

Cyclic Shift Matrix—A New Tool for the Translation Matching Problem

Xiurui Geng[✉] and Weitun Yang[✉]

Abstract—For numerous applications in image registration, sub-pixel translation estimation is a fundamental task, and increasing attention has been given to methods based on image phase information. However, we have found that none of these methods is universal. In other words, for any one of these methods, we can always find some image pairs which will not be well matched. In this paper, by introducing the cyclic shift matrix (CSM), we present a new model for the translation matching problem and derive a least squares solution for the model. In addition, by repeatedly applying the CSM to the matching image, an iterative CSM method is proposed to further improve the matching accuracy. Furthermore, we show that the traditional phase-based matching algorithms can only achieve an exact solution when there is a cyclic shift relationship between the images to be matched. The proposed method is evaluated using simulated and real images and demonstrates a better performance in both accuracy and robustness compared with the state-of-the-art methods.

Index Terms—Cyclic shift matrix (CSM), phase correlation, sub-pixel translation.

I. INTRODUCTION

TO ADDRESS the image translation registration problem, traditional cross-correlation methods, such as normalized cross correlation and zero-normalized cross correlation, are time-consuming and often inaccurate in many cases [1], [2]. Because of this, the phase information-based matching method has begun to attract attention and is now widely used due to its accuracy and effectiveness [3]–[6]. Translation matching methods (TMMs) based on phase information are generally derived from the Fourier shift property [7]. Assuming a sub-pixel shift between two images $\mathbf{G}(x, y)$ and $\mathbf{H}(x, y)$ is x_0 and y_0 in the x - and y -directions, respectively, and the relationship can be expressed as

$$\mathbf{H}(x, y) = \mathbf{G}(x - x_0, y - y_0). \quad (1)$$

After the Fourier transformation (FT), the relationship is transformed into

$$\hat{\mathbf{H}}(u, v) = \hat{\mathbf{G}}(u, v) \exp\{-i(ux_0 + vy_0)\} \quad (2)$$

Manuscript received October 11, 2018; revised January 27, 2019 and June 11, 2019; accepted June 14, 2019. Date of publication July 15, 2019; date of current version October 31, 2019. This work was supported in part by the projects of the National Natural Science Foundation of China under Grant 41601402, in part by the National Major Scientific Instruments Development Project under Grant 41427805, and in part by High Resolution Satellite 5 Application Common Key Technology under Grant 30-Y20A28-9004-15/17. (*Corresponding authors:* Xiurui Geng; Weitun Yang.)

The authors are with the Institute of Electronics, Chinese Academy of Sciences, Beijing 100864, China, also with the Key Laboratory of Technology in Geo-Spatial Information Process and Application Systems, Chinese Academy of Sciences, Beijing 100864, China, and also with the University of the Chinese Academy of Sciences, Beijing 100049, China (e-mail: gengxr@sina.com.cn; tsubasalover@163.com).

Color versions of one or more of the figures in this article are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TGRS.2019.2923684

where $\hat{\mathbf{G}}(u, v)$ and $\hat{\mathbf{H}}(u, v)$ are the corresponding FT of $\mathbf{G}(x, y)$ and $\mathbf{H}(x, y)$, and u and v are the coordinates in the frequency domain. As can be seen, the pixel shifts of images in the spatial domain are transformed into the Fourier domain as linear phase differences [7]. The normalized cross-power spectrum matrix is defined as

$$\mathbf{P}(u, v) = \frac{\hat{\mathbf{G}}(u, v)\hat{\mathbf{H}}^*(u, v)}{|\hat{\mathbf{G}}(u, v)\hat{\mathbf{H}}^*(u, v)|} = \exp\{i(ux_0 + vy_0)\} \quad (3)$$

where $*$ stands for the complex conjugate, and it can be seen that the \mathbf{P} matrix is only related to the shifts.

Currently, the use of the \mathbf{P} matrix can be divided into two categories. The first class of method gives an estimate of x_0 and y_0 by finding the main peak location of the inverse FT (IFT) of the normalized cross-power spectrum matrix \mathbf{P} [8]–[10]. In the case of integer pixel shifts, the IFT of \mathbf{P} is a Dirac delta function centered on (x_0, y_0) [2], and it could be correctly captured by a simple IFT for \mathbf{P} . As for the sub-pixel shift situation, some methods introduce the procedure of interpolation to achieve sub-pixel precision [11]. However, these are easily affected by noise and other interference [12]. Accuracy could also be increased to $1/k$ pixel by the procedure of up-sampling using the factor k . Unfortunately, this becomes too much time-consuming when k is large. In [13], an improved method is proposed to increase the speed of the up-sampling: it first computes the initial shift estimate according to the conventional IFT method with an up-sampling factor $k = 2$, and then a much larger k is applied to the data in a 1.5×1.5 neighborhood around the initial estimate. It should be noted that this type of method is restricted by the value of k , and the precision could only reach $1/k$ pixel at most.

The other type of method directly uses the phase information of \mathbf{P} matrix to estimate the pixel shift. The phase shift angle of the matrix \mathbf{P} can be expressed as a linear function of the shift (x_0, y_0) , and thus, it represents a 2-D plane through the origin of the u - v coordinates. Based on this fact, many methods have been proposed [5], [14], [15]. The Stone method [14] uses the least squares adjustment to fit the 2-D plane after removing the high-frequency and small-magnitude spectral components. Since the phase shift angle is 2π wrapped in the frequency domain, only shifts less than one pixel can be measured using the Stone method [16]. Hoge [17] applied a singular value decomposition (SVD) to the \mathbf{P} matrix and transformed the 2-D phase unwrapping problem into two separate 1-D unwrapping problems. Then, the unwrapping phase information of the dominant rank-one approximation of the \mathbf{P} matrix is used to estimate (x_0, y_0) . In addition, since the phase information could be easily interrupted

by noise, some optimization strategies [18], [19] are introduced to the SVD method to further improve its performance. For example, the method in [18] proposes a robust extension to the SVD method using a so-called “projection” masking operator under the assumption that the noise is additive white Gaussian noise. And, in [20], a random sample consensus (RANSAC) [21], [22] algorithm is introduced to make the SVD method more reliable and effective.

These algorithms can achieve good results in many cases. However, they are not universal and cannot obtain good results in certain situations. In this paper, by introducing the cyclic shift matrix (CSM), we propose a new model for the image translation matching problem (TMP) and obtain a least squares solution for the model, named the CSM method. Interestingly, we found that the traditional TMM based on phase information could only get the exact solution when there was a cyclic shift relationship between images to be matched. In addition, an iterative cyclic shift method is presented and this is expected to further improve the estimation accuracy of TMM.

II. METHODS

In this section, we first introduce some basic properties of the CSM, followed by a detailed derivation of the CSM method. Then, the applicable condition of the traditional phase-based methods is given in a lemma. In addition, by repeated use of the CSM, an iterative version of CSM is presented. Details are stated as follows.

A. Cyclic Shift Model

In this paper, a cyclic shift matrix \mathbf{Q} is introduced to simulate the translation of images in the horizontal and vertical directions. The basic form of the \mathbf{Q} matrix is defined as

$$\mathbf{Q} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 1 \\ 1 & 0 & \cdots & \cdots & 0 \end{bmatrix} \quad (4)$$

which could be generated by cyclically moving each column of the identity matrix.

Using the \mathbf{Q}^T matrix to left multiply an image is equivalent to a cyclic shift of the image in the vertical direction, that is, the first line of the image becomes the second row, the second line becomes the third row, and so on. It is worth noting that due to the cyclic shift property of the \mathbf{Q} matrix, the last line of the original image is moved to the first row in the new image. Similarly, when image \mathbf{I} is right-multiplied by the \mathbf{Q} matrix, it results in a horizontal cyclic shift of the image. For example, a matrix \mathbf{I} of 3×5 size is shown in Fig. 1(a). When we multiply the left- and right-hand sides of the image by $\mathbf{Q}_{3 \times 3}^T$ and $\mathbf{Q}_{5 \times 5}$, respectively, the transformed image $\mathbf{I}' = \mathbf{Q}_{3 \times 3}^T \mathbf{I} \mathbf{Q}_{5 \times 5}$ can be obtained [Fig. 1(b)]. As can be seen, under the effect of the $\mathbf{Q}_{3 \times 3}^T$ and $\mathbf{Q}_{5 \times 5}$ matrix, the image is cyclically shifted down and to the right by a single pixel.

