Three-dimensional state space realization algorithm: noise suppression of fluorescence microscopy images and point spread functions

Xuming Lai[1,3], E. Sally Ward[1], Zhiping Lin[2] and Raimund J. Ober[1,3]

[1]Center for Immunology NB9.106, University of Texas Southwestern Medical Center at Dallas, 6000 Harry Hines Boulevard, Dallas, TX 75235-8576, USA.
[2]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore.
[3]Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75083-0688, USA.

ABSTRACT

A recently developed algorithm is applied to calculate a state space realization of a 3D microscopy image set. It is based on interpreting the image set as the impulse response of a 3D separable system. As an application it is shown how this algorithm, combined with approximation steps, can be used to suppress noise in 3D experimental point spread functions. The approach was motivated by a well known problem that a noisy point spread function degrades the results of deconvolution algorithms for the restoration of 3D fluorescence microscopy image sets. The proposed approach can also be applied to 3D fluorescence microscopy image sets of cells.

Keywords: Point spread function, estimation, state space algorithm, noise suppression

1. INTRODUCTION

The point spread function is of central importance in any deconvolution algorithm. In blind algorithms the point spread function is estimated as part of the algorithm whereas in non-blind algorithms the point spread function is assumed to be known. In non-blind algorithms either an experimentally acquired point spread function or a theoretically calculated one is used. It is well known that a theoretical point spread function is often an imprecise model for the actual point spread function.[1,2] Since it is known that an inaccurate point spread function can affect the performance of deconvolution algorithms[3] the experimental point spread function would appear to be the preferred choice of a point spread function model for a deconvolution algorithm. However, the significant noise components in experimental point spread functions are serious obstacles to their use in deconvolution algorithms, since it is known that a noisy point spread function can compromise the results of deconvolution algorithms.

It therefore appears that a promising approach is to suppress noise or random components in an experimental point spread function and to use a smoothed point spread function in deconvolution. As a standard method, a Gaussian filter is often used to smooth 3D image sets, including point spread functions.[4,5] Since the approach is based on a weighted average of neighboring pixels it typically results in the loss of sharp details in the point spread functions. Another approach to suppress noise is to average the acquired data from 20-30 beads and then to average the resulting point spread function cylindrically about the optical axis.[6,7] However, this approach removes any potential asymmetry in the point spread function.

Many advanced signal processing techniques call for the use of state space models. For example, state space realizations are often used for multi-dimensional filter design (see[8–11] and references therein). Therefore the question arises whether it is possible to use state space techniques to suppress noise in experimental point spread functions. Here we will show that it is indeed possible to calculate the state space representation of a 3D experimental point spread function, which is treated as the impulse response of a 3D separable system. In particular, we show how the algorithm can lead to a significant reduction in the noise components of the experimental point spread function.

Send correspondence to Raimund J. Ober; E-mail: ober@utdallas.edu.

Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing XII, edited by José-Angel Conchello, Carol J. Cogswell, Tony Wilson, Proc. of SPIE Vol. 5701 (SPIE, Bellingham, WA, 2005) · 1605-7422/05/$15 · doi: 10.1117/12.587051

In Section 2 we review the algorithm introduced in[12] and discuss its application to general three- and two-dimensional image sets. As a concrete example, in Section 3 we discuss the application of the algorithm to the smoothing of three-dimensional experimental point spread functions. Section 4 is devoted to a study of the improvement of deconvolution algorithms if smoothed point spread functions are used instead of non-smooth ones.

2. ALGORITHM

The algorithm that was developed by Ober et al.[12] is based on the generalization of a realization approach of one-dimensional sequences as it is encountered in systems and control theory.[13]

A general n-order linear state space description of a 1D discrete system with constant coefficients takes the form

