Abstract—This paper proposes a compressive sensing method for the phased array weather radar (PAWR), which is capable of three-dimensional observation with high spatial resolution in 30 seconds. Because of the large amount of observation data, which is more than 1 gigabyte per minute, data compression is an essential technology to operate PAWR in the real world. Even though many conventional studies applied compressive sensing (CS) to weather radar measurements, their reconstruction quality should be further improved. To this end, we define a new cost function that expresses prior knowledge about weather radar measurements, i.e., local similarities. Since the cost function is convex, we can derive an efficient algorithm based on the so-called convex optimization techniques, in particular simultaneous direction method of multipliers (SDMM). Simulation results show that the proposed method outperforms the conventional methods for real observation data with improvement of 4% in the normalized error.

I. INTRODUCTION

Extreme weather events, including local torrential rainfall, thunderstorms, blasts of wind, and hailstorms, are increasing recently [1]. Such phenomena are caused by rapid growth of cumulonimbus within 30 minutes. This quick growth means that it is not useful to use conventional radars with parabolic antennas, because such radars need 10 minutes to scan three-dimensional (3D) structure. Further, conventional radars can only observe elevation angles from approximately 10 to 20 degrees. To resolve these problems, phased array weather radar (PAWR) was developed. This radar is capable of scanning three dimensional data in 30 seconds exploiting digital beam forming technology without mechanical movements [2], [3].

PAWR can observe a range of 60km at the sampling interval of 100 meters. This means that the acquired data consist of 600 measurements for each direction. This data is observed at 300 and 110 points for azimuth and elevation directions, respectively. PAWR observes 13 aspects of the atmosphere, including the amount of rainfall. The description length of each data is normally two bytes. Thus, the data size for every single revolution of PAWR is $13 \times 600 \times 300 \times 110 \times 2 = 514.8$ megabyte, which results in a large amount of data, more than 1 gigabyte every minute and 1 terabyte every day. This makes it difficult to operate PAWRs in the real world and data compression is an essential technology for PAWR.

The recent advancing technique for this purpose is compressed sensing (CS) [4]–[6]. Under an assumption that the target signal can be sparsely represented by a certain transform (such as discrete cosine or wavelet transforms), compressed sensing enables us to acquire only a few samples without a significant loss of target information. CS was first applied to hard target radars [7]–[9]. Later on, CS has been applied to weather radars [10]. Yu et al. exploited CS to the refractivity retrieval based on the so-called Lasso estimator [11]. Mishra et al. assumed that matrices consisting of weather data are the approximate of low-rank ones and, based on the assumption, proposed a reconstruction method from randomly selected two-dimensional measurements for a single elevation angle [12]. Shimamura et al. exploited random Gaussian sensing matrix to one-dimensional vector along each range direction and reconstructed data using sparsity of coefficients of discrete cosine or wavelet transforms (DCT or DWT) [13].

To further improve the reconstruction quality, this paper proposes a novel compressed sensing method. We uniformly and randomly select the two dimensional PAWR measurements for each elevation angle to compress the data. Then, we reconstruct the data using the following three sorts of knowledge as follows. First, the observation error should be minimized. Second, the difference between adjacent measurements should be sparse. Third, the difference between adjacent DCT coefficients should be sparse. The third knowledge was successfully exploited in the movie reconstruction [14] and is transformed to the PAWR measurements reconstruction. We define a cost function so that the three sorts of prior knowledge are incorporated and reconstruct the two-dimensional data by minimizing the cost function. Since the cost function is convex, we can derive an efficient algorithm based on one of the convex optimization techniques. Simulations results show that the proposed method outperforms the conventional methods for real measurements obtained by a PAWR equipped in Osaka prefecture, Japan.

The rest of the present paper is organized as follows. Section
II prepares mathematical preliminaries. Section III formulates our random sensing scheme. In section IV, we define the cost function for two-dimensional reconstruction from compressed data. Section V proposes the reconstruction algorithm based on a convex optimization technique. Section VI is devoted to simulations. Section VII concludes the paper.