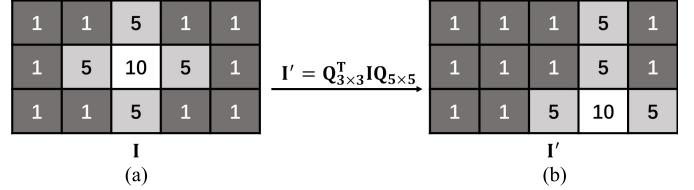


Fig. 1. Diagram of a CSM in an integer pixel shift situation. (a) Original image. (b) Transformed image.

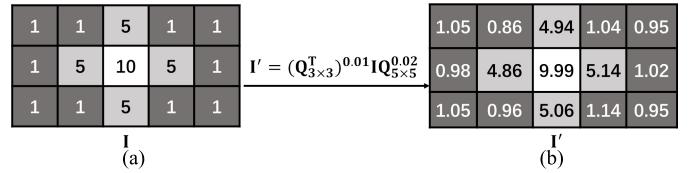


Fig. 2. Diagram of a CSM in a sub-pixel shift situation. (a) Original image. (b) Transformed image.

The above example shows that the cyclic shift of arbitrary integer pixels of an image can be precisely described by the CSM. In fact, we can also use the fractional order power of the \mathbf{Q} matrix to simulate a sub-pixel translation of an image with arbitrary precision. Assuming the eigenvalue and eigenvector matrix of the \mathbf{Q} matrix are \mathbf{D} and \mathbf{U} , respectively, the \mathbf{Q} matrix can be expressed as

$$\mathbf{Q} = \mathbf{UDU}^H \quad (5)$$

where the superscript H stands for a conjugate transpose. The fractional order power of the \mathbf{Q} matrix can be calculated as

$$\mathbf{Q}^x = \mathbf{UD}^x\mathbf{U}^H \quad (6)$$

where x can be any real number and \mathbf{D}^x can be obtained by simply calculating the x -power of its diagonal elements. For example, calculating the value of $\mathbf{Q}_{3 \times 3}^{0.01}$, we obtain

$$\mathbf{Q}_{3 \times 3}^{0.01} = \begin{bmatrix} 0.9999 & 0.0122 & -0.0120 \\ -0.0120 & 0.9999 & 0.0122 \\ 0.0122 & -0.0120 & 0.9999 \end{bmatrix}. \quad (7)$$

In fact, when the order of \mathbf{Q} is odd, it can be shown that \mathbf{Q}^x is still a real matrix for any real number x . Therefore, we assume that the number of both rows and columns of the image to be matched is odd in this paper. If the real image does not meet this condition, we can use image clipping or up-sampling to make it meet the condition. We also use matrix \mathbf{I} (Fig. 2) to illustrate the effect of matrix \mathbf{Q}^x when x is any real number. In Fig. 2, the original image \mathbf{I} [Fig. 2(a)] is transformed into \mathbf{I}' by multiplying its left- and right-hand sides with $(\mathbf{Q}_{3 \times 3}^T)^{0.01}$ and $\mathbf{Q}_{5 \times 5}^{0.02}$, respectively. Under the effect of the matrix multiplications, the image is cyclically shifted down and to the right by 0.01 and 0.02 pixels, respectively, and the generated image \mathbf{I}' , which is shown as Fig. 2(b), is also a real matrix. As can be seen, although the sub-pixel shift between images is very small, it can still be accurately reflected by the power of the \mathbf{Q} matrix.

Motivated by the above property of the \mathbf{Q} matrix, we can formulate the TMP as the optimization problem in the following:

$$\min_{x,y} f(x, y) = \min_{x,y} \|\mathbf{A} - (\mathbf{Q}_1^T)^x \mathbf{B} \mathbf{Q}_2^y\|_F \quad (8)$$

where \mathbf{A} and \mathbf{B} represent the image pair for registration, \mathbf{Q}_1 and \mathbf{Q}_2 are two CSMs, x and y are the pixel shifts we need to estimate, and $\|\cdot\|_F$ stands for the Frobenius norm. The dimensions of \mathbf{A} , \mathbf{B} , \mathbf{Q}_1 , and \mathbf{Q}_2 are $m \times n$, $m \times n$, $m \times m$, and $n \times n$, respectively. Since the \mathbf{Q} matrix is a real matrix, it is easy to verify that the following equations hold:

$$\mathbf{Q}^T = \mathbf{Q}^H = (\mathbf{Q}^H)^* = (\mathbf{U}\mathbf{D}^H\mathbf{U}^H)^* = \mathbf{U}^*\mathbf{D}\mathbf{U}^T. \quad (9)$$

Thus

$$(\mathbf{Q}^T)^x = \mathbf{U}^*\mathbf{D}^x\mathbf{U}^T. \quad (10)$$

Combining (6), (8), and (10), we can have

$$(\mathbf{Q}_1^T)^x \mathbf{B} \mathbf{Q}_2^y = \mathbf{U}_1^* \mathbf{D}_1^x \mathbf{U}_1^T \mathbf{B} \mathbf{U}_2 \mathbf{D}_2^y \mathbf{U}_2^H \quad (11)$$

where \mathbf{D}_1 and \mathbf{D}_2 are the eigenvalue matrices of \mathbf{Q}_1 and \mathbf{Q}_2 , respectively, and \mathbf{U}_1 and \mathbf{U}_2 are the corresponding eigenvector matrices.

As is well known, the discrete FT (DFT) of an image can be expressed as the form of matrix multiplication

$$\hat{\mathbf{A}} = \mathbf{F}_1 \mathbf{A} \mathbf{F}_2, \quad \hat{\mathbf{B}} = \mathbf{F}_1 \mathbf{B} \mathbf{F}_2 \quad (12)$$

where $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$ are the DFT of \mathbf{A} and \mathbf{B} , and \mathbf{F}_1 and \mathbf{F}_2 are the corresponding DFT matrices. Interestingly, it can be found that the DFT matrix is exactly the eigenvector matrix of \mathbf{Q} matrix, which means $\mathbf{U}_1 = \mathbf{F}_1$ and $\mathbf{U}_2 = \mathbf{F}_2$. In addition, since the DFT matrix is a symmetric complex matrix, $\mathbf{F} = \mathbf{F}^T$ and $\mathbf{F}^* = \mathbf{F}^H$. Therefore, after some simple algebraic operations, the DFT of (11) can be calculated as

$$\mathbf{F}_1 (\mathbf{Q}_1^T)^x \mathbf{B} \mathbf{Q}_2^y \mathbf{F}_2 = \mathbf{D}_1^x \hat{\mathbf{B}} \mathbf{D}_2^y. \quad (13)$$

Since \mathbf{D}_1^x and \mathbf{D}_2^y are both diagonal matrices, the right-hand side of (13) can be rewritten as

$$\mathbf{D}_1^x \hat{\mathbf{B}} \mathbf{D}_2^y = \hat{\mathbf{B}} \times (\text{diag}(\mathbf{D}_1^x) \text{diag}(\mathbf{D}_2^y)^T) \quad (14)$$

where \times represents the elementwise product operator.

It is worth noting that the DFT matrix is necessarily a unitary matrix, and an $n \times n$ DFT matrix \mathbf{F} can be expressed as

$$\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_n] \quad (15)$$

where \mathbf{f}_p is the p th column of \mathbf{F} matrix, and the q th element of \mathbf{f}_p is $f_{qp} = (1)/(\sqrt{n})e^{i((q-1)*(p-1)*2\pi)/(n)}$. Combining (5) and (15), the eigenvalue matrix \mathbf{D} of the \mathbf{Q} matrix can be rewritten as

$$\mathbf{D} = \mathbf{F}^H \mathbf{Q} \mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_n]^H \mathbf{Q} [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_n]. \quad (16)$$

Since \mathbf{Q} matrix is the CSM, when the column vector \mathbf{f}_p is left-multiplied by the \mathbf{Q} matrix, it will cause a cyclic shift for the vector, which is equivalent to multiplying \mathbf{f}_j by a factor $e^{i((p-1)*2\pi)/(n)}$ (i.e., $\mathbf{Q}\mathbf{f}_p = e^{i((p-1)*2\pi)/(n)}\mathbf{f}_p$). Then, (16) can be re-expressed as

$$\mathbf{D} = [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_n]^H \left[\mathbf{f}_1, e^{i\frac{2\pi}{n}} \mathbf{f}_2, \dots, e^{i\frac{(n-1)*2\pi}{n}} \mathbf{f}_n \right]. \quad (17)$$

It can be easily verified that the diagonal elements of \mathbf{D} are $d_{pp} = e^{i((p-1)*2\pi)/(n)}$, which means in (14), $\text{diag}(\mathbf{D}_1^x) = [e^{iu_1x}, e^{iu_2x}, \dots, e^{iu_mx}]^T$ and $\text{diag}(\mathbf{D}_2^y) = [e^{iv_1y}, e^{iv_2y}, \dots, e^{iv_ny}]^T$, where $u_j = ((j-1)*2\pi)/(m)$ ($j = 1, \dots, m$) and $v_k = ((k-1)*2\pi)/(n)$ ($k = 1, \dots, n$)

are the phase angles of the j th and k th eigenvalue of \mathbf{Q}_1 and \mathbf{Q}_2 , respectively.