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k) + Du(k),$$

$k = 0, 1, 2, \ldots$. The $n \times 1$ vector $x(k)$ is called the state vector. The input signal is the $m \times 1$ vector sequence $u(k)$, $k = 0, 1, 2, ...$, and $y(k), k = 0, 1, 2, ...$, is the $p \times 1$ output signal.[14] We have $A \in R^{n \times n}$, $B \in R^{n \times m}$, and $C \in R^{p \times n}$. The *impulse response* H of the system is the output sequence when $u(0) = 1$, and $u(k) = 0$, $k = 1, 2, 3, ...$, with zero initial state $x(0) = 0$. It is straightforward to verify that

$$H(k) = CA^{k-1}B, \ k = 1, 2, 3, \ldots$$

The parameter set (A, B, C) is also called a *state space realization* of the sequence $H(k)$, $k = 1, 2, 3, \ldots$. The converse problem is the one that is of relevance here. This *realization problem* addresses the question of given a sequence of real numbers/vectors/matrices $H(k)$, $k = 1, \ldots, N$, when does there exist a state space system (A, B, C) such that

$$H(k) = CA^{k-1}B, \ k = 1, 2, 3, \ldots, N.$$

A general realization result shows that for a finite sequence such a realization always exists.[13] The resulting system is, however, not unique. For example, a transformed (TAT^{-1}, TB, CT^{-1}) with a nonsingular matrix T will provide another realization of the same sequence. For numerical reasons the choice of a specific realization often plays an important role.

In[12] this realization approach was extended to three dimensional sequences given, for example, by a three dimensional pixelated finite image. A central result[12] is that a 3D discrete finite image set $P(k_1, k_2, k_3)$ $k_i = 1, \ldots, N_i$, $i = 1, 2, 3$, can be decomposed into three 1D sequences P_i, $i = 1, 2, 3$, i.e.,

$$P(k_1, k_2, k_3) = P_1(k_1)P_2(k_2)P_3(k_3), \quad k_i = 1, \ldots, N_i, \qquad (1)$$

$i = 1, 2, 3$. Each 1D sequence is then modeled as the impulse response of a system, i.e.,

$$P_i(k_i) = C_i A_i^{k_i - 1} B_i, \quad k_i = 1, 2, 3, ..., \quad i = 1, 2, 3. \qquad (2)$$

Therefore, a finite discrete 3D image set P can be represented as

$$P(k_1, k_2, k_3) = C_1 A_1^{k_1 - 1} B_1 C_2 A_2^{k_2 - 1} B_2 C_3 A_3^{k_3 - 1} B_3,$$

$k_i = 1, \ldots, N_i, \ i = 1, 2, 3$.[12]

To implement the realization algorithms the questions left are how to decompose the 3D image set P into three 1D sequences P_i, $i = 1, 2, 3$, and how to calculate the state space realization (A_i, B_i, C_i) for each 1D sequence P_i, $i = 1, 2, 3$. The specific algorithm introduced by Ober et al.,[12] is given in Section 6.

The algorithm depends crucially on singular value decompositions. Two singular value decompositions are used to decompose the 3D image set into three 1D sequences (1) while another three singular value decompositions are used to calculate the state space realizations for the three 1D sequences (2). The algorithm can be used in two modes. One is to obtain a full and exact representation of the 3D data set. For applications in

microscopy the approximation mode is probably more useful. For the purposes of approximation the singular value decompositions are used by 'dropping' small singular values in the factorization.

In each singular value decomposition, we will obtain a series of singular values. As a general rule, the larger a singular value is the more important it is for the approximation. The small singular values are insignificant and can be dropped in the approximation. Due to the presence of noise in the 3D image set, small singular values are corrupted by noise. By excluding those singular values, we are able to obtain a smoothed approximation of the noisy image set. Let r_1, r_2 denote the numbers of the dropped singular values in the first two singular value decompositions used to decompose the image set P and let s_1, s_2, s_3 be the numbers of the discarded singular values in the other three singular value decompositions used to calculate the state space realizations. The smoothed estimate of the 3D image set P obtained is then denoted as $P^{r_1,r_2;s_1,s_2,s_3}$.

This algorithm was applied to the noise reduction in 3D fluorescence microscopy images.[12] In the subsequent sections we will discuss the application of the algorithm to the noise reduction in experimentally acquired point spread functions.

3. NOISE SUPPRESSION OF POINT SPREAD FUNCTIONS

In this section, we illustrate how to use the algorithm to suppress noise of the experimental point spread function P_{exp} shown in Figure 1.

