An Efficient Sliding Window Processing for the Covariance Matrix Estimation

Yoshinari Iwakura [1], Junichiro Suzuki [2], Hiroyoshi Yamada [3],
Yoshio Yamaguchi [3], Masahiro Tanabe [2] , and Yoshikazu Shoji [2]
[1] Graduate School of Science and Technology, Niigata University.
Ikarashi 2-8050, Niigata-shi, 950-2181 Japan, yiwakura@wave.ie.niigata-u.ac.jp
[2] Radar & Sensor Systems Engineering Dept., Komukai Operations, Toshiba Corporation
1, Komukai Toshiba-cho, Saiwai-ku, Kawasaki-shi, 212-8581 Japan
{junichiro.suzuki, masahiro.tanabe, yoshikazu1.shoji}@toshiba.co.jp
[3] Faculty of Engineering, Niigata University
Ikarashi 2-8050, Niigata-shi, 950-2181 Japan, {yamada, yamaguchi}@ie.niigata-u.ac.jp

1. Introduction

The sliding window method is applied to the covariance matrix estimation of space-time adaptive processing (STAP) [1] and multiple-input multiple-output (MIMO) [2]. These are powerful techniques for the improvement of the target detection accuracy in airborne radar systems. In the conventional sliding window method, it is necessary to calculate a covariance matrix every time from some training samples for each cell under test (CUT) [3], [4]. Since computational load for covariance matrix estimation increases depending on degrees-of-freedom (DOF) and the number of CUTs of radar system, it is difficult to estimate it accurately in real-time.

In this report, we propose an efficient sliding window method. By using the proposed method, the series of covariance matrices of the CUT are calculated by only adding and subtracting recursion from the previous covariance matrices. We evaluate the proposed method from the viewpoint of the computational time and verify its effectivity.

2. Conventional Sliding Window Method

The spatial steering vector $\boldsymbol{s}_\theta$ and temporal steering $\boldsymbol{s}_{f_d}$ are defined as follows

$$\boldsymbol{s}_\theta = \left[\; 1, e^{j\frac{2\pi d}{\lambda}\sin\theta}, \ldots, e^{j(N-1)\frac{2\pi d}{\lambda}\sin\theta} \; \right]^T, \tag{1}$$

$$\boldsymbol{s}_{f_d} = \left[\; 1, e^{j2\pi\frac{f_d}{f_p}}, \ldots, e^{j(M-1)2\pi\frac{f_d}{f_p}} \; \right]^T, \tag{2}$$

where N and M are the number of elements and pulses, respectively. ^{T} is transpose. λ and d are the wavelength and the element spacing, respectively. θ is the spatial angle. Besides, f_d and f_p are the Doppler frequency and pulse repetition frequency (PRF). By using the spatial and temporal steering vectors, space-time steering vector $\boldsymbol{s}$ for STAP is defined as follows

$$\boldsymbol{s} = \boldsymbol{s}_{f_d} \otimes \boldsymbol{s}_\theta, \tag{3}$$

where $\otimes$ is the Kronecker product.

Figure 1 shows the basic concept of the pulse radar. As shown in this figure, the transmitted pulse is reflected by the target and ground surface. Distance to the target can be estimated by measuring the round trip time of transmitted pulse. In addition, the received signal vector $\boldsymbol{r}_i$ for the ith cell is defined as follows

$$\boldsymbol{r}_i = \boldsymbol{\alpha}(i)\boldsymbol{s} + \boldsymbol{n}(i), \tag{4}$$

where $\boldsymbol{\alpha}(i)$ is the complex amplitude vector of each incoming signal and $\boldsymbol{n}(i)$ is the additive Gaussian noise vector.