Let $\hat{\mathbf{C}} = \hat{\mathbf{A}} ./ \hat{\mathbf{B}}$, where $./$ stands for the elementwise division operator. Combining (12) and (14), (8) can be transformed into the following optimization problem:

$$\min_{x,y} f(x, y) = \min_{x,y} \|\hat{\mathbf{C}} - (\text{diag}(\mathbf{D}_1^x) \text{diag}(\mathbf{D}_2^y)^T)\|_F. \quad (18)$$

If two images \mathbf{A} and \mathbf{B} are circularly shifted versions of each other, then x and y must exist to make $\mathbf{A} = (\mathbf{Q}_1^T)^x \mathbf{B} \mathbf{Q}_2^y$, and thus, $\hat{\mathbf{C}} = \text{diag}(\mathbf{D}_1^x) \text{diag}(\mathbf{D}_2^y)^T$ holds. It can be found that the modules of all the elements in $\hat{\mathbf{C}}$ are 1 since the modules of all the elements of $\text{diag}(\mathbf{D}_1^x) \text{diag}(\mathbf{D}_2^y)^T$ are evidently equal to 1. Therefore, in order to obtain the optimal solution $(x$ and $y)$ of the model (8) or (18), we only need to pay attention to their phase information. Because

$$\text{diag}(\mathbf{D}_1^x) \text{diag}(\mathbf{D}_2^y)^T = \begin{bmatrix} e^{i(u_1 \cdot x + v_1 \cdot y)} & \dots & e^{i(u_1 \cdot x + v_n \cdot y)} \\ \vdots & \ddots & \vdots \\ e^{i(u_m \cdot x + v_1 \cdot y)} & \dots & e^{i(u_m \cdot x + v_n \cdot y)} \end{bmatrix} \quad (19)$$

the phase information of $(\text{diag}(\mathbf{D}_1^x) \text{diag}(\mathbf{D}_2^y)^T)$ can be extracted from (19)

$$\begin{aligned} & \text{Phase}(\text{diag}(\mathbf{D}_1^x) \text{diag}(\mathbf{D}_2^y)^T) \\ &= \begin{bmatrix} u_1 \cdot x + v_1 \cdot y & \dots & u_1 \cdot x + v_n \cdot y \\ \vdots & \ddots & \vdots \\ u_m \cdot x + v_1 \cdot y & \dots & u_m \cdot x + v_n \cdot y \end{bmatrix}. \end{aligned} \quad (20)$$

Let us define a third-order tensor $\mathcal{X} \in \mathbb{R}^{m \times n \times 2}$, and its frontal slices are given by

$$\begin{aligned} \mathcal{X}_{::1} &= \begin{bmatrix} u_1 & \dots & u_1 \\ \vdots & \ddots & \vdots \\ u_m & \dots & u_m \end{bmatrix} \\ \mathcal{X}_{::2} &= \begin{bmatrix} v_1 & \dots & v_n \\ \vdots & \ddots & \vdots \\ v_1 & \dots & v_n \end{bmatrix}. \end{aligned} \quad (21)$$

Denote the $\text{Phase}(\hat{\mathbf{C}})$ as the phase matrix of $\hat{\mathbf{C}}$, then the optimization problem in (18) can be further transformed into

$$\min_{x,y} f(x, y) = \min_{x,y} \|\text{Phase}(\hat{\mathbf{C}}) - \mathcal{X} \times_3 \mathbf{s}\|_F \quad (22)$$

where $\mathbf{s} = [x, y]^T$ are the parameters we need to estimate, and \times_3 represents the 3-mode product operator. The vector and matrix form of $\text{Phase}(\hat{\mathbf{C}})$ and \mathcal{X} are denoted, respectively, as

$$\begin{aligned} \mathbf{c} &= \text{vec}(\text{Phase}(\hat{\mathbf{C}})) \\ \mathbf{X} &= [\text{vec}(\mathcal{X}_{::1}), \text{vec}(\mathcal{X}_{::2})] \end{aligned} \quad (23)$$

where $\text{vec}(\cdot)$ is the vectorization operator. Theoretically, in the absence of noise, we could directly obtain the exact solution of (22) by using the least squares method, which is given by

$$\mathbf{s} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{c}. \quad (24)$$

It could be verified that the method presented above, named the CSM method, is equivalent to Stone's method [14] in

theory. And it should be noted that the above derivation can only be established when there is a cyclic shift relationship between the images to be matched, which is hardly satisfactory in real situations. For practical applications, the least squares solution (24) can only be considered as an approximate estimation for a real sub-pixel shift.

B. Applicable Condition of the Cross-Power Spectrum Method

In this section, we explore the applicable condition of the cross-power spectrum-based method and obtain an interesting finding which is stated in Lemma 1.

Lemma 1: Consider two $m \times n$ images (\mathbf{G} and \mathbf{H}) to be matched, and the traditional phase-based methods can obtain the exact solution for any sub-pixel shifts only when x_0 and y_0 exist to make the equation $\mathbf{G} = (\mathbf{Q}_1^H)^{x_0} \mathbf{H} \mathbf{Q}_2^{y_0}$ hold, where \mathbf{Q}_1 and \mathbf{Q}_2 are the CSMs defined by (4).

Proof: Denote $\hat{\mathbf{G}}$ and $\hat{\mathbf{H}}$ as the DFT of the images \mathbf{G} and \mathbf{H} , respectively, and their cross-power spectrum matrix is

$$\mathbf{P} = (\hat{\mathbf{G}} \times \hat{\mathbf{H}}^*) / |\hat{\mathbf{G}} \times \hat{\mathbf{H}}^*| = (\hat{\mathbf{G}} / \hat{\mathbf{H}}) / |\hat{\mathbf{G}} / \hat{\mathbf{H}}| \quad (25)$$

where $|\cdot|$ represents the module operation on each element of the matrix.

When there is cyclic shift relationship between the images \mathbf{G} and \mathbf{H} , according to the derivation in Section II-A, x_0 and y_0 must exist to make the following equation hold:

$$\hat{\mathbf{G}} / \hat{\mathbf{H}} = \text{diag}(\mathbf{D}_1^{x_0}) \text{diag}(\mathbf{D}_2^{y_0})^T \quad (26)$$

where \mathbf{D}_1 and \mathbf{D}_2 are the eigenvalue matrices of \mathbf{Q}_1 and \mathbf{Q}_2 , respectively. As discussed in Section II-A, since $\text{diag}(\mathbf{D}_1^x) = e^{i\mathbf{u}x_0}$ and $\text{diag}(\mathbf{D}_2^{y_0}) = e^{i\mathbf{v}y_0}$, where $\mathbf{u} = (u_1, u_2, \dots, u_m)^T$, $\mathbf{v} = (v_1, v_2, \dots, v_n)^T$, and $u_j (j = 1, \dots, m)$ and $v_k (k = 1, \dots, n)$ are defined as above, (26) can be expressed as

$$\hat{\mathbf{G}} / \hat{\mathbf{H}} = e^{i\mathbf{u}x_0} e^{i\mathbf{v}^T y_0}. \quad (27)$$

In this case, the module of each element in $\hat{\mathbf{G}} / \hat{\mathbf{H}}$ is equal to 1. Therefore, we can rewrite (27) as

$$(\hat{\mathbf{G}} / \hat{\mathbf{H}}) / |\hat{\mathbf{G}} / \hat{\mathbf{H}}| = e^{i\mathbf{u}x_0} e^{i\mathbf{v}^T y_0} \quad (28)$$

which is exactly the discrete matrix form of (3).

The traditional phase-based methods are all based on (1), which is only established on the assumption that the size of an image is infinite. However, real images are discrete and finite, so it is impossible for them to meet this condition. Interestingly, (28) indicates that (2) or (3) still holds if there is a cyclic shift relationship between the images to be matched, and the exact solution can be obtained by the least squares method in the absence of noise. Therefore, we can conclude that the cross-power spectrum-based methods or the phase-based methods can only obtain an exact solution for the sub-pixel TMP when a circularly shifted relationship exists between the images to be matched. ■

For real images, when the cyclic shift relationship between images is not satisfied, (28) does not hold. This is why none of the phase-based methods can obtain an exact solution for the TMP.