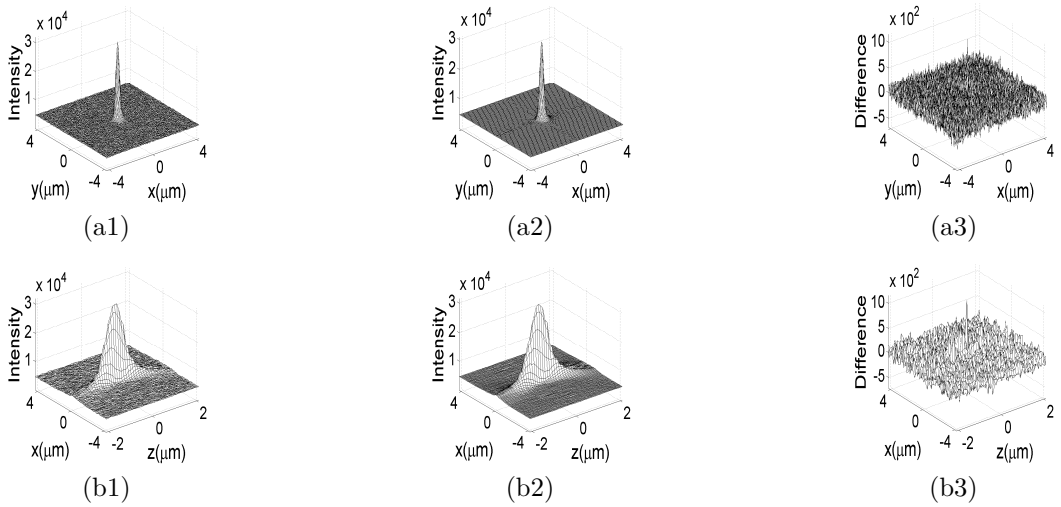

Figure 1. Comparison of the experimental point spread function P_{exp} and its smoothed estimate $P_{exp}^{122,119;29,64,9}$ (Section 3). **a1**, **a2**, **a3** show the $22th$ frame ($P_{exp}(k_1, 22, k_3)$, $k_1, k_3 = 1, \ldots, 128$), the $22th$ frame ($P_{exp}^{122,119;29,64,9}(k_1, 22, k_3)$, $k_1, k_3 = 1, \ldots, 128$) and their difference. These frames are cross sections of the 3D image sets orthogonal to the optical axis. **b1**, **b2**, **b3** show the cross section $P_{exp}(65, k_2, k_3)$, $k_2 = 1, \ldots, 43$, $k_3 = 1, \ldots, 128$, the cross section $P_{exp}^{122,119;29,64,9}(65, k_2, k_3)$, $k_2 = 1, \ldots, 43$, $k_3 = 1, \ldots, 128$ and their difference. These frames are cross sections of the 3D image sets along the optical axis. For more information about the coordinate system see Section 3.

The point spread function was acquired at $100nm$ increments and the pixel size is 67×67 nm^2 in the object space. The cropped image set P_{exp} consists of 43 frames each being a 128×128 pixel array. In[12] it was shown that the identification of the middle index of the pixel array with the optical axis is numerically favorable. Therefore, the middle index k_2 of P_{exp} is set to correspond to the optical axis, i.e. the z axis and the lateral axis x (y) is assigned to the index k_3 (k_1). As a result, the size of P_{exp} is $128 \times 43 \times 128$.

As described in Section 2, we suppress noise of the point spread function by excluding small singular values of the five singular value decompositions used in the algorithm. The key question is how to determine the number of retained singular values in each singular value decomposition in order to generate a smoothed estimate of the point spread function.

As an example we will show in detail how to determine the number of retained singular values in the first singular value decomposition. The same approach can be used to determine the number of retained singular values of the other singular value decompositions.

Figure 2. The singular values of the first singular value decomposition (Approximation Step 1) and their importance in the approximation. **a**. The singular values of the first singular value decomposition; **b**. The root-mean-square errors (RMSE) between the smoothed point spread functions and the experimental point spread function P_{exp}. The smoothed estimates are obtained by using different numbers of retained singular values in the first singular value decomposition (Approximation Step 1) but keeping 20 singular values of the second singular value decomposition and all singular values in the other singular value decompositions.

