3.8637e-03
$\boldsymbol{t}_k$	10	3.7233e-04
$\boldsymbol{d}_k$	50	8.7142e-03
$\boldsymbol{e}_k$	50	1.0338e-04
$\boldsymbol{t}_k$	50	4.8780e-06

while the third-order difference information $\boldsymbol{T}_k$ can be expressed as follows:

$$\boldsymbol{T}_k = \frac{1}{6}\alpha^3\Delta^3\boldsymbol{x}_k \tag{23}$$

where

$$\Delta^3\boldsymbol{x}_k = (\boldsymbol{d}_k - \boldsymbol{d}_{k-1}) - (\boldsymbol{d}_{k-1} - \boldsymbol{d}_{k-2}). \tag{24}$$

Let

$$\boldsymbol{e}_k = \frac{1}{2}\Delta^2\boldsymbol{x}_k \tag{25}$$

$$\boldsymbol{t}_k = \frac{1}{6}\Delta^3\boldsymbol{x}_k. \tag{26}$$

In simulations, we observed the modulus of $\boldsymbol{d}_k$, $\boldsymbol{e}_k$, and $\boldsymbol{t}_k$ in the iteration process which are listed in Table I for comparison.

It can be seen that the difference between the three items is very large. After ten iterations, the modulus of the second-order vector is ten times as big as that of the third-order vector, and the modulus of the second-order vector is smaller than that of the first-order by one power of ten. After 50 iterations, the modulus of the third-order vector is smaller than that of the second-order vector by 100 times. And the third-order vector item $\boldsymbol{T}_k$ has one more α_k than the second-order vector $\boldsymbol{E}_k$, making it smaller since α_k is smaller than one. In the meantime, as $\boldsymbol{T}_k$ is usually a large vector, its modulus is so small that it can be ignored, not to mention higher order items.

Considering the balance between the cost of system resources for storing and the growth of speed, second-order difference information may be effective to increase the accuracy of prediction and enhance the previous acceleration scheme, and higher order items could be abandoned. Therefore, the second-order vector extrapolation scheme is used in this article. As for the importance of the second order, it will be discussed in detail in the simulation results.

Using the idea that the next iteration vector can be predicted by the current iteration vector and the previous iteration vector, a predicted vector should be calculated before each operation of ISTA, then the iteration path will be extended, and the algorithm can be accelerated. Mathematically, ISTA can be accelerated as follows:

$$\begin{aligned}\boldsymbol{y}_k &= \boldsymbol{x}_k + \alpha_k\left(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\right)\\ &\quad + \frac{\alpha_k^2}{2}\left[(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) - (\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-2})\right]\end{aligned} \tag{27}$$

Algorithm 2: SO-AISTA.

Input: parameters needed for ISTA iteration λ, t
Intial step: set $\boldsymbol{x}_0$ to **0**,
calculate $\boldsymbol{x}_1$ according to (14)
calculate $\boldsymbol{x}_2$ according to (14)

FOR: $k = 2$ to n repeat
Compute the prediction step α_k
$\boldsymbol{d}_k = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$
$\alpha_k\left(\frac{\sum \boldsymbol{d}_k\cdot\boldsymbol{d}_k}{\sum \boldsymbol{d}_{k-1}\cdot\boldsymbol{d}_{k-1}}\right)^{\frac{1}{2}}$
If $\alpha_k > 1$, set the value of α_k to 1
Construct the prediction vector $\boldsymbol{y}_k$
$\boldsymbol{y}_k = \boldsymbol{x}_k + \alpha_k(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) +$
$\frac{\alpha_k^2}{2}[(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) - (\boldsymbol{x}_k - 1 - \boldsymbol{x}_{k-2})]$
Compute the next iteration vector $\boldsymbol{x}_k + 1$ through (14) from $\boldsymbol{y}_k$

$$\boldsymbol{x}_{k+1} = \psi(\boldsymbol{y}_k) \tag{28}$$

$$\boldsymbol{d}_k = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}. \tag{29}$$

Since α_k is a key parameter that determines the length of the prediction step, which can affect the acceleration directly, its selection is important to ensure the efficiency of the algorithm. Here, if we keep using the α_k constructed in AISTA, the acceleration may not be improved further as the α_k in second-order extrapolation does not change much compared with that in the first-order extrapolation. Thus, we adopt a new step construction way where only the geometric convergence of the iterative sequence is taken into account, and α_k is constructed as follows:

$$\alpha_k = \left(\frac{\sum \boldsymbol{d}_k\cdot\boldsymbol{d}_k}{\sum \boldsymbol{d}_{k-1}\cdot\boldsymbol{d}_{k-1}}\right)^{\frac{1}{2}},\ 0 < \alpha_k < 1. \tag{30}$$

A detailed derivation of α_k will be given in Appendix. According to (30), α_k is constructed according to the proportion of vector's inner product, and it is always positive. Besides, since the vector series $\boldsymbol{d}_k$ decrease gradually when approaching the optimum solutions, α_k would not be larger than one.

The procedure of the proposed algorithm is shown in Algorithm 2. We can find that no more additional computation is needed in the algorithm compared with ISTA, only the storage for an extra iteration vector.

E. Time Complexity Analysis

The basic iteration step of ISTA is shown in (14), and we can find that each iteration includes two matrix-vector multiplications. Matrix-vector multiplication is dominant in an iteration whose time complexity is $O(N^2)$. Besides, the other calculations in each iteration are vector subtractions whose time complexity is so small that it can be ignored. Therefore, the time complexity of each ISTA iteration is $O(N^2)$, and N is the length of the iteration vector.