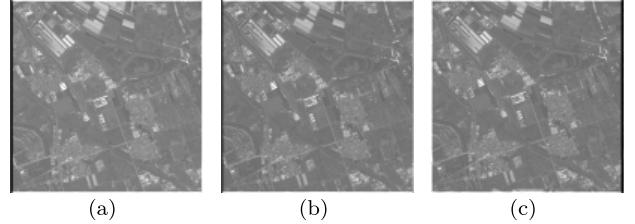


Fig. 3. Reference and matching images. (a) Reference image. (b) Matching image with $x = 0.5$ and $y = 0.5$. (c) Matching image with $x = 2.5$ and $y = 2.5$.

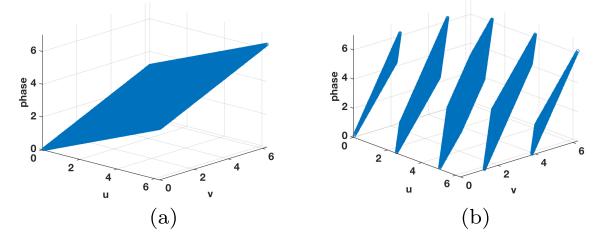


Fig. 4. Phase information of different shifts. (a) Phase information with $x = 0.5$ and $y = 0.5$. (b) Phase information with $x = 2.5$ and $y = 2.5$.

C. Implementation of CSM

Theoretically, our method works for any sub-pixel translation of two images to be matched. However, as mentioned above, the phase wrapping phenomenon appears when the sub-pixel shift is large [14], [16]. For example, an image [Fig. 3(a)] of 129×129 size is used as the reference to generate two circularly shifted images with shifts $(0.5, 0.5)$ [Fig. 3(b)] and $(2.5, 2.5)$ [Fig. 3(c)]. The phase information, which is defined as (20), of the generated images is shown in Fig. 4.

As can be seen, the phase ranges smoothly from 0 to 2π when $x = 0.5$ and $y = 0.5$, and there is no phase wrapping phenomenon in Fig. 4(a). In contrast, in the case of $x = 2.5$ and $y = 2.5$, the phase angle splits into five parts and the phase is heavily wrapped, as shown in Fig. 4(b).

As proved in Section II-A, the values of u_j and v_k both range from 0 to 2π , and thus, in theory, the length of the phase range of (20) is not greater than $2\pi(|x| + |y|)$. However, no matter how large the value of $(|x| + |y|)$, the phase of (20) will always locate between 0 and 2π in the actual calculation. Therefore, the phase wrapping phenomenon is unavoidable when $|x| + |y| > 1$, as shown in Fig. 4(b).

One solution to the above problem is phase unwrapping. However, it is hard for phase unwrapping to obtain high precision results when noise and other interference exist in the images. An alternative way is to make the theoretical phase and the actual phase both locate in $[0, 2\pi]$, which implies $|x| + |y| \leq 1$. As a result, we roughly divide our method into two stages: the integer pixel shift calculation (IPSC) and the sub-pixel shift calculation (SPSC). The detailed steps are as follows.

- 1) *Integer Pixel Calculation:* Because the inverse DFT (IDFT) has proved effective in dealing with the problem of integer pixel translation, at the IPSC stage, we directly use the IDFT of the cross-power spectrum matrix to make a coarse registration for images to locate the integer part of (x, y) [2]. Then, the registration images will

be resized according to the result of the transformation. For example, if the integer part of (x, y) is $(6, 6)$ and the size of the image pair is 129×129 , then we crop the images to size 123×123 .

- 2) *Frequency Masking and Noise Removing*: Since high-frequency components are more susceptible to noise [23] and most of the energy is concentrated in the low-frequency regions for natural images [24], we should mask out the high-frequency component before the estimation. In this paper, we preprocess the phase information [Phase(\hat{C})] with an ideal low-pass filter.
- 3) *Initial Registration*: After eliminating the high-frequency component, we directly use (24) to obtain the initial sub-pixel shift estimate $(x_s^{(0)}, y_s^{(0)})$.
- 4) *Fine Registration*: Due to the existence of noise and other interference, even after removing the integer part of the shifts and the high-frequency component, the phase wrapping phenomenon may still exist. In this situation, the performance of CSM or other phase-based methods is still not guaranteed. To address this issue, we iteratively apply our CSM method to the image pair to gradually reduce the influence of noise and interference. Details are as follows.
 - a) A new matching image $\mathbf{B}^{(1)}$ is generated according to the initial estimate $(x_s^{(0)}, y_s^{(0)})$: $\mathbf{B}^{(1)} = (\mathbf{Q}_1^T)^{x_s^{(0)}} \mathbf{B}^{(0)} \mathbf{Q}_2^{y_s^{(0)}}$. Then, in order to eliminate the impact of unmatched regions, the outermost pixel of the image $\mathbf{B}^{(1)}$ is removed, and the same operation is performed on the reference image \mathbf{A} .
 - b) We apply the CSM method to the resized reference image \mathbf{A} and the generated matching image $\mathbf{B}^{(1)}$ to obtain the revised estimation $(x_s^{(1)}, y_s^{(1)})$. Then, we generate the new matching image $\mathbf{B}^{(2)} = (\mathbf{Q}_1^T)^{x_s^{(1)}} \mathbf{B}^{(0)} \mathbf{Q}_2^{y_s^{(1)}}$ and resize it in the same manner.

Repeating the above steps, we can gradually correct the sub-pixel shift estimation. It should be noted that because the subsequent images are generated by the original matching image $\mathbf{B}^{(0)}$, their sizes will not get smaller with an increase in the number of iterations. The combination of the integer and sub-pixel shift is the final translation estimation.

The key to the above steps is the repeated use of the CSM. In this way, the shift between the images is continuously reduced. Thus, the phase wrapping problem can be gradually alleviated, and the matching accuracy is expected to be increasingly high. The pseudo-code is given in Algorithm 1.

III. EXPERIMENTS AND RESULTS

Simulated and real images were used to evaluate the performance of the CSM method. In simulated cases, we conducted experiments with two types of displacement images: cyclic shift images and images with known offsets. In order to evaluate the accuracy of our algorithm, we compared the CSM method with five classical phase correlation methods: the traditional DFT [7] method, the DFT with up-sampling (DFT-US) [13] method, the traditional SVD method [17], the SVD-Ransac method [20] and Stone's method [14].

Algorithm 1 Pseudo-Code to Implement the Iterative CSM Method

Input: The reference image \mathbf{A} , the matching image \mathbf{B} , the number of iterations k .

Output: Estimation of the pixel shifts (x, y)

- 1: **Integer pixel shift calculation:**
- 2: Calculating the integer shift $(x_{\text{integer}}, y_{\text{integer}})$ by the DFT method in [7], and obtaining the new matching image pair $\mathbf{A}^{(0)}$ and $\mathbf{B}^{(0)}$ by resizing the images (\mathbf{A} and \mathbf{B}) according to $(x_{\text{integer}}, y_{\text{integer}})$.
- 3: **Sub-pixel calculation:**
- 4: Calculating the Phase(diag(\mathbf{D}_1^x)diag(\mathbf{D}_2^y) T) and Phase(\hat{C}) according to Section II-A.
- 5: Applying an ideal low-pass filter to the Phase(diag(\mathbf{D}_1^x)diag(\mathbf{D}_2^y) T) and Phase(\hat{C}) to reduce the influence of noise and interference.
- 6: Calculating the sub-pixel estimate $(x_s^{(0)}, y_s^{(0)})$ according to (24).
- 7: **for** $j = 1 \rightarrow k - 1$ **do**
- 8: Generating the new matching image $\mathbf{B}^{(j)}$ by $\mathbf{B}^{(j)} = (\mathbf{Q}_1^T)^{x_s^{(j-1)}} \mathbf{B}^{(0)} \mathbf{Q}_2^{y_s^{(j-1)}}$.
- 9: Repeating steps 4, 5 and 6 with images $\mathbf{A}^{(0)}$ and $\mathbf{B}^{(j)}$, and obtaining the new sub-pixel estimation $(x_s^{(j)}, y_s^{(j)})$.
- 10: $(x_s^{(j)}, y_s^{(j)}) = (x_s^{(j)}, y_s^{(j)}) + (x_s^{(j-1)}, y_s^{(j-1)})$.
- 11: **end for**
- 12: $(x, y) = (x_{\text{integer}}, y_{\text{integer}}) + (x_s^{(k)}, y_s^{(k)})$.