II. MATHEMATICAL PRELIMINARIES

The vec is a function that converts a matrix $X$ into a vector $x$ by stacking columns of $X : x = \text{vec}(X)$. The inverse conversion is denoted by $\text{mat} : X = \text{mat}(x)$. The Kronecker products $A \otimes B$ for matrices $A$ and $B$ is defined by

$$A \otimes B = \begin{pmatrix} a_{1,1}B & a_{1,2}B & \cdots \\ a_{2,1}B & a_{2,2}B & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}.$$ Using this operator, $AXB$ is denoted by $\text{mat}((B^T \otimes A)\text{vec}(X))$, where $B^T$ is a transpose of $B$. It holds that $(A_1 \otimes B_1)(A_2 \otimes B_2) = (A_1 A_2) \otimes (B_1 B_2)$. A class of lower semicontinuous convex functions from $R^N$ to $]-\infty, \infty]$ is denoted by $\Gamma_0(R^N)$. The proximity operator is defined for $f \in \Gamma_0(R^N)$ as

$$\text{prox}_f(x) = \arg\min_{u \in R^N} f(u) + \frac{1}{2} \|u - x\|^2.$$ For example, if $f(x) = \lambda \|x\|_1 = \lambda \sum_{n=1}^{N} |x_n|$ with a positive real number $\lambda$, proximity operator is given by $\text{prox}_f(x) = (\text{softThresh}(x_n))_{n=1}^{N}$ where

$$\text{softThresh}(x) = \begin{cases} x - \lambda, & (x \geq \lambda), \\ 0, & (-\lambda < x < \lambda), \\ x + \lambda, & (x \leq -\lambda), \end{cases}$$

and $x_n$ is $n$-th element of $x$. For every $k \in \{1, \ldots, K\}$, let $g_k \in \Gamma_0(R^{N_k})$ and let $L_k \in R^{N_k \times N}$. Assume that

$$L_0 g_i \in \text{ri dom } g_1 \ldots, L_K g_i \in \text{ri dom } g_K \ (q \in R^K)$$

where ri $C$ stands for the relative interior of a nonempty set $C$ [15], and $\sum_{k=1}^{K} g_k(L_k x) \rightarrow \infty$ as $\|x\| \rightarrow \infty$,

$$Q = \sum_{k=1}^{K} L_k^T L_k$$

is invertible. The problem of finding

$$\arg\min_{x \in R^N} \sum_{k=1}^{K} g_k(L_k x) \tag{1}$$

can be solved by the following algorithm. Step 2 updates each variable to minimize each term $g_k(L_k x)$. These updated variables are combined and updated in Step 3 to minimize the entire cost function in (1).

Algorithm 1: SDMM

1. Set appropriate initial values for $x$ and $t_k \ (k = 1 \sim K)$

   $$s_k \leftarrow L_k x$$

   $$r_k \leftarrow \text{prox}_{g_k}(s_k + t_k)$$

   $$t_k \leftarrow t_k + s_k - r_k$$

3. Update $x$ by $Q^{-1} \sum_{k=1}^{K} L_k^T (r_k - t_k)$

4. Go to 2 until a stopping condition is met

III. DATA COMPRESSION BY RANDOM SELECTION

The phased array weather radar captures thirteen parameters during each scan. Among them, we focus on the reflection intensity, which indicates the amount of rainfall. Other parameters can be handled in a similar way. The conventional study [13] exploited a random Gaussian matrix to compress one dimensional vector along range direction for every azimuth and elevation angles. To reconstruct the vector, not only the compressed measurements but also the random Gaussian matrix are needed. This means that we need to transfer both the measurements and the matrix, which is a large volume data. To reduce the data amount, we can use a fixed random Gaussian matrix for every one-dimensional data. This sampling scheme may decrease the reconstruction quality. To solve these problems, we adopt the random selection sampling scheme as well as [12]. The reflection intensity of radar is 3D data, whose dimension is $N_H = 600$ in the range direction, $N_V = 300$ in the azimuth angle, and $N_W = 110$ in the elevation angle. One slice within the three dimensional data for a fixed elevation angle is extracted and forms the target two dimensional data $X_0$ of the size $N_H \times N_V$. The total number of data is $N = N_H N_V$. $A$ is the function that randomly selects $N_{\alpha}(< N)$ elements out of $N$ data and preserve them, while the rest of them are converted to zero, yielding the observation data $Y$:

$$Y = A(X_0). \tag{2}$$

The compression rate $\alpha$ is defined by $N_{\alpha}/N$. Fig. 1 shows original observed data and compressed data ($\alpha = 25\%$). The random pattern changes for every elevation angle so that the reconstruction quality gets higher. The selected position is transferred only using four bits, which is not a heavy burden for data transmission. The next section proposes a method for estimating $X_0$ from the compressed data $Y$ and the random selected position information embedded in the function $A$.