Compared with ISTA, the extra computation of SO-AISTA is the calculation of the predicated vector through vector dot

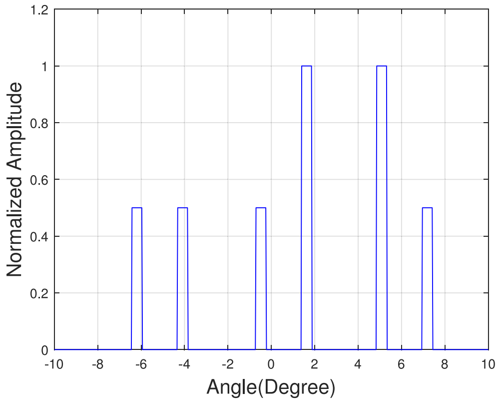


Fig. 3. Targets.

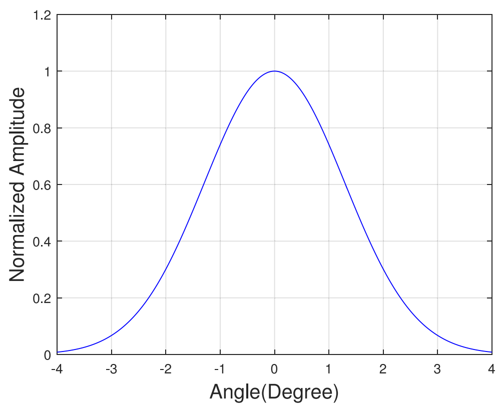


Fig. 4. Antenna pattern.

products and simple addition whose computational complexity is $O(N)$. Therefore, the time complexity of SO-AISTA is $O(N^2) + O(N)$. The extra computation burden $O(N)$ is relatively lower compared with that of ISTA when N is large. Moreover, compared with AISTA, SO-AISTA does not add additional computational complexity, only the storage of few vectors.

IV. Simulation and Experimental Results

In this section, simulations and real data processing are conducted to verify the effectiveness of the SO-AISTA. And the performance of SO-AISTA is compared with TwISTA and FISTA.

A. Simulations of Extended Targets

As shown in Fig. 3, the targets are distributed from -10° to $+10^\circ$. The antenna pattern is shown in Fig. 4. Besides, the pulse repetition frequency is 2000 Hz, and the antenna scanning velocity is 60°/s.

1) Simulations in High Signal-to-Noise Ratio (SNR): In order to observe the performance of the proposed algorithm under high SNR, the SNR is set to 30 dB here.

The real beam data in azimuth with Gaussian white noise is shown in Fig. 5(a). As we can see, the resolution in azimuth is very poor and the adjacent targets cannot be distinguished due to the poor azimuth resolution.

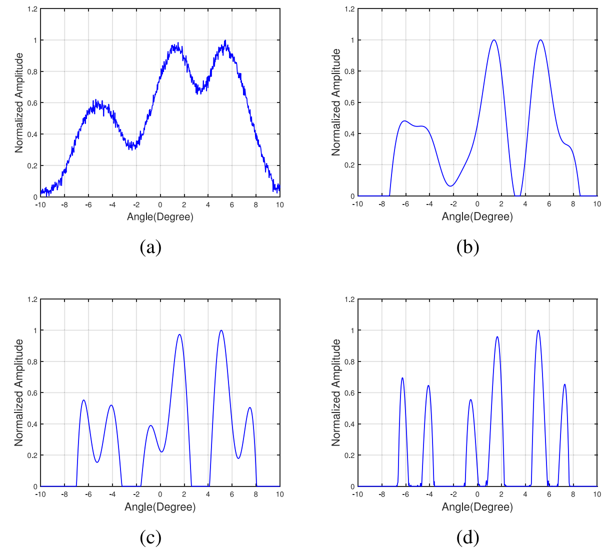


Fig. 5. Simulation results of extended targets. (a) Real beam data. (b) ISTA. (c) AISTA. (d) SO-AISTA.

TABLE II
Simulation Results

Algorithm	RMSE	Execution time
ISTA	0.3891	0.1505 s
AISTA	0.2991	0.1527 s
SO-AISTA	0.1844	0.1529 s

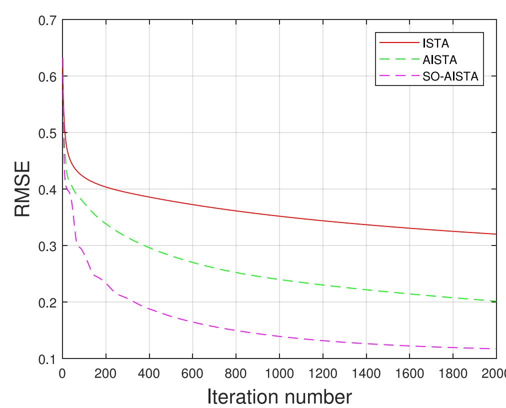


Fig. 6. RMSE curves.

Three different algorithms mentioned in Section III are adopted to process the real beam data, respectively, and the results with the same iterations are shown in Fig. 5, where Fig. 5(b) denotes the result of ISTA, Fig. 5(c) represents the result of AISTA, and Fig. 5(d) is the result of SO-AISTA.

From Fig. 5(b), we can see that the azimuth resolution is not significantly improved due to the slow convergence of ISTA. However, the other two algorithms can improve the azimuth resolution greatly and SO-AISTA has the best result.