The traditional DFT method determines the pixel shifts by directly using the peak location of the IDFT result of the cross-power spectrum matrix \mathbf{P} . The DFT-US method raises the accuracy up to $1/k$ pixel in theory by up-sampling the data with a factor of k , which is set to 10 in the following experiments. The traditional SVD method uses the phase information of the dominant singular vector of the cross-power spectrum matrix \mathbf{P} to estimate the shifts, and the SVD-Ransac method adds the phase fringe filter [15] and the Ransac algorithm to reduce the influence of noise and interference. A frequency masking procedure is required in the SVD method, the SVD-Ransac method, Stone's method, and the CSM method. In the following experiments, the diameter of the masking is set to half the length of the short edge of the matching images for these methods.

The estimation error e is defined as the 2-norm of the offset between the estimation and the ground truth, which is given by $e = (e_x^2 + e_y^2)^{1/2}$, where e_x and e_y are the offsets in the x - and y -directions, respectively. In order to obtain a result in the statistical sense, we conducted multiple experiments with different parameters for each method, and three metrics were used to evaluate their performance: the mean value of e , the max value of e , and the standard deviation σ_r of e .

As for the real image case, we chose a hyperspectral image, which was acquired using a push-broom imaging spectrometer, to evaluate the performance of the CSM method. The push-broom imaging spectrometer has a misalignment between the cameras of the sensors, so there may exist sub-pixel offsets

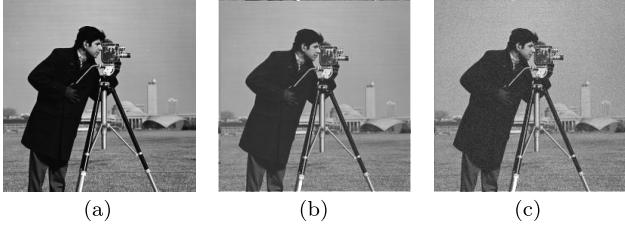


Fig. 5. Image used in the cyclic shift experiment. (a) Original image. (b) Example of circularly shifted image generated by \mathbf{Q} matrix with sub-pixel (2.5, 2.5). (c) Example of noise-adding circularly shifted image generated by \mathbf{Q} matrix with sub-pixel shift (2.5, 2.5) and $\sigma_g = 0.03$.

TABLE I
RESULTS OF CSM WITH $k = 1$ FOR THE CYCLIC SHIFT EXPERIMENT

pixel shifts (x_t, y_t)	estimation (x_t, y_t)	2-norm error
(0.1, 0.1)	(0.1, 0.1)	0
(0.7, 0.7)	(0.7, 0.7)	0
(1.3, 1.3)	(1.3, 1.3)	0
(1.9, 1.9)	(1.9, 1.9)	0
(2.5, 2.5)	(2.5, 2.5)	0

between different bands in an image, which will lead to spectral distortion. Therefore, we applied our algorithm to different bands of a hyperspectral image to estimate (and calibrate) the translation offset between them. Since there is no ground truth of the offset for the used data, the CSM algorithm will not be compared with other algorithms and the performance is evaluated only by visual inspection.

A. Cyclic Shift Images

In the cyclic shift experiment, we directly generated the simulated data using the cyclic shift matrix \mathbf{Q} , and the original image \mathbf{A} used in the experiment is shown in Fig. 5(a). The circularly shifted images are generated according to $\mathbf{B}_i = (\mathbf{Q}^T)^{x_i} \mathbf{A} \mathbf{Q}^{y_i}$, where x_i and y_i are pixel shifts in the x - and y -directions, respectively. In this section, the pixel shifts (x_i, y_i) range from 0.1 to 2.5 in steps of 0.3 pixel for both the x - and y -directions. Thus, we can generate 81 shifted images from the reference. One of the generated images with (2.5, 2.5) pixel shifts is illustrated in Fig. 5(b), and the matching results of the CSM method with the number of iterations $k = 1$ are listed in Table I.

As can be seen, no matter how large we set the pixel shifts, the CSM method could always obtain the exact solution, which means the CSM method can completely deal with the situation of cyclic shift. However, for a real image, there always exists some noise, so we conduct the following experiment, where Gaussian white noise is added to the reference and matching images.

The images are first normalized within $[0, 1]$. Then, the zero-mean Gaussian white noise with the standard deviation $\sigma = 0.03$ is added to each of them, and the corrupted image corresponding to Fig. 5(b) is shown in Fig. 5(c). The number of iterations for CSM is set to $k = 1, 3$, and 5. The results for different methods are tabulated in Table II. It can be seen that in the presence of noise, none of these methods can obtain an exact solution. Among them, the CSM ($k = 3$ and 5)

TABLE II
RESULTS OF DIFFERENT METHODS FOR THE CYCLIC SHIFT IMAGES WITH NOISE

Method	Mean error	Max error	Standard deviation of error
SVD	0.9978	1.7585	0.3833
SVD-Ransac	0.0205	0.0549	0.0120
DFT	0.3960	0.7071	0.1531
DFT-Upsampling	0.0836	0.1212	0.0214
Stone	0.0282	0.0756	0.0131
CSM($k = 1$)	0.0282	0.0756	0.0131
CSM($k = 3$)	0.0095	0.0229	0.0048
CSM($k = 5$)	0.0094	0.0221	0.0047

method achieves the best results in all three metrics, while the performance of the SVD method is the worst, which implies that the SVD method is susceptible to noise. The result of Stone's method is exactly the same as that of CSM ($k = 1$), since they are both based on the least squares method. It is worth noting that the performance of CSM is improved with the increasing number of iterations, which indicates that the iteration strategy presented in the step of fine registration is indeed effective.

B. Images With Known Offsets

In this section, we use several real image pairs with known offsets to evaluate the performance of CSM. The original images used in this section are high-resolution images from the Gaofen-2 (GF-2) satellite with a $4\text{ m} \times 4\text{ m}$ resolution and a size of 6900×7300 pixels. In order to evaluate the algorithms quantitatively, we use the following method to obtain the ground truth of the sub-pixel shifts: we first crop two sub-images to size 1290×1290 from the original image at intervals of s pixels. Then, we down-sampled the image pair by a factor of k . As a result, the shift of the new image pair becomes s/k pixels. The value of k is set to 10 in all experiments.

In addition, two kinds of down-sampling methods are used in this paper: the mean down-sampling (MDS) method and the directly down-sampling (DDS) method. The difference between them is that MDS uses the average gray value of a region, for example, the average gray value of a region size 10×10 , to represent the region, while DDS picks out the pixels in the top-left corner to represent the region.

In the following, we evaluate the effect of two dominant error sources (aliasing and noise) on the sub-pixel phase correlation methods. Experiments are carried out with a variety of noise and aliasing, which are named the aliasing experiment and noise experiment, respectively.

1) *Aliasing Experiment:* Three sub-images (1290×1290 pixels) of the GF-2 satellite imagery, as shown in Fig. 6(a)–(c), are selected as the original reference images. For each of them, 50 matching images (1290×1290 pixels) are cropped from the original images (6900×7300 pixels) with the integer offsets ranging from 1 to 50 pixels by a step of 1 pixel in both the x - and y -directions. To simulate the effect of aliasing, we first apply a 2-D Gaussian function with a support size of g_s , which is set to 25×25 , and a standard deviation of σ as the blurring kernel to the