IV. 2-DIMENSIONAL RECONSTRUCTION BY UTILIZING LOCAL SIMILARITY

Since the reconstruction problem is ill-posed, some prior information is necessary to reconstruct the data at a high-quality. To this end, we exploit the following prior knowledge. That is, rain falling areas continuously exist. Therefore, differences of neighboring data along the range or azimuth directions are small in and out of the area, but large on the boundary parts.
In other words, the differences can be supposed to be sparse. Moreover, according to research about movie reconstruction, if neighboring data are similar, so are the DCT coefficients [14]. Hence, we propose to reconstruct the two-dimensional data \( X_0 \) by minimizing the cost function defined below.

Let \( D_H \) be an \((N_H - 1) \times N_H\) matrix defined as
\[
D_H = \begin{pmatrix}
-1 & 1 & 0 & \ldots & 0 \\
0 & -1 & 1 & \ldots & 0 \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
0 & \ldots & 0 & -1 & 1 \\
\end{pmatrix}.
\]

The difference matrix \( D_V \), which is \((N_V - 1) \times N_V\), is defined in the same way. Then, the anisotropy total variation \( TV_1(X) \) is defined as
\[
TV_1(X) = \|D_H X\|_1 + \|X D_V^T\|_1.
\]

Sparsifying matrices of the sizes \( N_H \times N_H \) and \( N_V \times N_V \), such as one-dimensional DCT, are denoted by \( C_H \) and \( C_V \), respectively. The cost function is then defined as
\[
f(X) = \frac{1}{2} \|A(X) - Y\|_F^2 + \lambda_1 TV_1(X) \\
+ \lambda_2 \|C_H X D_V^T\|_1 + \lambda_3 \|D_H X C_V^T\|_1,
\]
where \( \| \cdot \|_F \) is the Frobenius norm. In the function, third term is the distance difference of transformed coefficients of azimuth vector, and fourth term is the azimuth difference of transformed coefficients of distance vector. Finally, the entire data is reconstructed by finding the solution to the problem:
\[
\hat{X} = \arg\min_{X \in \mathbb{R}^{N_H \times N_V}} f(X).
\]

V. RECONSTRUCTION ALGORITHM

To derive a reconstruction algorithm, we transform the cost function \( f(X) \) into a formula that can be applied to the SDMM. First, the function \( f(X) \) is converted into a vector format \( \vec{f}(x) \). The functions \( g_1 \sim g_5 \) are defined as
\[
g_1(z_1) = \frac{1}{2} \|A z_1 - y\|^2, \\
g_2(z_2) = \lambda_1 \|z_2\|_1, g_3(z_3) = \lambda_1 \|z_3\|_1, \\
g_4(z_4) = \lambda_2 \|z_4\|_1, g_5(z_5) = \lambda_3 \|z_5\|_1,
\]
where \( y = \text{vec}(Y) \) and \( A \) is an \( N \times N \) diagonal matrix whose elements corresponding to the selected measurements by the function \( A \) are one and zero otherwise. Further, we define \( L_1 = I_V \otimes I_H, L_2 = I_V \otimes D_H, L_3 = D_V \otimes I_H, L_4 = D_V \otimes C_H, \) and \( L_5 = C_V \otimes D_H \). Then, (3) can be converted into the form of
\[
\hat{X} = \text{mat} \left\{ \arg\min_{x \in \mathbb{R}^N} \sum_{k=1}^{5} g_k(L_k x) \right\}.
\]

The functions \( g_1 \sim g_5 \) satisfy the conditions described in Section 2. Hence, (3) is now in the form of (1), thus we can solve the problem by using SDMM.