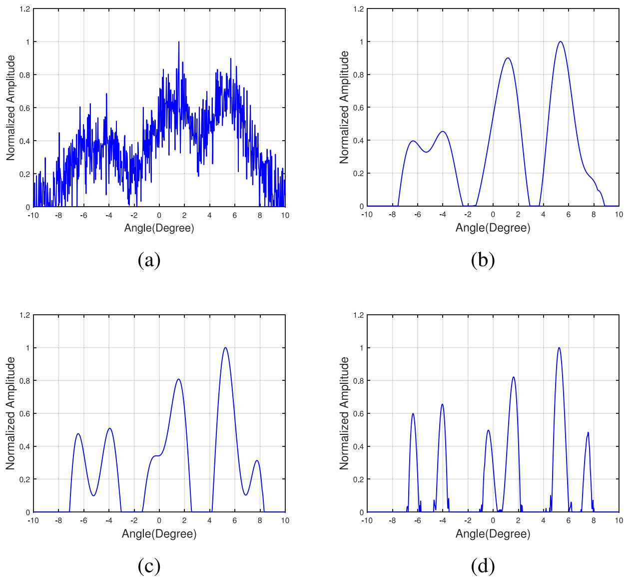


Fig. 7. Simulation results of extended targets. (a) Real beam data. (b) ISTA. (c) AISTA. (d) SO-AISTA.

TABLE III
SIMULATION RESULTS

Algorithm	RMSE	Execution time
ISTA	0.3741	0.0889 s
AISTA	0.2954	0.0924 s
SO-AISTA	0.1968	0.0927 s

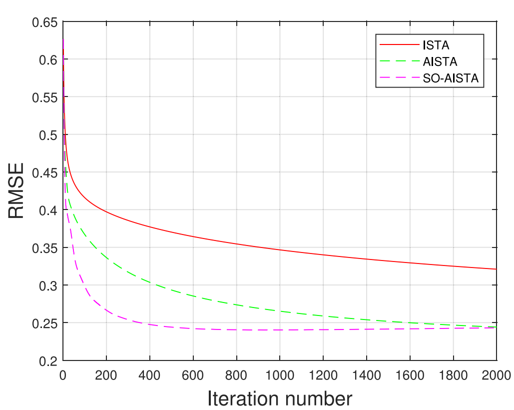


Fig. 8. RMSE curves.

Besides, to compare the performance of these three algorithms further, the RMSE [root-mean-square error, defined in (31)] of each algorithm is also calculated, and it is listed in Table II together with the execution times of each algorithm. With the same iteration number, the three algorithms have similar execution times. It means that the accelerated algorithms of ISTA bring little additional computational burden, which is consistent with the theoretical analysis. Moreover, the RMSE of the two accelerated algorithms are lower than that of ISTA at the same

TABLE IV
SIMULATION PARAMETERS

Parameter	Value
Carrier frequency	10 GHz
Band width	50 MHz
Antenna scanning velocity	100 $^\circ/s$
Pulse repetition frequency	2000 Hz

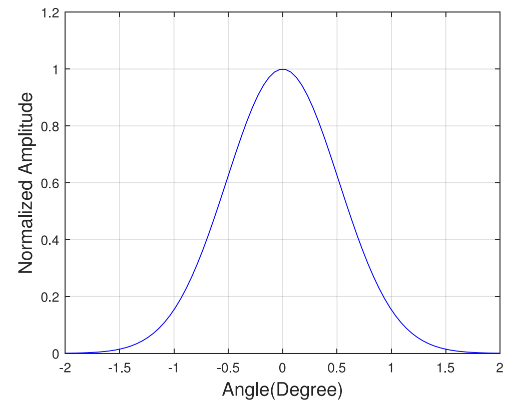


Fig. 9. Antenna pattern.

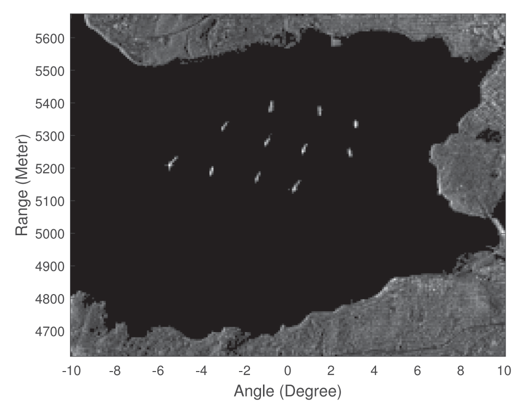


Fig. 10. Original scene.

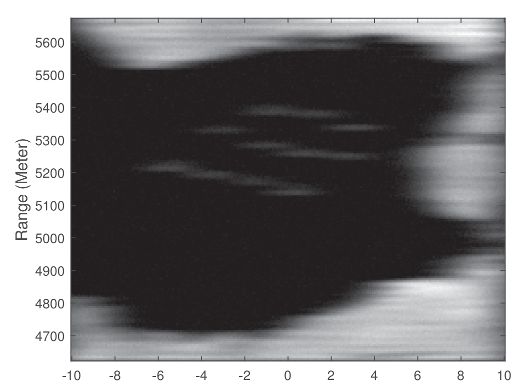


Fig. 11. Real beam image.