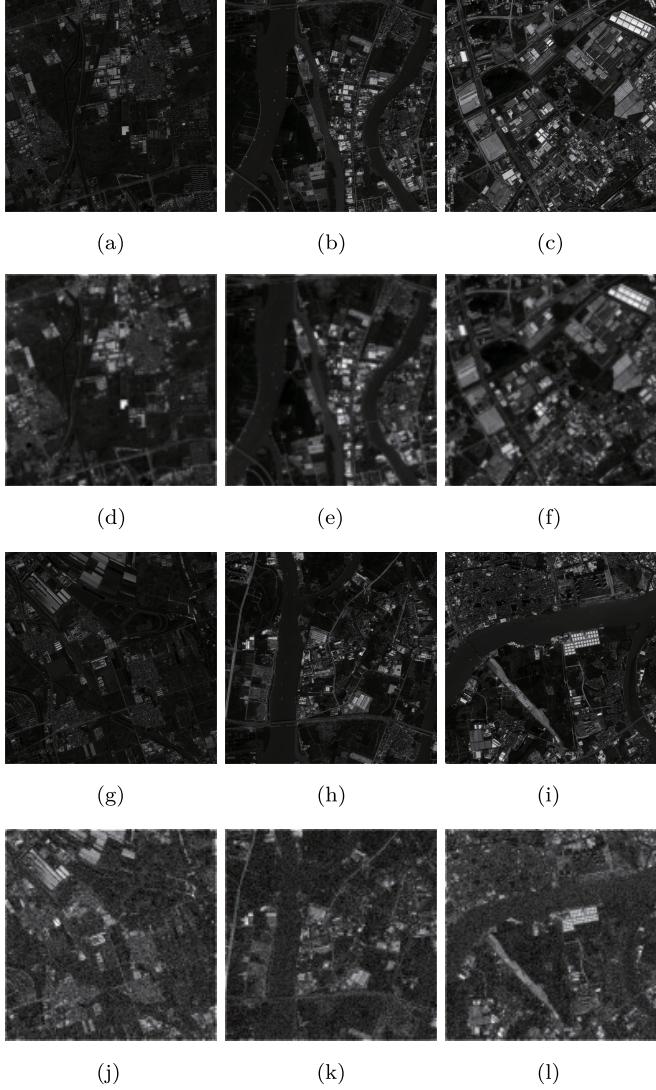


Fig. 6. Original and generated images used in experiments. (d)–(f) Value of σ is set to 2. (j)–(l) Value of σ_g is set to 0.03. (a) Sub-image 1 used in the aliasing experiment (1290×1290). (b) Sub-image 2 used in the aliasing experiment (1290×1290). (c) Sub-image 3 used in the aliasing experiment (1290×1290). (d) Down-sampled image 1 with aliasing (129×129). (e) Down-sampled image 2 with aliasing (129×129). (f) Down-sampled image 3 with aliasing (129×129). (g) Sub-image 1 used in the noise experiment (1290×1290). (h) Sub-image 2 used in the noise experiment (1290×1290). (i) Sub-image 3 used in the noise experiment (1290×1290). (j) Down-sampled image 1 with noise (129×129). (k) Down-sampled image 2 with noise (129×129). (l) Down-sampled image 3 with noise (129×129).

original reference and matching images. Then, 150 image pairs (129×129 pixels) used in the experiment can be generated by each of the two down-sampling methods described above. By controlling the value of σ , we could change the amount of aliasing. In the experiment, the value of σ ranges from 0 to 5 in steps of 0.5 and the generated data where $\sigma = 2$ are shown in Fig. 6(d)–(f). Also, the number of iterations k of the CSM method is set to $k = 1, 3$, and 5, where $k = 1$ means that there is no iteration. We use the results of all the 150 image pairs for error statistics, and the error curves with respect to the variance (σ) are plotted in Fig. 7. Since the DFT method can only solve the translation problem of integer pixels, its result will not be included in the figure.

As can be seen, in the case of MDS, the curves of the three metrics (average error, maximum error, and standard derivation) are relatively flat. It means that the performances of these methods are all insensitive to the degree of aliasing when MDS is adopted. However, their performance can be greatly improved as the value of σ increases in the DDS situation.

The accuracy of the DFT method can be improved by introducing an up-sampling operation, and on the whole, the curves of DFT-US show a relatively flat trend. Similarly, after the introduction of the Ransac algorithm, the SVD-Ransac method has a better performance than SVD, as shown in Fig. 7. As expected, the performance of Stone's method and CSM ($k = 1$) are exactly the same.

The CSM method with $k = 3$ and 5 outperforms all the other methods for all the metrics, which indicates that the iteration strategy in the CSM algorithm is effective in real image cases. It is worth noting that the results of CSM ($k = 3$) are almost the same as that of CSM ($k = 5$), which means that CSM can obtain a good performance without too many iterations.

2) *Noise Experiment*: In this section, we select three different sub-images (1290×1290) from the GF-2 satellite imagery, which are shown in Fig. 6(g)–(i), as the original reference images and generate the image pairs (129×129) in the same way as Section III-B1. After normalizing the image value into $[0, 1]$, a zero-mean Gaussian white noise with a standard variance of σ_g is added into all the image pairs to analyze the influence of noise on the performance of the methods. In this experiment, the value of σ_g ranges from 0 to 0.05 in steps of 0.005. The images with $\sigma_g = 0.03$ are shown in Fig. 6(j)–(l). The number of iterations, k , for CSM is also set to $k = 1, 3, 5$. The results of the 150 image pairs are used for statistics, and the error curves of the different methods with respect to σ_g are shown in Fig. 8.

As can be seen, when $\sigma_g = 0$, which means there is no noise in the images to be matched, none of the methods can obtain an exact solution. Among them, the CSM method achieves the best results in all three metrics for both the MDS and DDS situations. In addition, the performance of CSM with $k = 3$ and 5 is obviously better than others, which again indicates the effectiveness of the iteration strategy.

Different from aliasing, the influence of noise is significant on the sub-pixel shift estimation. With an increase of σ_g , all the methods tend to deteriorate, and the trend is more obvious in the DDS situation. As can be seen from Fig. 8, although the precision of DFT-US is not very high, it has a strong anti-noise performance. The SVD method and SVD-Ransac perform well when σ_g is small. However, they are susceptible to noise and their results significantly become worse as σ_g grows. Stone's method shows the same performance as that of CSM ($k = 1$), which is as expected.

Of all the methods, CSM surpasses all the others, especially when $k = 3$ and 5. The performance of CSM with $k = 3$ and 5 is almost the same in the MDS situation. However, for DDS, the difference between the methods gradually becomes apparent with the increase of σ_g . The possible reason is that the DDS mode cannot reduce the influence of noise, while the average down-sampling operation in MDS can improve the signal-to-noise ratio of data. Therefore, when the image

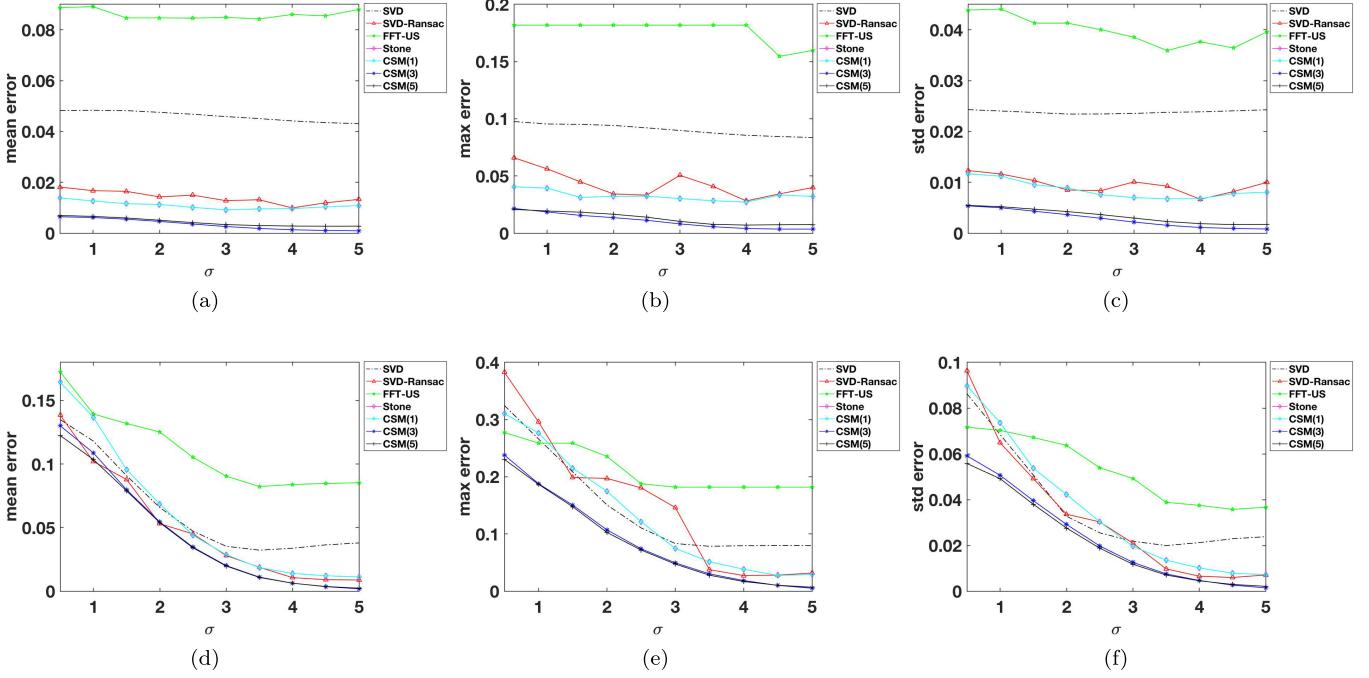


Fig. 7. Results of the aliasing experiment. The legend “CSM(k)” indicates the number of iterations of the CSM algorithm. The results of the DFT method are not plotted for a better visualization. (a) Mean error in the MDS case. (b) Max error in the MDS case. (c) Standard derivation in the MDS case. (d) Mean error in the DDS case. (e) Max error in the DDS case. (f) Standard derivation in the DDS case.