Figure 2.a shows the singular values of the first singular value decomposition used in the algorithm. We observe that the singular values change sharply. Only less than 10 singular values have significant values, while the rest of the singular values are relatively small. We generate the smoothed estimates of P_{exp} with different numbers of retained singular values of the first singular value decomposition but with the same number of retained singular values of the other singular value decompositions in the algorithm. Then we calculate the root-mean-square errors between the smoothed estimates and P_{exp} and plot them in Figure 2.b. (For two 3D sequences $P(k_1,k_2,k_3)$, $\hat{P}(k_1,k_2,k_3) \in \mathbb{R}$, $k_i = 1,2,\ldots,N_i$, $i = 1,2,3$, the root-mean-square error is defined by $\epsilon = \left(\frac{1}{N_1N_2N_3}\sum_{k_1=1}^{N_1}\sum_{k_2=1}^{N_1}\sum_{k_3=1}^{N_3}(P(k_1,k_2,k_3)-\hat{P}(k_1,k_2,k_3))^2\right)^{1/2}$.) As the number of retained singular values decreases, the root-mean-square error increases gradually because more noise components are suppressed. When the number of retained singular values is five or less, the root-mean-square error increases rapidly. This implies that the main features of the experimental point spread function are severely distorted in the approximation due to the loss of large singular values. We have to choose the number of retained singular values such that noise suppression is maximized without severe distortions in the approximation. There is no apparent optimal choice (Figure 2) and we have chosen to retain the first six singular values.

In a similarly way, we decide to retain 9, 100, 200, 120 singular values of the second, third, fourth and fifth singular value decompositions. The singular values of the last three singular value decompositions typically change more slowly. This is because the noise components in the experimental point spread function have already been suppressed to a large extent in the first two singular value decompositions.

The total numbers of the singular values of each singular value decomposition are 128, 128, 129, 264 and 129 and we discarded 122, 119, 29, 64 and 9 singular values in the first, second, third, fourth and fifth singular value decompositions. Following the notation introduced in Section 2 the smoothed point spread function is denoted as $P_{exp}^{122,119;29,64,9}$.

Cross sections of $P_{exp}^{122,119;29,64,9}$, shown in Figure 1, appear to be fairly smooth and the difference between the smoothed point spread function and the experimental point spread function appears as random noise.

4. DECONVOLUTION RESULTS

We tested the influence of noise suppression of point spread functions on the performance of the accelerated Richardson-Lucy algorithm[15] with both simulated data and acquired microscopy images.

From the results of the simulation we find that the root-mean-square error between the simulated images and the best recovered images using a smoothed point spread function is typically between one-fifth to one-fourth of the root-mean-square error between the simulated images and the best recovered images using a noise point spread function. In addition, as the number of the iterations increases, the recovered images using the noisy point spread function diverge significantly faster than the recovered images using the smoothed point spread function. It becomes clear that the performance of the accelerated Richardson-Lucy algorithm is very sensitive to the noise levels of the point spread functions.

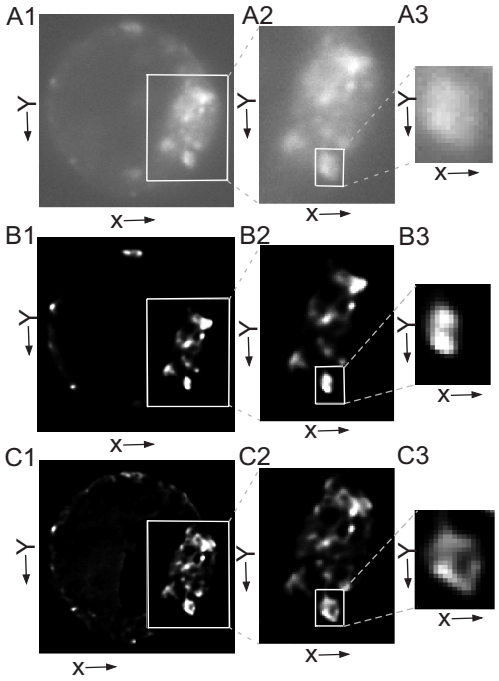

Figure 3. Comparison of the cropped recovered images of a Jurkat cell transfected with FcRn-GFP via the accelerated Richardson-Lucy algorithm with 300 iterations. **a1**, **b1** and **c1** show the cropped $40th$ frame ($Im_c(k_1, 40, k_3)$, $k_1 = 60, \ldots, 190$, $k_3 = 45, \ldots, 175$) of the acquired image set Im_c, the $40th$ frame of the recovered image set using P_{exp} (Figure 1), and $P_{exp}^{122,119;29,64,9}$ (Figure 1), respectively. The coordinate system is the same as in Figure 1.