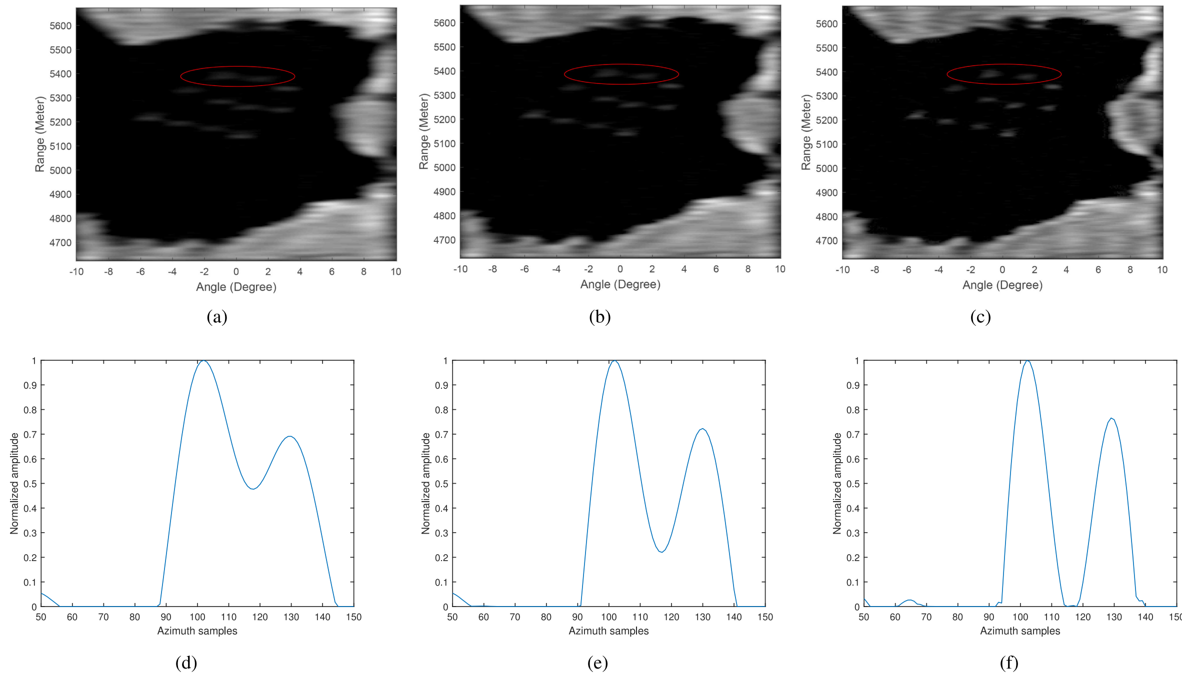


Fig. 12. Simulation results of area targets. (a) ISTA. (b) AISTA. (c) SO-AISTA. (d) Profile of ISTA. (e) Profile of AISTA. (f) Profile of SO-AISTA.

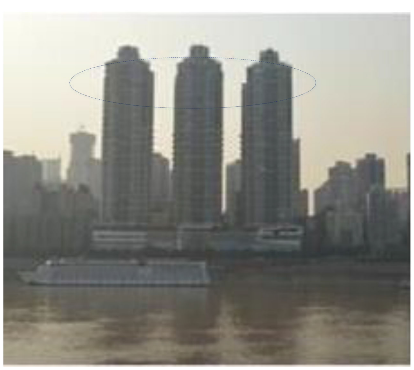


Fig. 13. Original scene.

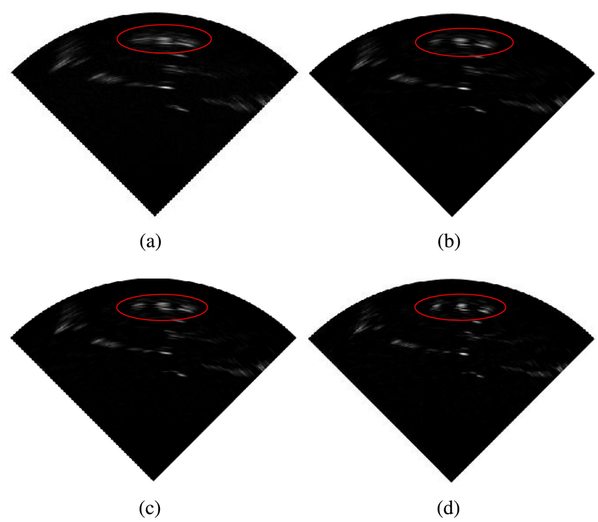


Fig. 14. Experimental results. (a) Real beam image. (b) ISTA. (c) AISTA. (d) SO-AISTA.

iteration number, and SO-AISTA has much lower RMSE, which indicates that both the accelerated versions of ISTA are effective, and SO-AISTA is better

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}((\boldsymbol{x}_k)_i - (\boldsymbol{x}_*)_i)^2}{N}} \tag{31}$$

where $\boldsymbol{x}_*$ denotes the true solution.

Furthermore, the curves of RMSE of the three algorithms are shown in Fig. 6. It can be seen that the RMSE of the two accelerated algorithms decrease faster than that of ISTA, and the RMSE of SO-AISTA decreases much faster, which proves that both AISTA and SO-AISTA can accelerate ISTA, and SO-AISTA is more efficient.

2) Simulations in Low SNR: For the purpose of testing the effectiveness of the proposed algorithm under low SNR, the SNR is set to 10 dB here, and Fig. 7(a) is the real beam data in azimuth with Gaussian white noise. It can be seen that the azimuth resolution is extremely poor, and the targets are difficult to be distinguished.

Aiming at improving the azimuth resolution, three algorithms (ISTA, AISTA, SO-AISTA) are used to process the real beam data with the same iterations, respectively, and the results are shown, respectively, in Fig. 7, where Fig. 7(b) denotes the result of ISTA, Fig. 7(c) is the result of AISTA, and Fig. 7(d) denotes the result of SO-AISTA.