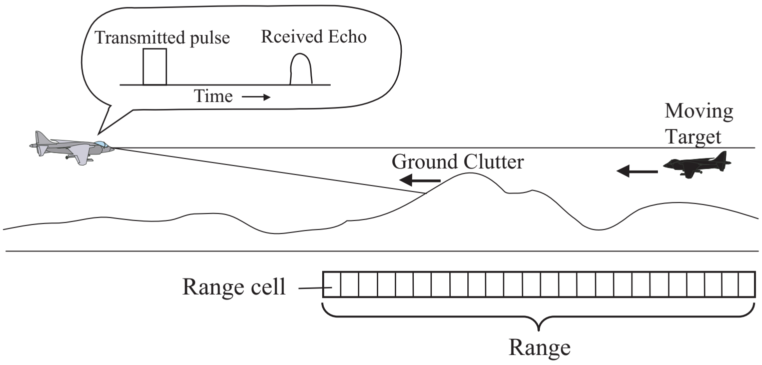

Figure 1: The concept of the pulse radar

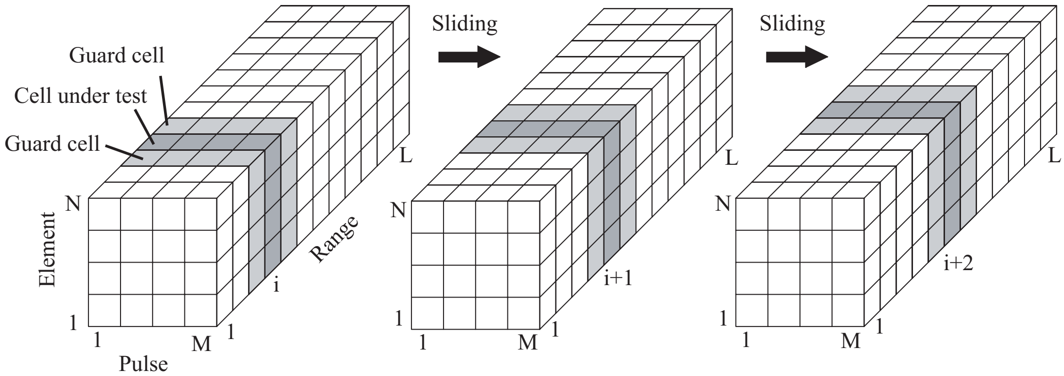

Figure 2: The concept of the the conventional sliding window method

Figure 2 shows the concept of covariance matrix estimations with the conventional sliding window method. In this figure, an L range cells example having 1 sliding size is shown. The cells in the both sides of the CUT are called guard cells.

The covariance matrix $\boldsymbol{R}^{[i]}$ for the ith CUT can be estimated by

$$\boldsymbol{R}^{[i]} = \frac{1}{K}\{\boldsymbol{r}_1\boldsymbol{r}_1^H + \cdots + \boldsymbol{r}_{i-2}\boldsymbol{r}_{i-2}^H + \boldsymbol{r}_{i+2}\boldsymbol{r}_{i+2}^H + \cdots + \boldsymbol{r}_L\boldsymbol{r}_L^H\}, \tag{5}$$

where ^{H} is the conjugate transpose and K is the number of training samples. The CUT is the cell that is applied weight for the target detection.

It is considered that the desired target signal is possibly spread in several range cells, therefore the covariance matrix estimation should be done without the CUT and adjacent cells (*guard cell*) for preventing the mixture of the target signal components.

3. Proposed Sliding Window Method

3. 1 Efficient Sliding Window Processing

Figures 3 and 4 show the concept of covariance matrix estimation with the proposed sliding window method and its flow chart, respectively. First, all of the covariance matrices $\boldsymbol{R}_i$ ($i = 1, 2, \ldots, L$) are calculated by

$$\boldsymbol{R}_i = \boldsymbol{r}_i\boldsymbol{r}_i^H. \tag{6}$$

and stored in the memory before the covariance matrix $\boldsymbol{R}^{[i]}$ of the target CUT. Next step is the covariance matrix estimation for the CUT when we employ 1 guard cell in both sides, for example. In this case the first CUT can be defined by

$$\boldsymbol{R}^{[2]} = \frac{1}{K}\{\boldsymbol{R}_4 + \cdots + \boldsymbol{R}_L\}. \tag{7}$$

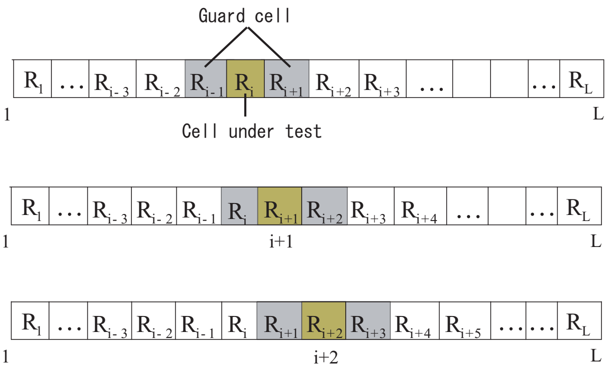

Figure 3: The concept of the proposed sliding window method

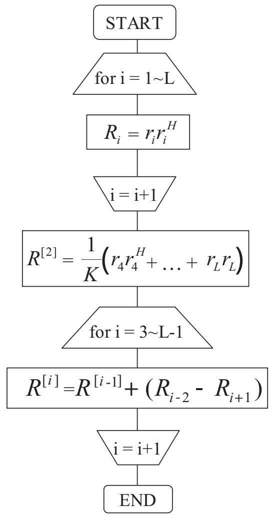

Figure 4: The flow chart of the proposed sliding window method

This covariance matrix is stored for the next covariance matrix estimation. The following covariance matrices can be estimated as follows

$$\boldsymbol{R}^{[i]} = \boldsymbol{R}^{[i-1]} + \{\boldsymbol{R}_{i-2} - \boldsymbol{R}_{i+1}\}. \tag{8}$$

3. 2 Grouping Range Cells

Furthermore, we introduce grouping range cells in order to reduce the computational load of the covariance matrix estimation. The above-mentioned estimation is applied after the grouping. The concept of the grouping range cells is shown in Fig. 5. In this example, the number of grouping range cells is 3.