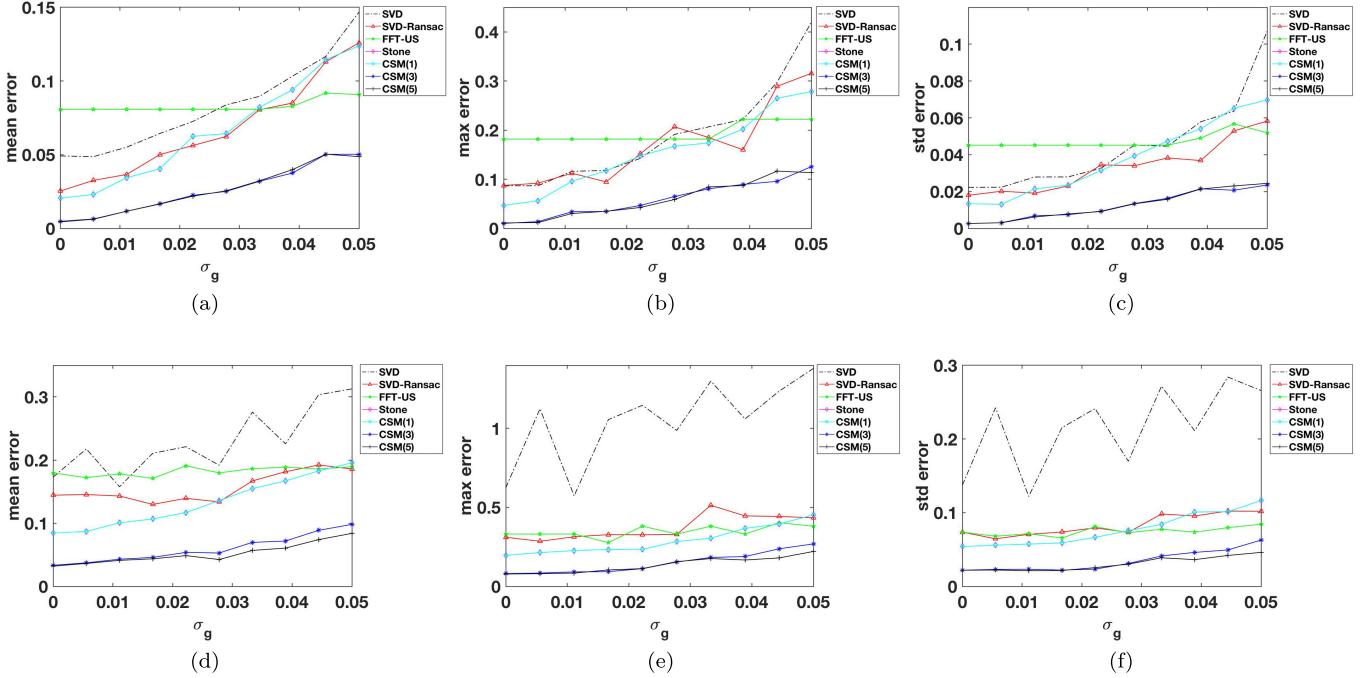


Fig. 8. Results of the noise experiment. The legend “CSM(k)” indicates the number of iterations of the CSM algorithm. The results of the DFT method are not plotted for a better visualization. (a) Mean error in the MDS case. (b) Max error in the MDS case. (c) Standard derivation in the MDS case. (d) Mean error in the DDS case. (e) Max error in the DDS case. (f) Standard derivation in the DDS case.

is heavily corrupted by noise, more iterations in CSM are required in the case of DDS to obtain a better result.

In addition, the computational time of all these methods is demonstrated in Table III. All the algorithms mentioned above are programmed in MATLAB on a computer with 3.0-GHz CPU and 8-GB memory. From the table, we can

see that: 1) the computational time required for all methods is relatively short, and the computational complexity of DFT is the lowest due to its simplicity and 2) the time cost of CSM with $k = 1$ is comparable with other methods. And, the computational time of CSM(k) grows linearly with the number of iterations k .

TABLE III
COMPUTATIONAL TIME OF DIFFERENT METHODS

Method	Total time (s)
SVD	6.37e-3
SVD-Ransac	7.41e-3
DFT	2.35e-3
DFT-Upsampling	5.18e-3
Stone	5.14e-3
CSM($k = 1$)	6.94e-3
CSM($k = 3$)	0.020
CSM($k = 5$)	0.031



Fig. 9. False color image of farmland data (R: 596.70 nm, G: 545.90 nm, and B: 451.10 nm).

C. Real Hyperspectral Image

As mentioned above, there may exist sub-pixel offsets between different bands in an image which is acquired by the push-broom imaging spectrometer, so it is necessary to estimate and calibrate the offset to prevent spectral distortion. In this section, the hyperspectral image acquired by push-broom hyperspectral imaging (PHI) in Japan is used to evaluate the performance of the CSM algorithm.

The PHI spectrometer was developed by the Shanghai Institute of Technical Physics, Chinese Academy of Sciences. The data are composed of 80 bands with the wavelength ranging from 411.9 to 832.79 nm. The scene in the image is covered by a variety of farmland and is thus referred to as farmland data. The original size of the farmland data is 350×570 pixels and we crop a 129×129 pixels sub-image to conduct the experiment. The false color image is shown in Fig. 9.

The 4th and 35th bands of the farmland data are shown in Fig. 10. As can be seen from the corresponding local magnification images, compared to band 4, band 35 is slightly higher in the vertical direction, and there is no obvious offset in the horizontal direction.

The experimental process can be described as follows: we first choose a certain band of the data as the reference image **A** and choose another band as the image **B** to be matched. Then, CSM ($k = 5$) is applied to these two images and we can obtain the pixel offsets (x_s, y_s) in both the x - and y -directions. Finally, we generate the calibrated image $\tilde{\mathbf{B}}$ by cyclically shifting the image **B** according to (x_s, y_s) .

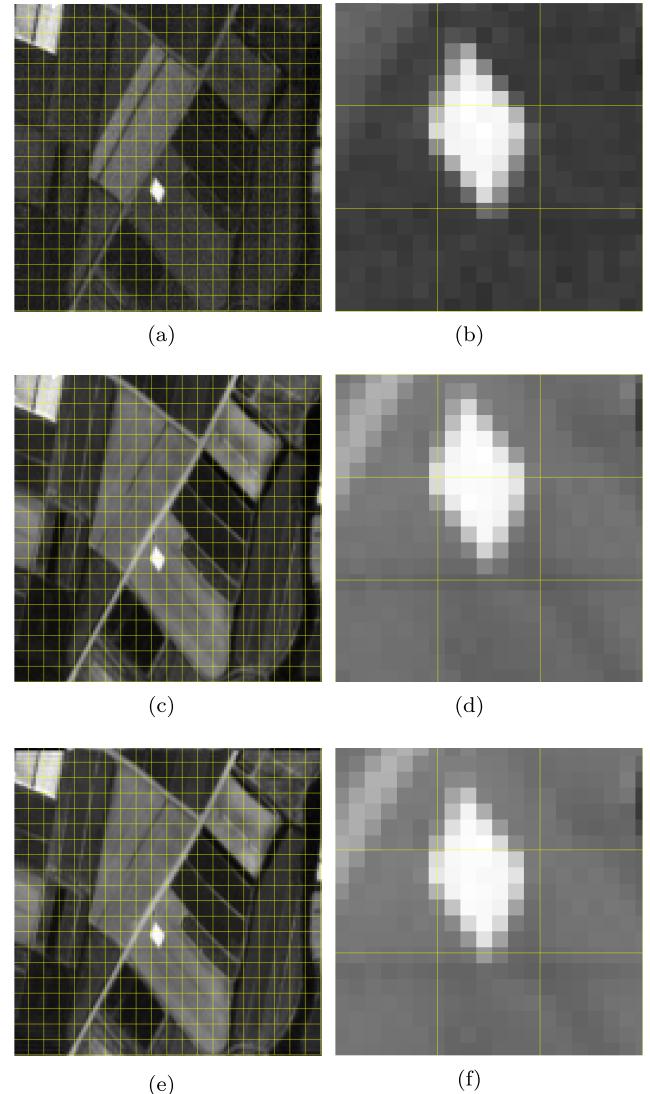


Fig. 10. Translation matching experiment between different bands for the PHI farmland data. (a) Reference image A (band 4). (b) Local magnification of A. (c) Image B to be matched (band 35). (d) Local magnification of B. (e) Calibrated image $\tilde{\mathbf{B}}$ (band 35). (f) Local magnification of $\tilde{\mathbf{B}}$