It can be seen that all of the three algorithms can improve the azimuth resolution to a certain extent, and it is worth noting that

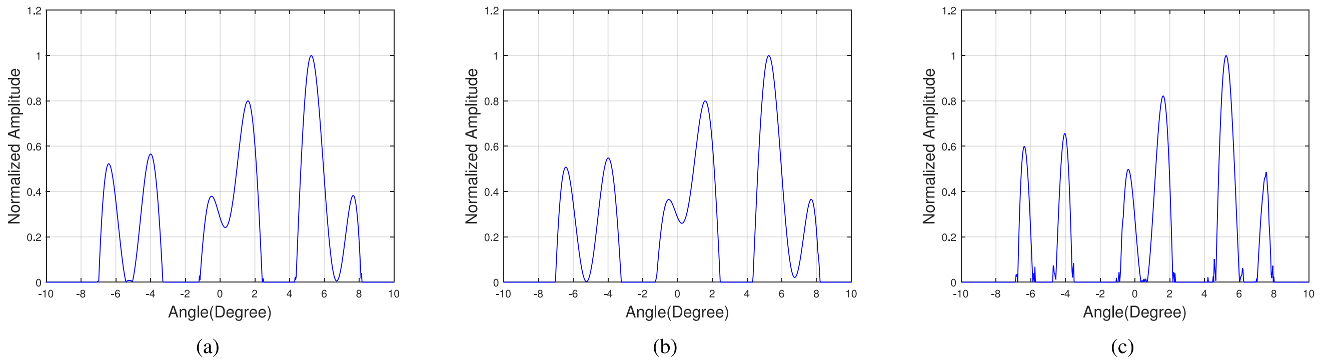


Fig. 15. Results of different algorithms. (a) TwISTA. (b) FISTA. (c) SO-AISTA.

SO-AISTA can separate the six targets completely, but ISTA and AISTA still need a lot of iterations to achieve this.

Besides, Table III gives the RMSE and execution time of each algorithm. Under the same number of iterations, the execution time of SO-AISTA is similar to that of AISTA. Moreover, the two accelerated algorithms do not add too much additional computational burden.

Fig. 8 shows the RMSE curves of the three algorithms. We can see that SO-AISTA has the fastest convergence speed.

B. Simulations of Area Targets

In this section, the simulations of area targets are conducted to validate the effectiveness of the proposed algorithm further. The main parameters used in the simulations are shown in Table IV.

Fig. 9 shows the antenna pattern. The original scene is shown as Fig. 10 and Fig. 11 denotes the real beam echo in azimuth with the SNR of 30 dB. We can see that the targets in the scene are aliased due to the poor azimuth resolution.

Then, different algorithms are used to process the echo with the same iterations, and the results obtained by the three algorithms are shown in Fig. 12, where Fig. 12(a) is the result of ISTA, Fig. 12(b) denotes the result of AISTA, and Fig. 12(c) shows the result of SO-AISTA. In order to illustrate the convergence speed of different algorithms clearly, an enlarged view of the details in the circle of Fig. 12 is given in the profile, just as shown in Fig. 12(d)–(f), respectively. Besides, the execution time of each algorithm is 8.6587, 8.7203, and 8.7204 s, respectively. It can be seen that with the similar execution time, all the three methods can improve the azimuth resolution, and SO-AISTA does the best.

C. Experimental Results

In this section, real data processing is conducted to verify the performance of the algorithm. The original scene is shown in Fig. 13. As we can see, there are three buildings in the imaging area.

The real beam image of scanning radar is shown in Fig. 14(a). Regrettably, it is difficult to distinguish the three buildings due to the poor azimuth resolution.

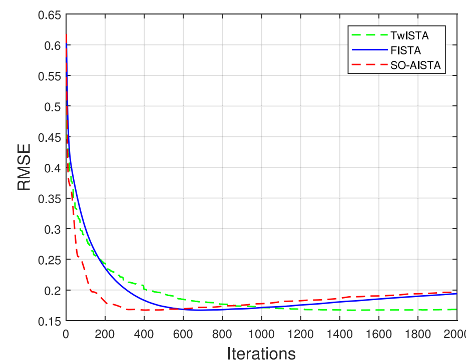


Fig. 16. RMSE curves.

In order to improve the azimuth resolution, the three algorithms (ISTA, AISTA, and SO-AISTA) are used to process the real beam data.

The results of the three algorithms with ten iterations are given in Fig. 14, where Fig. 14(b)–(d) represents the results obtained by ISTA, AISTA, and SO-AISTA, respectively, and the execution time of each algorithm is 9.8043, 9.8982, and 9.9004 s, respectively. From the results, we can see that ISTA, AISTA, and SO-AISTA all can distinguish the three buildings. SO-AISTA is the best in sharpening the real beam image and making the three targets completely distinguishable, which indicates that SO-AISTA has the fastest convergence speed.

D. Comparisons With Other Algorithms

In this section, comparisons are made among the AISTAs through simulations of extended targets.

Using the real beam data shown in Fig. 7(a), the processed results of different algorithms with the same iterations are given in Fig. 15, where Fig. 15(a) denotes the result of TwISTA, Fig. 15(b) is the result of FISTA, and Fig. 15(c) denotes the result of SO-AISTA.

From Fig. 15, we can find that with the same iterations, the performance of TwISTA and FISTA is similar, while SO-AISTA gives the best results as it distinguishes the targets completely.

Furthermore, the RMSE curves of the three methods are illustrated in Fig. 16. We can find that the RMSE of SO-AISTA decreases fastest.

V. Conclusion

Based on vector extrapolation and Taylor series expansion, a prediction iteration vector is constructed using the first- and the second-order difference information of the iterative vectors, and SO-AISTA is proposed in this article. The results of simulations and real data processing show that SO-AISTA can accelerate ISTA and perform even better than the other three accelerated methods AISTA, FISTA, and TwISTA without extra computational burden.

Appendix
Step Length α_k for Prediction Operation

In this section, we will further explain why the prediction step size does not adopt the step construction method of the first-order vector extrapolation.