We deconvolve an image set (Figure 3) of a Jurkat cell transfected by FcRn-GFP with the experimental point spread function P_{exp} (Figure 1) and the smoothed point spread function $P_{exp}^{122,119;29,64,9}$ (Figure 1) using the accelerated Richardson-Lucy algorithm with 300 iterations. Cross sections of both recovered images are shown in Figure 3. The difference between both recovered images is obvious. When the smoothed point spread function is used, more details are visible. For example, the typical doughnut shape of a sorting endosome is only visible in the reconstruction that uses the smoothed point spread function.

Further details on the application of the algorithm to the smoothing of point spread functions can be found in.[16]

5. CONCLUSIONS

In this paper we have discussed the use of the algorithm proposed in[12] to suppress noise in fluorescence mircoscopy image sets such as experimental point spread functions. The analysis has shown that this algorithm is capable of effectively suppressing randomness or noise in experimental point spread functions.

We have also examined the influence of noise suppression on the point spread function on the performance of the accelerated Richardson-Lucy algorithm. It has been shown that the deconvolution algorithm gives significantly improved results when smoothed point spread functions are used.

6. ACKNOWLEDGEMENTS

This research was supported in part by the National Institutes of Health (grants R01 AI50747 and R01 AI39167).

Appendix

Algorithm Let $P(k_1, k_2, k_3)$, $k_i = 1, 2, \ldots, N_i$, $i = 1, 2, 3$, represent a three-dimensional data array.

1. Subtract an estimated background level $\hat{b}$ (e.g. the mean of the data points near the boundary of the data array P) from data array P, and define $\Phi(k_1, k_2, k_3) := P(k_1, k_2, k_3) - \hat{b}$, $k_i = 1, 2, \ldots, N_i$, $i = 1, 2, 3$ (if no approximation is carried out, take $\hat{b} = 0$).

2. Arrange the entries of the resulting data array Φ to form a matrix Q_3 as

$$Q_3 = \begin{bmatrix} \Phi(1,1,1) & \Phi(1,1,2) & \cdots & \Phi(1,2,1) & \Phi(1,2,2) & \cdots & \Phi(1,3,1) & \Phi(1,3,2) & \cdots & \Phi(1,N_2,N_3) \\ \Phi(2,1,1) & \Phi(2,1,2) & \cdots & \Phi(2,2,1) & \Phi(2,2,2) & \cdots & \Phi(2,3,1) & \Phi(2,3,2) & \cdots & \Phi(2,N_2,N_3) \\ \Phi(3,1,1) & \Phi(3,1,2) & \cdots & \Phi(3,2,1) & \Phi(3,2,2) & \cdots & \Phi(3,3,1) & \Phi(3,3,2) & \cdots & \Phi(3,N_2,N_3) \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \Phi(N_1,1,1) & \Phi(N_1,1,2) & \cdots & \Phi(N_1,2,1) & \Phi(N_1,2,2) & \cdots & \Phi(N_1,3,1) & \Phi(N_1,3,2) & \cdots & \Phi(N_1,N_2,N_3) \end{bmatrix}.$$

3. (Approximation step 1) Decompose Q_3 via the singular value decomposition* as $Q_3 = U_1 \Sigma_1 V_1$. Partition $\Sigma_1 = diag(\hat{\Sigma}_1, \hat{\tilde{\Sigma}}_1)$, $U_1 = [\hat{U}_1, \hat{\tilde{U}}_1]$, and $V_1 = \begin{bmatrix} \hat{V}_1 \\ \hat{\tilde{V}}_1 \end{bmatrix}$ conformally, where $\hat{\Sigma}_1 \in \mathbb{R}^{K_1 \times K_1}$ and $\hat{\tilde{\Sigma}}_1 \in \mathbb{R}^{r_1 \times r_1}$. Define $\Phi_1^{r_1}(i) \in \mathbb{R}^{1 \times K_1}$, $i = 1, \ldots, N_1$, such that $\begin{bmatrix} \Phi_1^{r_1}(1) \\ \Phi_1^{r_1}(2) \\ \Phi_1^{r_1}(3) \\ \vdots \\ \Phi_1^{r_1}(N_1) \end{bmatrix} := \hat{U}_1 \hat{\Sigma}_1^{1/2}$ (for a diagonal matrix $D = diag(d_1, d_2, \ldots, d_n)$, $d_i \geq 0$, $i = 1, 2, \ldots, n$, we define $D^{1/2} = diag(d_1^{1/2}, d_2^{1/2}, \ldots, d_n^{1/2})$) and define $R_3^{r_1}(i) \in \mathbb{R}^{K_1 \times 1}$, $i = 1, \ldots, N_2 N_3$, such that $[R_3^{r_1}(1), \ldots, R_3^{r_1}(N_2 N_3)] := \hat{\Sigma}_1^{1/2} \hat{V}_1$. Note that r_1 denotes the number of discarded singular values in this step.