We note here that the size of matrices appearing in (5) is mainly \( N \times N \), where \( N = 180,000 \) in the present case. To store such huge matrices requires 64.8 gigabyte if two bytes are assigned for each element. To be able to handle such huge matrices, we need further manipulations. In particular, let us describe how to compute \( \text{prox}_{g_1} \) and the application of \( Q^{-1} \).

First, \( \text{prox}_{g_1} (x) \) is given by [16]
\[
\text{prox}_{g_1}(x) = (I_V \otimes I_H + A^T A)^{-1}(x + A^T y).
\]

By a simple manipulation, we can show that this operator is computed by
\[
\text{prox}_{g_1}(x) = \text{vec}(A(X + Y)),
\]
where \( A \) is a function that divide the matrix element corresponding to the selected measurement by \( A \) by two, and does not change the matrix element otherwise.

Next, the application of \( Q^{-1} \) to a vector can be computed as follows. In our case, \( Q \) is in the form:
\[
Q = I_V \otimes I_H + 2 I_V \otimes (D_H^T D_H) + 2 (D_V^T D_V) \otimes I_H.
\]

It is analytically shown [17] that the eigenvalue \( \delta_n \) and the eigenvector \( u_n \) \((n = 0, \ldots, N_H - 1)\) of matrix \( D_H^T D_H \) in (7) are given by
\[
\delta_n = 2 - 2 \cos \left( \frac{n \pi}{N_H} \right),
\]
\[
u_n = \begin{cases} 
(1, 1, \ldots, 1)/\sqrt{N_H}, & (n = 0), \\
c_n D_H^T v_n, & (n = 1, \ldots, N_H - 1),
\end{cases}
\]
where,
\[
v_n = (\sin(n \pi/N_H) \ldots \sin((N_H - 1)n \pi/N_H))^T.
\]

The eigenvalues and eigenvectors for \( D_V^T D_V \) can also be obtained similarly. Then, eigenvalue decompositions \( D_H^T D_H = U_H \Delta_H U_H^T \) and \( D_V^T D_V = U_V \Delta_V U_V^T \) are obtained and applied to (7). Hence, \( Q \) is now expressed in the form of
\[
Q = (U_V \otimes U_H)(I_V \otimes I_H + 2 I_V \otimes \Delta_H + 2 \Delta_V \otimes I_H)(U_V^T \otimes U_H^T).
\]

This is nothing but the eigenvalue decomposition of \( Q \). Its eigenvalue \( \sigma_j \) \((j = 0, \ldots, N - 1)\) is given by
\[
\sigma_j = 9 - 4 \cos \left( \frac{n \pi}{N_H} \right) - 4 \cos \left( \frac{n \pi}{N_V} \right) \equiv \sigma_{n,m},
\]
where \( m = \lfloor j/N \rfloor \), \( n = j - mN \) and \( | r | \) is the greatest integer not exceeding \( r \). Therefore, \( Q^{-1} \) is expressed by

\[
Q^{-1} = (U_{V} \otimes U_{H}) \text{diag}(1/\sigma_{0}, \ldots, 1/\sigma_{N-1}) (U_{V}^{\top} \otimes U_{H}^{\top}). \tag{8}
\]

This expression allows us to compute the application of \( Q^{-1} \) to a vector \( \phi \) in the matrix form, as

\[
Q^{-1} \phi = \text{vec}(U_{H} \Sigma (U_{H}^{\top} \Phi U_{V}) U_{V}^{\top}),
\]

where \( \Sigma(\cdot) \) is a function that multiply \( 1/\sigma_{n,m} \) to the \( n,m \)th element of matrix and \( \Phi = \text{mat}(\phi) \).

From the discussions above, we propose the two-dimensional PAWR measurements reconstruction algorithm as follows:

**Algorithm 2: PAWR Data Reconstruction Using SDMM**

1. Set initial matrices for \( X \) and \( T_{k} \) (\( k = 1 \sim 5 \))
   
   \[
   S_{k} \leftarrow \text{mat}(L_{k} \text{vec}(X))
   \]
   
   \[
   R_{k} \leftarrow \text{prox}_{\frac{1}{\Omega}(S_{k} + T_{k})}
   \]
   
   \[
   T_{k} \leftarrow T_{k} + S_{k} - R_{k}
   \]