$\boldsymbol{x}_k$ is known as an iterative vector, and

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{d}_{k+1}. \tag{A.1}$$

Besides, $\boldsymbol{x}^*$ is known as the true solution, and the objective function of the least squares solution can be rewritten as

$$F(\boldsymbol{x}) = -(\boldsymbol{x}-\boldsymbol{x}^*)^T \boldsymbol{H}(\boldsymbol{x}-\boldsymbol{x}^*). \tag{A.2}$$

It is not difficult to find that the gradient of the objective function is

$$\nabla F(\boldsymbol{x}) = -2\boldsymbol{H}(\boldsymbol{x}-\boldsymbol{x}^*). \tag{A.3}$$

According to the gradient descent method

$$\begin{aligned}\boldsymbol{d}_{k+1} &= t\nabla F(\boldsymbol{x}_k)\\ &= -2t\boldsymbol{H}(\boldsymbol{x}_k-\boldsymbol{x}^*).\end{aligned} \tag{A.4}$$

Assuming that $\boldsymbol{H}$ is a symmetric positive definite matrix, it can be eigenvalue-decomposed as follows:

$$\boldsymbol{H} = \boldsymbol{S}\Lambda_{\boldsymbol{H}}\boldsymbol{S}^{-1} \tag{A.5}$$

where $\Lambda_{\boldsymbol{H}}$ is the eigenvalue matrix of $\boldsymbol{H}$, and $\boldsymbol{S}$ is the reversible eigenvector matrix.

Let

$$\boldsymbol{B} = \boldsymbol{I} - 2t\boldsymbol{H} \tag{A.6}$$

and

$$t < \frac{1}{2\max(\Lambda_{\boldsymbol{H}})} \tag{A.7}$$

then

$$\Lambda_{\boldsymbol{B}} = \boldsymbol{I} - 2t\Lambda_{\boldsymbol{H}} \tag{A.8}$$

and $\boldsymbol{x}_k$ can be represented as follows:

$$\boldsymbol{x}_k = \boldsymbol{S}\Lambda_{\boldsymbol{B}}^k\boldsymbol{S}^{-1}(\boldsymbol{x}_0-\boldsymbol{x}^*)+\boldsymbol{x}^*. \tag{A.9}$$

Thus

$$\boldsymbol{d}_k = \boldsymbol{S}\Lambda_{\boldsymbol{B}}^k\left(\boldsymbol{I}-\Lambda_{\boldsymbol{B}}^{-1}\right)\boldsymbol{S}^{-1}(\boldsymbol{x}_0-\boldsymbol{x}^*) \tag{A.10}$$

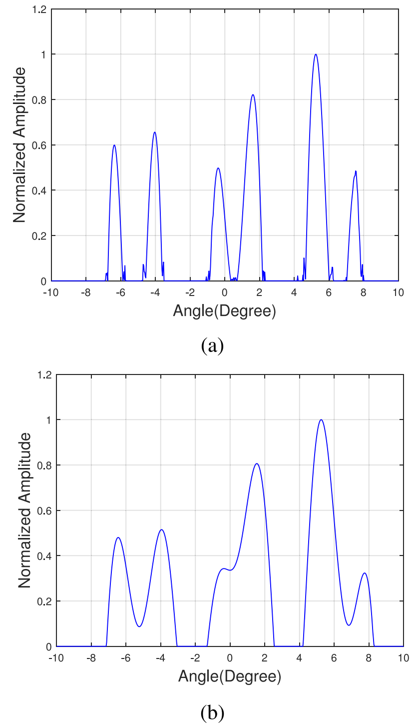


Fig. 17. Results of different step sizes. (a) $\alpha_k = (\frac{\sum \boldsymbol{d}_k\cdot\boldsymbol{d}_k}{\sum \boldsymbol{d}_{k-1}\cdot\boldsymbol{d}_{k-1}})^{\frac{1}{2}}$. (b) $\alpha_k = \frac{\sum \boldsymbol{g}_k\cdot\boldsymbol{g}_{k-1}}{\sum \boldsymbol{g}_{k-1}\cdot\boldsymbol{g}_{k-1}}$.

$$\boldsymbol{x}_{k-\delta} = \boldsymbol{S}\Lambda_{\boldsymbol{B}}^{k-\delta}\boldsymbol{S}^{-1}(\boldsymbol{x}_0-\boldsymbol{x}^*)+\boldsymbol{x}^* \tag{A.11}$$

$$\boldsymbol{d}_{k-\delta} = \boldsymbol{S}\Lambda_{\boldsymbol{B}}^{k-\delta}\left(\boldsymbol{I}-\Lambda_{\boldsymbol{B}}^{-1}\right)\boldsymbol{S}^{-1}(\boldsymbol{x}_0-\boldsymbol{x}^*). \tag{A.12}$$

The difference between the adjacent iteration vectors is

$$\boldsymbol{h}_{k,-\delta} = \boldsymbol{x}_k - \boldsymbol{x}_{k-\delta} = \boldsymbol{S}\Lambda_{\boldsymbol{B}}^{k}\left(\boldsymbol{I}-\Lambda_{\boldsymbol{B}}^{-\delta}\right)\boldsymbol{S}^{-1}(\boldsymbol{x}_0-\boldsymbol{x}^*). \tag{A.13}$$

Then, the vector obtained by iterating δ steps forward can be predicted as follows:

$$\boldsymbol{x}_{k+\delta} = \boldsymbol{S}\Lambda_{\boldsymbol{B}}^{k+\delta}\boldsymbol{S}^{-1}(\boldsymbol{x}_0-\boldsymbol{x}^*)+\boldsymbol{x}^* \tag{A.14}$$

and the difference at $\boldsymbol{x}_{k+\delta}$ is

$$\begin{aligned}\boldsymbol{h}_{k,\delta} &= \boldsymbol{x}_{k+1} - \boldsymbol{x}_k\\ &= \boldsymbol{S}\Lambda_{\boldsymbol{B}}^{k+\delta}\left(\boldsymbol{I}-\Lambda_{\boldsymbol{B}}^{-1}\right)\boldsymbol{S}^{-1}(\boldsymbol{x}_0-\boldsymbol{x}^*).\end{aligned} \tag{A.15}$$