4. Arrange $R_3^{r_1}(1), \ldots, R_3^{r_1}(N_2 N_3)$ to form Q_2 as

$$Q_2 := \begin{bmatrix} R_3^{r_1}(1) & R_3^{r_1}(2) & R_3^{r_1}(3) & \cdots & R_3^{r_1}(N_3) \\ R_3^{r_1}(N_3+1) & R_3^{r_1}(N_3+2) & R_3^{r_1}(N_3+3) & \cdots & R_3^{r_1}(2N_3) \\ R_3^{r_1}(2N_3+1) & R_3^{r_1}(2N_3+2) & R_3^{r_1}(2N_3+3) & \cdots & R_3^{r_1}(3N_3) \\ \vdots & \vdots & \vdots & & \vdots \\ R_3^{r_1}((N_2-1)N_3+1) & R_3^{r_1}((N_2-1)N_3+2) & R_3^{r_1}((N_2-1)N_3+3) & \cdots & R_3^{r_1}(N_2 N_3) \end{bmatrix},$$

   where $Q_2 \in \mathbb{R}^{N_2 K_1 \times N_3}$.

5. (Approximation step 2) Decompose Q_2 via the singular value decomposition as $Q_2 = U_2 \Sigma_2 V_2$. Partition $\Sigma_2 = diag(\hat{\Sigma}_2, \hat{\tilde{\Sigma}}_2)$, $U_2 = [\hat{U}_2, \hat{\tilde{U}}_2]$, and $V_2 = \begin{bmatrix} \hat{V}_2 \\ \hat{\tilde{V}}_2 \end{bmatrix}$ conformally, where $\hat{\Sigma}_2 \in \mathbb{R}^{K_2 \times K_2}$ and $\hat{\tilde{\Sigma}}_2 \in \mathbb{R}^{r_2 \times r_2}$.

*A singular value decomposition of a two-dimensional $M \times L$ matrix Q is defined by the factorization $Q = U \Sigma V$, where U, V are matrices of sizes M-by-K and K-by-L, respectively, such that $U^* U = I$ and $V V^* = I$, in which I denotes an identity matrix, and $\Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_K) = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_K \end{bmatrix}$ with singular values $\sigma_i > 0$, $i = 1, \ldots, K$.

Define $\Phi_2^{r_2}(i) \in \mathbb{R}^{K_1 \times K_2}$, $i = 1, \ldots, N_2$, such that $\begin{bmatrix} \Phi_2^{r_2}(1) \\ \Phi_2^{r_2}(2) \\ \Phi_2^{r_2}(3) \\ \vdots \\ \Phi_2^{r_2}(N_2) \end{bmatrix} := \hat{U}_2 \hat{\Sigma}_2^{1/2}$, and define $\Phi_3^{r_2}(i) \in \mathbb{R}^{K_2 \times 1}$, $i = 1, \ldots, N_3$, such that $[\Phi_3^{r_2}(1),\ \Phi_3^{r_2}(2),\ \ldots,\ \Phi_3^{r_2}(N_3)] := \hat{\Sigma}_2^{1/2} \hat{V}_2$. Note that r_2 denotes the number of discarded singular values in this step.