2. Compute \( \Phi = \text{mat}(\{ \sum_{k=1}^{5} L_{k}^{T} \text{vec}(R_{k} - T_{k}) \}) \)

3. Update \( X \) by \( U_{H} \Sigma (U_{H}^{\top} \Phi U_{V}) U_{V}^{\top} \)

4. Go to 2 until a stopping condition is met

VI. SIMULATIONS

To show the effectiveness of the proposed method, we conducted simulations using real data obtained by a PAWR equipped at Osaka University in Suita Campus, Japan, on June 19th, 2013 (Data 1) and March 30th, 2014 (Data 2). From the entire three-dimensional data, a single slice of 10th elevation angle (approximately 8.1 degree from the horizontal line) was extracted and the target measurements \( X_{0} \) were set. The observed data \( Y \) is generated by uniform and random selection of \( \alpha = 25\% \) out of the entire measurements in \( X_{0} \). We input \( Y \) and the selected positions into Algorithm 2, which reconstructed two-dimensional data. The sparsifying transform is DCT in this simulation. The regularization parameters are \( \lambda_{1} = 0.06 \), \( \lambda_{2} = 0.3 \) and \( \lambda_{3} = 0.1 \). These parameters were determined in an ad hoc manner. For comparison, we also reconstructed the entire measurements by Mishra’s and Shimamura’s methods in [12] and [13], respectively. Note that Mishra’s method compresses the entire data in the same way as our method while Shimamura’s method exploits the Gaussian random matrix to compress the data. In both cases, the compression rate was 25%.

Figs. 2 and 3 show the simulation results for Data 1 and Data 2, respectively. Within these figures, (a), (b) and (c) show the original observed measurements, its magnification of the rectangle area in (a), and the compressed measurements, respectively. Figures (d), (e), and (f) show the reconstructed measurements by the proposed, Mishra’s, and Shimamura’s methods, respectively. Figures (g), (h), and (i) show the magnification of the rectangle of (d), (e), and (f), respectively. We can see that the proposed method reconstructed the original observed measurements better than the other methods especially for high dBZ areas indicated by yellow or red in Figs. 2 and 3. The radial artifacts can be seen in (f) and (i), but not in (d) and (g). The normalized errors for the reconstruction \( \hat{X} \) computed by \( \| \hat{X} - X_{0} \|_{F} / \| X_{0} \|_{F} \) are shown below (d), (e), and (f) in parentheses. The proposed method could improve the error by approximately 4% compared to Mishra’s method and 6% compared to Shimamura’s method.

These simulations are conducted using Matlab on iMac (OS 10.10, Intel Core i5, 270GHz, 8GB). The time for reconstruction by the proposed method was 146 seconds, while those for the Mishra’s and Shimamura’s methods were 79 and 237 seconds, respectively. Thus, proposed method is 1.6 times faster than Shimamura’s method, but 1.8 times slower than Mishra’s methods. Note that the proposed method can be further accelerated by exploiting the structure of the vector \( v_{n} \) by using a discrete sine transform (DST).

VII. CONCLUSION

This paper proposed a compressive sensing method for the phased array weather radar (PAWR). First, we defined a new cost function that expresses prior knowledge about weather radar measurements, i.e., local similarities of measurements themselves and sparsifying transformed coefficient vectors. Since the cost function is convex, an efficient algorithm was derived based on the so-called convex optimization techniques, in particular simultaneous direction method of multipliers (SDMM). Simulation results showed that the proposed method outperforms the conventional methods for real observation data with approximately 4% normalized error improvements with faster computation. To further accelerate the proposed method is one of our future works.

ACKNOWLEDGMENT

The authors are grateful to Prof. Jeremy White in Ritsumeikan University for improving English expressions.

REFERENCES


(a) Original observed data  
(b) Magnification of rectangle in (a)  
(c) 25% random selected data  

(d) Proposed method (13.64%)  
(e) Mishra’s Method (17.58%)  
(f) Shimamura’s Method (19.71%)  

(g) Magnification of rectangle in (d)  
(h) Magnification of rectangle in (e)  
(i) Magnification of rectangle in (f)  

Fig. 2. Simulation results for Data 1 obtained on June 19th, 2013.
Fig. 3. Simulation results for Data 2 obtained on March 30th, 2014.