Only the geometric convergence of the iteration sequence is considered, regardless of the similarity of the gradient direction. Here, we guarantee that the amplitude relationship satisfies this least squares projection condition, as described in the following:

$$\|\boldsymbol{h}_{k,\delta}\|_2 = \alpha_k\|\boldsymbol{h}_{k,-\delta}\|_2. \tag{A.16}$$

Then, we have

$$\alpha_k = \frac{\|\boldsymbol{h}_{k,\delta}\|_2}{\|\boldsymbol{h}_{k,-\delta}\|_2}. \tag{A.17}$$

Since

$$\boldsymbol{I} - \Lambda_{\boldsymbol{B}}^{-\delta} \approx \delta\left(\boldsymbol{I}-\Lambda_{\boldsymbol{B}}^{-1}\right) \tag{A.18}$$

$$\begin{aligned}\boldsymbol{h}_{k,\delta} &\approx \delta \boldsymbol{S} \Lambda_{\boldsymbol{B}}^{k} \left(\boldsymbol{I} - \Lambda_{\boldsymbol{B}}^{-1}\right) \boldsymbol{S}^{-1} \left(\boldsymbol{x}_0 - \boldsymbol{x}_s\right) \\ &= \delta \boldsymbol{d}_k \end{aligned} \tag{A.19}$$

and

$$\boldsymbol{h}_{k,-\delta} \approx \delta \boldsymbol{d}_{k-\delta}. \tag{A.20}$$

Therefore

$$\alpha_k = \frac{\|\boldsymbol{d}_k\|_2}{\|\boldsymbol{d}_{k-1}\|_2} = \left(\frac{{\boldsymbol{d}_k}^T \boldsymbol{d}_k}{{\boldsymbol{d}_{k-1}}^T \boldsymbol{d}_{k-1}}\right)^{\frac{1}{2}}. \tag{A.21}$$

In order to prove that the new step size scheme is more effective and feasible than the original step construction method, the one-dimensional simulations in low SNR are performed again, then the methods with different step sizes are adopted to process the echo data with the same iterations.

Fig. 17 is the results of the two schemes, where Fig. 17(a) is the result of $\alpha_k = (\frac{\sum \boldsymbol{d}_k \cdot \boldsymbol{d}_k}{\sum \boldsymbol{d}_{k-1} \cdot \boldsymbol{d}_{k-1}})^{\frac{1}{2}}$, and Fig. 17(b) is the result of $\alpha_k = \frac{\sum \boldsymbol{g}_k \cdot \boldsymbol{g}_{k-1}}{\sum \boldsymbol{g}_{k-1} \cdot \boldsymbol{g}_{k-1}}$. From Fig. 17(b), we can find that the result using the old construction way of α_k has no more obvious acceleration than the one-order vector extrapolation shown in Fig. 7(c). However, by using the new α_k in second-order extrapolation, the acceleration is improved further.

References

[1] A. K. Lohner, “Improved azimuthal resolution of forward looking SAR by sophisticated antenna illumination function design,” *IEE Proc.- Radar, Sonar Navigat.*, vol. 145, no. 2, pp. 128–134, Apr. 1998.

[2] J. Balke, “Field test of bistatic forward-looking synthetic aperture radar,” in *Proc. IEEE Int. Radar Conf.*, Arlington, VA, USA, May 2005, pp. 424–429.

[3] W. Pu *et al.*, “Motion errors and compensation for bistatic forward-looking SAR with cubic-order processing,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 6940–6957, Dec. 2016.

[4] W. Pu, W. Li, J. Wu, Y. Huang, J. Yang, and H. Yang, “An azimuth-variant autofocus scheme of bistatic forward-looking synthetic aperture radar,” *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 689–693, May 2017.

[5] W. Li, Y. Yang, J. Huang, J. Kong, and L. Wu, “An improved Radon-transform-based scheme of Doppler centroid estimation for bistatic forward-looking sar,” *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 2, pp. 379–383, Mar. 2011.

[6] W. Li, J. Yang, Y. Huang, and L. Kong, “Third-order doppler parameter estimation of bistatic forward-looking SAR based on modified cubic phase function,” *IEICE Trans. Commun.*, vol. 95, no. 2, pp. 581–586, Jan. 2012.

[7] T. Sutor, F. Witte, and A. Moreira, “New sector imaging radar for enhanced vision: SIREV,” *Proc. SPIE*, vol. 3364, 1998, pp. 39–47.

[8] S. Dai and W. Wiesbeck, “High resolution imaging for forward looking SAR with multiple receiving antennas,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Honolulu, HI, USA, Jul. 2000, vol. 5, pp. 2254–2256.

[9] W. Li, J. Yang, and Y. Huang, “Keystone transform-based space-variant range migration correction for airborne forward-looking scanning radar,” *Electron. Lett.*, vol. 48, no. 2, pp. 121–122, Jan. 2012.

[10] D. R. Wehner, High Resolution Radar, vol. 44. Norwood, MA, USA: Artech House, 1995.

[11] X. Zhang, L. Xu, L. Xu, and D. Xu, “Direction of departure (DOD) and direction of arrival (DOA) estimation in MIMO radar with reduced-dimension music,” *IEEE Commun. Lett.*, vol. 14, no. 12, pp. 1161–1163, Dec. 2010.

[12] S. C. Chan and H. H. Chen, “Uniform concentric circular arrays with frequency-invariant characteristics—Theory, design, adaptive beamforming and DOA estimation,” *IEEE Trans. Signal Process.*, vol. 55, no. 1, pp. 165–177, Jan. 2007.

[13] X. Mestre and M. Á. Lagunas, “Modified subspace algorithms for DOA estimation with large arrays,” *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 598–614, Feb. 2008.

[14] W. X. Sheng and D. G. Fang, “Angular superresolution for phased antenna array by phase weighting,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, no. 4, pp. 1450–1814, Oct. 2001.