6. (Approximation step 3, 4, 5) Calculate the realizations $(A_1^{r_1;s_1},\ B_1^{r_1;s_1},\ C_1^{r_1;s_1})$, $(A_2^{r_2;s_2},\ B_2^{r_2;s_2},\ C_2^{r_2;s_2})$ and $(A_3^{r_2;s_3},\ B_3^{r_2;s_3},\ C_3^{r_2;s_3})$ of $\Phi_1^{r_1}$, $\Phi_2^{r_2}$ and $\Phi_3^{r_2}$, respectively, via the realization step (see below) for some $s_1,\ s_2,\ s_3 \geq 0$, where $\Phi_1^{r_1}(i) \in \mathbb{R}^{1 \times K_1}$, $i = 1, \ldots, N_1$, $\Phi_2^{r_2}(i) \in R^{K_1 \times K_2}$, $i = 1, \ldots, N_2$, and $\Phi_3^{r_2}(i) \in \mathbb{R}^{K_2 \times 1}$, $i = 1, \ldots, N_3$, from step 3 and step 5. Note that s_1, s_2 and s_3 are the numbers of singular values discarded in the corresponding realization step.

7. Calculate the estimate $\Phi^{r_1,r_2;s_1,s_2,s_3}$ as

$$\Phi^{r_1,r_2;s_1,s_2,s_3}(k_1,k_2,k_3) = C_1^{r_1;s_1}(A_1^{r_1;s_1})^{k_1-1} B_1^{r_1;s_1} C_2^{r_2;s_2}(A_2^{r_2;s_2})^{k_2-1} B_2^{r_2;s_2} C_3^{r_2;s_3}(A_3^{r_2;s_3})^{k_3-1} B_3^{r_2;s_3}, \quad k_i = 1, 2, \ldots, N_i,\ i = 1, 2, 3. \tag{1}$$

8. Add the estimated background level $\hat{b}$ to the estimation $\Phi^{r_1,r_2;s_1,s_2,s_3}$ such that

$$P(k_1,k_2,k_3) \approx P^{r_1,r_2;s_1,s_2,s_3}(k_1,k_2,k_3) := \Phi^{r_1,r_2;s_1,s_2,s_3}(k_1,k_2,k_3) + \hat{b}, \quad k_i = 1, 2, \ldots, N_i,\ i = 1, 2, 3. \tag{2}$$

Realization step Let $P(i) \in \mathbb{R}^{p \times m}$, $i = 1, 2, \ldots, N$, be a finite one-dimensional sequence.

1. Construct the $(N+1)p \times (N+1)m$ Hankel matrix

$$H = \begin{bmatrix} P(1) & P(2) & \cdots & P(N-1) & P(N) & 0 \\ P(2) & P(3) & \cdots & P(N) & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ P(N) & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix},$$

   where 0 denotes a block of zeros of size $p \times m$.

2. Let $H = U \Sigma V$ be a singular value decomposition.
3. Partition $\Sigma = diag(\Sigma_1, \Sigma_2)$, $\Sigma_1 \in \mathbb{R}^{n \times n}$, $\Sigma_2 \in \mathbb{R}^{s \times s}$, $U = [U_1, U_2]$, $U_1 \in \mathbb{R}^{(N+1)p \times n}$, $U_2 \in \mathbb{R}^{(N+1)p \times s}$, and $V = \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}$, $V_1 \in \mathbb{R}^{n \times (N+1)m}$, $V_2 \in \mathbb{R}^{s \times (N+1)m}$, conformally. We also allow for the trivial partition in which the second components are empty, i.e. $s = 0$.
4. Let $C^s \in \mathbb{R}^{p \times n}$ be the first p rows of $U_1 \Sigma_1^{1/2}$.
5. Let $B^s \in \mathbb{R}^{n \times m}$ be the first m columns of $\Sigma_1^{1/2} V_1$.
6. Let $U_1 = \begin{bmatrix} \bar{U}_{11}^1 \\ \vdots \\ \bar{U}_{N1}^1 \\ \bar{U}_{(N+1)1}^1 \end{bmatrix}$, where $\bar{U}_{t_1 1}^1 \in \mathbb{R}^{p \times n}$ for all $t_1 = 1, \ldots, N+1$, and define $U_1^{\uparrow} = \begin{bmatrix} \bar{U}_{21}^1 \\ \vdots \\ \bar{U}_{(N+1)1}^1 \end{bmatrix}$ and $U_1^{\downarrow} = \begin{bmatrix} \bar{U}_{11}^1 \\ \vdots \\ \bar{U}_{N1}^1 \end{bmatrix}$. Then let $A^s = \Sigma_1^{-1/2} U_1^{\downarrow *} U_1^{\uparrow} \Sigma_1^{1/2} \in \mathbb{R}^{n \times n}$.