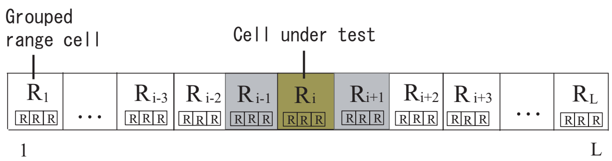

Figure 5: The concept of the grouping range cells

The covariance matrix $\boldsymbol{R}_i$ for each grouped range cell is defined by

$$\boldsymbol{R}_i = \frac{1}{G} \sum_{j=i\times G}^{(i+1)\times G} \boldsymbol{R}_j, \tag{9}$$

where G is the number of grouping range cells.

4. Simulation Results

In this section, we evaluated the computational time for covariance matrix estimations of the proposed method by computer simulation. As shown in Table 1, simulation condition was set to 8 elements , 8 pulses coherent processing interval (CPI), 512 CUTs, and 3 slide sizes. Simulation results were computed using Matlab® with *tic/toc* function by Intel® Core2 CPU 2.66 GHz with 4 GB RAM.

Figure 6 shows results of the computational time of the conventional and the proposed method. As shown in this figure, the computational time is almost flat for the number of training samples by the proposed method. In this simulation condition, the proposed method becomes more efficient than the conventional method when the number of training samples is 8 or more.

Figure 7 shows results of the computational time of the proposed method with the grouping and without the grouping. In this simulation condition, the computational time becomes about half by the grouping.

Table 1: Simulation Conditions

Number of degrees-of-freedom $(N \times M)$	8×8
Number of CUTs	512
Number of slide sizes	3

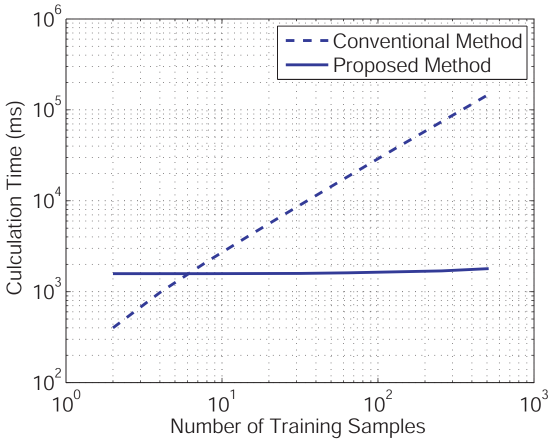

Figure 6: Number of training samples vs. calculation time

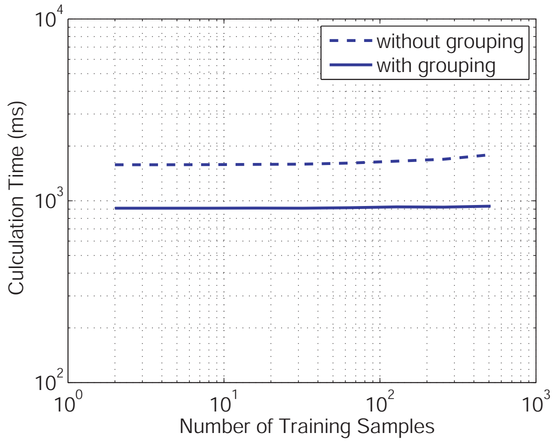

Figure 7: Number of training samples vs. calculation time

5. Conclusions

In this report, we proposed an efficient sliding window processing which calculates the series of covariance matrices by only adding and subtracting recursion with the stored data. We showed that the proposed method is more efficient than the conventional method, and confirmed the reduction of computational time by computer simulation.

References

[1] J. R. Guerci, *Space-Time Adaptive Processing for Radar*, Norwood, MA: Artech House, 2003.

[2] E. Fishler, A. Haimowich, R. Blum, L. Cimini, D. Chizhik, and R. Valenzuela, "MIMO radar: An idea whose time has come," in *Proc. IEEE radar Conf.* pp. 71-78, Apr., 2004.

[3] R. Klemm, *Principles of space-time adaptive processing*, IEE Press, London, U.K., 2002.

[4] J. Ward and S. M. Kogon, "Space-Time Adaptive Processing (STAP) for AMTI and GMTI Radar," tutorial slides, *CD of the Proc. IEEE radar Conf.* 26-29. Apr., 2004.