[15] Y. Zhang, Y. Zhang, W. Li, Y. Huang, and J. Yang, “Super-resolution surface mapping for scanning radar: Inverse filtering based on the fast iterative adaptive approach,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 127–144, Jan. 2018.

[16] C. S. Miller, “Enhanced angle resolution in scanning beam systems,” in *Proc. IEEE Aerosp. Appl. Conf.*, Aspen CO, USA, Feb. 1995, vol. 1, pp. 333–341.

[17] J. Guan, J. Yang, Y. Huang, and W. Li, “Angular super-resolution algorithm based on maximum entropy for scanning radar imaging,” in *Proc. IEEE Geosci. Remote Sens. Symp.*, Jul. 2014, pp. 3057–3060.

[18] J. Guan, J. Yang, Y. Huang, and W. Li, “Maximum *a posteriori* based angular superresolution for scanning radar imaging,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 3, pp. 2389–2398, Jul. 2014.

[19] K. Tan, W. Li, J. Pei, Y. Huang, and J. Yang, “An I/Q-channel modeling maximum likelihood super-resolution imaging method for forward-looking scanning radar,” *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 6, pp. 863–867, Jun. 2018.

[20] Y. Zhang, W. Li, Y. Zhang, Y. Huang, and J. Yang, “A fast iterative adaptive approach for scanning radar angular superresolution,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 11, pp. 5336–5345, Nov. 2015.

[21] Y. Zhang, Y. Zhang, Y. Huang, W. Li, and J. Yang, “Angular superresolution for scanning radar with improved regularized iterative adaptive approach,” *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 6, pp. 846–850, Jun. 2016.

[22] Y. Zha, Y. Huang, and J. Yang, “Augmented Lagrangian method for angular super-resolution imaging in forward-looking scanning radar,” *J. Appl. Remote Sens.*, vol. 9, no. 1, pp. 1–15, 2015.

[23] H. M. Chen, M. Li, Z. Wang, Y. Lu, P. Zhang, and Y. Wu, “Sparse super-resolution imaging for airborne single channel forward-looking radar in expanded beam space via l_p regularisation,” *Electron. Lett.*, vol. 51, no. 11, pp. 863–865, 2015.

[24] L. Liu, X. Dong, J. Zhu, and D. Zhu, “An improved adaptive regularization method for forward looking azimuth super-resolution of a dual-frequency polarized scatterometer,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 6, pp. 2145–2159, Jun. 2016.

[25] J. M. Bioucas-Dias and M. A. T. Figueiredo, “A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration,” *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2992–3004, Dec. 2007.

[26] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

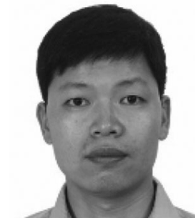
[27] K. Tan, W. Li, Y. Huang, and J. Yang, “Angular resolution enhancement of real-beam scanning radar base on accelerated iterative shinkage/thresholding algorithm,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Beijing, China, 2016, pp. 929–932.

[28] M. A. Richards, *Fundamentals of Radar Signal Processing*. New York, NY, USA: McGraw-Hill, 2005.

[29] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 586–597, Dec. 2007.

[30] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring,” in *Proc. IEEE Int. Conf. Speech Signal Process.*, 2009, pp. 693–696.

[31] D. S. Biggs and M. Andrews, “Acceleration of iterative image restoration algorithms,” *Appl. Opt.*, vol. 36, no. 8, pp. 1766–1775, 1997.



Wenchao Li (Member, IEEE) received the B.S. and M.S. degrees in information engineering from the Chengdu University of Technology, Chengdu, China, in 2005 and 2007, respectively, and the Ph.D. degree in electronic engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, in 2012.

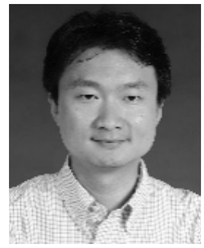
From 2016 to 2017, he was a Visiting Scholar with The Ohio State University, Columbus, OH, USA. He is currently an Associate Research Fellow with the School of Information and Communication Engineering, UESTC. His research interests include radar forward-looking imaging, synthetic aperture radar imaging, and superresolution.

Meihua Niu (Student Member, IEEE) received the B.S. degree in electronic information science and technology from Yunnan University, Kunming, China, in 2017. She is currently working toward the M.S. degree in signal and information processing with the University of Electronic Science and Technology of China, Chengdu, China.

Her research interests include radar signal processing and inverse problem in radar imaging.

Yongchao Zhang (Member, IEEE) received the B.S. degree in electronic information engineering from Hainan University, Haikou, China, in 2011, and the Ph.D. degree in electronic engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2017.

From 2016 to 2017, he was a Visiting Student with Lund University, Lund, Sweden. He is currently an Associate Research Fellow with the School of Information and Communication Engineering, UESTC. His research interests include array signal processing and inverse problem in radar applications.



Yulin Huang (Member, IEEE) received the B.S. and Ph.D. degrees in electronic engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2002 and 2008, respectively.

From 2013 to 2014, he was a Visiting Researcher with the University of Houston, Houston, TX, USA. Since 2016, he has been a Professor with the School of Electronic Engineering, UESTC. His research interests include synthetic aperture radar imaging and target detection and recognition.

Jianyu Yang (Member, IEEE) received the B.S. degree from the National University of Defense Technology, Changsha, China, in 1984, and the M.S. and Ph.D. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 1987 and 1991, respectively.

In 2005, he visited the Massachusetts Institute of Technology, Cambridge, MA, USA. Since 1999, he has been a Professor with the School of Electronic Engineering, UESTC. His research interests include radar forward-looking imaging, synthetic aperture radar imaging, and superresolution.