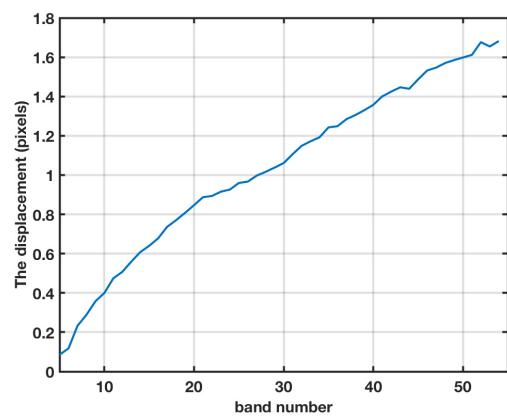


Fig. 11. Displacement between different bands for the PHI farmland data. Band 4 is chosen as the reference band.

When selecting band 4 as the reference image **A** and band 35 as the image **B**, the offset obtained by the CSM method is $(0.07, 1.24)$, which is roughly consistent with visual

inspection. The calibrated image $\tilde{\mathbf{B}}$ is shown in Fig. 10(e), and it can be seen that image $\tilde{\mathbf{B}}$ and the reference image \mathbf{A} are well aligned in both the horizontal and vertical directions.

Furthermore, we use band 4 as the reference image \mathbf{A} and calculate a series of offsets from band 5 to band 54. The result is shown in Fig. 11. It can be seen that the displacement between the image pairs gradually increases with the band number. Therefore, in order to ensure the correctness of the spectra, it is necessary to carry out the translation calibration between different bands, and the presented CSM method will be a good choice.

IV. CONCLUSION

In this paper, by introducing the cyclic shift matrix, we propose the CSM method to address the sub-pixel TMP. Then, we give the necessary conditions on which the traditional phase-based translation matching algorithms can achieve an exact solution, that is, the images to be matched must have a cyclic shift relationship. In addition, with the introduction of the CSM, it is convenient for us to carry out the cyclic shift of arbitrary sub-pixels, which motivates us to propose an iterative version of CSM to further improve the matching accuracy. Experiments with simulated and real data demonstrate the superiority of the iterative CSM method. In addition, besides the TMP, CSM could also be used to deal with the rotation matching problem. We believe that the CSM will play an important role in the image matching field.

ACKNOWLEDGMENT

The author (X. Geng) would like to thank his most respected uncle Dekun Geng (1949–2019) for his long-term love and encouragement. His uncle will live forever in his heart.

REFERENCES

- [1] T. Heid and A. Käab, “Evaluation of existing image matching methods for deriving glacier surface displacements globally from optical satellite imagery,” *Remote Sens. Environ.*, vol. 118, pp. 339–355, Mar. 2012.
- [2] H. Foroosh, J. B. Zerubia, and M. Berthod, “Extension of phase correlation to subpixel registration,” *IEEE Trans. Image Process.*, vol. 11, no. 3, pp. 188–200, Mar. 2002.
- [3] J. G. Liu and H. Yan, “Phase correlation pixel-to-pixel image co-registration based on optical flow and median shift propagation,” *Int. J. Remote Sens.*, vol. 29, no. 20, pp. 5943–5956, 2008.
- [4] Y.-H. Jiang, G. Zhang, X.-M. Tang, D. Li, W.-C. Huang, and H.-B. Pan, “Geometric calibration and accuracy assessment of ZiYuan-3 multispectral images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 7, pp. 4161–4172, Jul. 2014.
- [5] S. Leprinse, S. Barbot, F. Ayoub, and J.-P. Avouac, “Automatic and precise orthorectification, coregistration, and subpixel correlation of satellite images, application to ground deformation measurements,” *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1529–1558, Jun. 2007.
- [6] S. C. Park, M. K. Park, and M. G. Kang, “Super-resolution image reconstruction: A technical overview,” *IEEE Signal Process. Mag.*, vol. 20, no. 3, pp. 21–36, May 2003.
- [7] C. Kuglin, D. A. Hines, C. Kuglin, C. D. Kuglin, D. C. Hines, and D. Hines, “The phase correlation image alignment method,” in *Proc. Int. Conf. Cybern. Soc.*, 1975, pp. 163–165.
- [8] I. E. Abdou, “Practical approach to the registration of multiple frames of video images,” *Proc. SPIE*, vol. 3653, pp. 371–382, Dec. 1998.
- [9] V. Argyriou and T. Vlachos, “A study of sub-pixel motion estimation using phase correlation,” in *Proc. Brit. Mach. Vis. Conf.*, Edinburgh, U.K., Sep. 2006, pp. 387–396.
- [10] J. Xie, F. Mo, C. Yang, P. Li, and S. Tian, “A novel sub-pixel matching algorithm based on phase correlation using peak calculation,” *Int. Arch. Photogram. Remote Sens.*, vol. XLI-B1, pp. 253–257, Jul. 2016.
- [11] J. Ren, J. Jiang, and T. Vlachos, “High-accuracy sub-pixel motion estimation from noisy images in Fourier domain,” *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1379–1384, May 2010.
- [12] Q. Tian and M. N. Huhns, “Algorithms for subpixel registration,” *Comput. Vis. Graph. Image Process.*, vol. 35, no. 2, pp. 220–233, 1986.
- [13] M. Guizar-Sicairos, S. T. Thurman, and J. R. Fienup, “Efficient subpixel image registration algorithms,” *Opt. Lett.*, vol. 33, no. 2, pp. 156–158, 2008.
- [14] H. S. Stone, M. T. Orchard, E.-C. Chang, and S. A. Martucci, “A fast direct Fourier-based algorithm for subpixel registration of images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 10, pp. 2235–2243, Oct. 2001.
- [15] J. G. Liu and H. Yan, “Robust phase correlation methods for sub-pixel feature matching,” in *Proc. 1st Conf. Syst. Eng. Auton. Syst.*, 2006, p. A13.
- [16] H. Foroosh and M. Balci, “Sub-pixel registration and estimation of local shifts directly in the Fourier domain,” in *Proc. Int. Conf. Image Process.*, vol. 3, Oct. 2004, pp. 1915–1918.
- [17] W. S. Hoge, “A subspace identification extension to the phase correlation method,” *IEEE Trans. Med. Imag.*, vol. 22, no. 2, pp. 277–280, Feb. 2003.
- [18] Y. Keller and A. Averbuch, “A projection-based extension to phase correlation image alignment,” *Signal Process.*, vol. 87, no. 1, pp. 124–133, 2007.
- [19] O. R. Chum and J. Matas, “Optimal randomized RANSAC,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1472–1482, Aug. 2008.
- [20] X. Tong *et al.*, “A novel subpixel phase correlation method using singular value decomposition and unified random sample consensus,” *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4143–4156, Aug. 2015.
- [21] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm, “USAC: A universal framework for random sample consensus,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 2022–2038, Aug. 2013.
- [22] H. Wang and D. Suter, “MDPE: A very robust estimator for model fitting and range image segmentation,” *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 139–166, 2004.
- [23] P. Vandewalle, S. Süsstrunk, and M. Vetterli, “A frequency domain approach to registration of aliased images with application to super-resolution,” *EURASIP J. Adv. Signal Process.*, vol. 2006, no. 1, 2006, Art. no. 071459.
- [24] A. Oliva and A. Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.



Xiurui Geng received the Ph.D. degree in hyperspectral remote sensing from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, China, in 2005.

He is currently a Professor with the Key Laboratory of Technology in Geo-spatial Information Process and Application Systems, Institute of Electronics, Chinese Academy of Sciences. His research interests include pattern recognition, machine learning, and matrix theory.



Weitun Yang received the B.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2014, and the M.S. degree in signal processing from the Institute of Electronics, Chinese Academy of Sciences, Beijing, in 2017, where he is currently pursuing the Ph.D. degree with the Key Laboratory of Technology in Geo-spatial Information Process and Application Systems.

His research interests include hyperspectral target detection, image registration, and machine learning.