Note that the symbols r_1, r_2, s_1, s_2 and s_3 denote the numbers of dropped singular values in each singular value decomposition. When $r_1 = r_2 = s_1 = s_2 = s_3 = 0$ we have $P = P^{0,0;0,0,0}$, i.e a perfect reconstruction.[12] For the sake of simplifying the notation, we write $P^{rs} := P^{r_1,r_2;s_1,s_2,s_3}$, with $rs := \{r_1, r_2; s_1, s_2, s_3\}$.

REFERENCES

1. S. F. Gibson and F. Lanni, "Experimental test of an analytical model of aberration in an oil-immersion objective lens used in three-dimensional light microscopy," *Journal of the Optical Society of America* **8**(10), pp. 1601–1613, 1991.
2. J. G. McNally, C. Preza, J. A. Conchello, and J. L. J. Thomas, "Artifacts in computational optical-sectioning microscopy," *Journal of the Optical Society America A* **11**(3), pp. 1056–1067, 1994.
3. C. Preza, M. I. Miller, L. J. Thomas, Jr., and J. G. McNally, "Regularized linear method for reconstruction of three-dimensional microscopic objects from optical sections," *Journal of the Optical Society America A* **9**(2), pp. 219–228, 1992.
4. H. Chen, J. R. Swedlow, M. Grote, J. W. Sedat, and D. A. Agard, "The collection, processing, and display of digital three-dimensional images of biological specimens," in *Handbook of Biological Confocal Microscopy*, J. B. Pawley, ed., pp. 197–210, Plenum Press, New York, 2nd ed., 1995.
5. G. M. P. van Kempen and L. J. van Vliet, "Improving the restoration of textured objects with prefiltering," in *Proceeding of 3rd Annual Conference of the Advanced School for Computing and Imaging (ASCI'97)*, pp. 174–179, 1997.
6. P. J. Shaw, "Computer reconstruction in three-Dimensional fluorescence microscopy," in *Electronic Light Microscopy: The Principles and Practice of Video-Enhanced Contrast, Digital Intensified Fluorescence, and Confocal Scanning Light Microscopy*, D. M. Shotton, ed., ch. 9, pp. 211–230, Wiley-Liss, Inc., 1993.
7. P. J. Shaw and D. J. Rawlins, "The point-spread functions of a confocal microscope: its measurement and use in deconvolution of 3-D data," *Journal of Microscopy* **163**(2), pp. 151–165, 1991.
8. B. Lashgari, L. M. Silverman, and J. F. Abramatic, "Approximation of 2-D separable in denominator filters," *IEEE Transactions on Circuits and Systems* **CAS-30**(2), pp. 107–121, 1983.
9. K. Hirano, M. Sakane, and M. Z. Mulk, "Design of three-dimensional recursive digital filters," *IEEE Transactions on Circuits and Systems* **CAS-31**(6), pp. 550–561, 1984.
10. A. Doi and T. Hinamoto, "A spatial-domain technique for the design of 3-D separable-denominator state-space digital filters," *Multidimensional Systems and Signal Processing* **12**, pp. 89–98, 2001.
11. T. Hinamoto and A. Doi, "Design of multidimensional separable-denominator digital filters in the spatial domain," in *Proceedings of IEEE Asia Pacific Conference on Circuits and Systems*, pp. 219–222, (Seoul, Korea), 1996.
12. R. J. Ober, X. Lai, Z. Lin, and E. S. Ward, "State space realization of a three-dimensional image set with application to noise reduction in fluorescent microscopy images of cells," *Multidimensional Systems and Signal Processing* **16**, pp. 7–48, 2005.
13. T. Kailath, *Linear systems*, Prentice-Hall, Inc., New Jersey, 1980.
14. W. J. Rugh, *Linear system theory*, Prentice-Hall, Inc., New Jersey, 2nd ed., 1996.
15. D. S. C. Biggs and M. Andrews, "Acceleration of iterative image restoration algorithms," *Applied Optics* **36**(8), pp. 1766–1775, 1997.
16. X. Lai, Z. Lin, E. S. Ward, and R. J. Ober, "Noise suppression of point spread functions and its influence on deconvolution of three-dimensional fluorescence microscopy image sets," *Journal of Microscopy* **217**, pp. 93–108, 2005.