IMPROVING RESTORATION OF MICROSCOPY IMAGES USING ITERATIVE PROTOTYPES AND A SEQUENCE OF SUPPORT CONSTRAINTS

Moacir P. Ponti-Jr

ICMC — Universidade de São Paulo — Campus de São Carlos
13560-970 São Carlos, SP, Brazil

Webpage: http://www.icmc.usp.br/~moacir — E-mail: moacir@icmc.usp.br

ABSTRACT

Images obtained by wide-field microscopy are specially corrupted by an out-of-focus blur on the axial direction. Constrained restoration algorithms are able to restore some of the frequencies lost on the acquisition process by using prior knowledge. We report an improved restoration algorithm using prototype images produced by the iterative Richardson-Lucy algorithm and a sequence of finite support constraint sets. The finite support shrinks as the nonlinear restoration is performed. Results showed an improved restoration on images with dense background, achieving a larger passband and restoration on axial direction. It encourages the development of new restoration algorithms based on the projection onto sequence of sets.

Index Terms — image restoration, microscopy, POCS

1. INTRODUCTION

Optical imaging systems often degrade the acquired images. Image restoration methods are designed to remove or reduce these effects, particularly for the reduction of blur and noise. In wide-field (WF) microscopy, images are severely affected by an out-of-focus blur and also by a signal dependent noise. There are recent efforts to improve such systems in order to acquire higher resolution and low noise images. However, most equipments currently used and widely available are based on WF microscopy, encouraging studies that aim to improve image quality after acquisition.

The restoration of images based on a priori information about the image and the system makes possible to estimate frequency components of the image that are not passed through the imaging system, and to reconstruct signals with missing samples or irregularly-sampled spaces [1]. The prior knowledge can be used to impose constraints that the restored image must satisfy. One of the most used restoration algorithm [1], even currently, is the Richardson-Lucy (RL) [2], that uses the point-spread function (PSF) of the system to obtain restoration through a probabilistic approach. Projection onto convex sets (POCS) is another method that can be used to perform constrained restoration, modeling prior information as constraints sets [3]. From 2006 to 2011, related studies tried to improved restoration, including a RL with Total Variation regularization [4], and an iterative filtering band extrapolation [5].

This study proposes a simple approach to improve the RL algorithm by using a POCS prototype image framework based on a sequence of finite support constraints. The RL algorithm is used due to its simplicity and known quality of the results. We use the fact that optical microscopy images are often sparse (background-dominated) to remove intensities at regions outside the object support and improve restoration.

As far as we know there are no previous studies including sequence of finite support sets and iterative image prototype to improve iterative restoration. This method is expected to restore frequencies within and outside the frequency support of the imaging system and consequently produce images of higher quality.

2. FLUORESCENCE MICROSCOPY IMAGES

The 3D microscopy image reconstruction is carried out by acquiring 2D images on several focal planes. These slices are stacked to generate a 3D image. The WF microscope uses light to illuminate the whole specimen, so the camera captures light coming from the focal plane and also from the planes above and below, causing an out-of-focus blur in both the lateral and (mostly) the focal plane directions.

The general imaging system for a WF microscope can be modeled by considering the object to be imaged as a function \( f(\mathbf{x}) \) in the 3D real space, where \( \mathbf{x} = (x, y, z) \), and a point-spread function (PSF) \( h(\mathbf{x}) \) of the system. Due to the photon counting nature of light-based sensors, the main source of noise is a signal-dependent noise, well modeled by a Poisson distribution. Considering each value of the observed image, \( g(\mathbf{x}) \), to be a realization of a random variable described by a Poisson process \( P \{ \} \), and \( * \) a convolution operator, the observed image formation is given by:

\[
g(\mathbf{x}) = P \{ f(\mathbf{x}) * h(\mathbf{x}) \}.
\]
Unfortunately, the support of practical OTF (Optical Transfer Function) – the normalized Fourier transform of the PSF – is defined by a small portion of the whole complex volume. Therefore, an inverse filter often does not exist. Fig. 1 depicts a frequency domain support for a WF microscope, showing \( \rho = (u, v)^{1/2} \) against \( w \) to illustrate how low and high frequencies are affected along the \( w \) axis.

![Fig. 1. Example of a wide-field microscope optical transfer function support](image)

3. RESTORATION METHODS

We propose a restoration algorithm based on a sequence of prototypes produced by RL iterations, and a sequence of finite support constraint sets. Each RL iteration is followed by a projection onto the current finite support convex set.

3.1. POCS framework

The first set and its projector are based on the image prototype set definition \([3]\). It is carried out by applying an operator \( O_n \) in the current estimate of the image \( \hat{f}_n \), and can be seen as a “correction” operator applied to \( f_n \). The \( x \) is going to be removed from the first set and projector equations to simplify the notation. We define a sequence of operators \( O_n \) as:

\[
O_n = \left[ \frac{g}{f_{n-1} \ast h} \right] \cdot h. \tag{2}
\]

The set of images that are deconvolved by the operator is defined as a sequence \( n = 1, \ldots, m \) of sets:

\[
C_1^m = \left\{ \hat{f}_n : O_n \cdot \hat{f}_{n-1} \right\}. \tag{3}
\]

Using the triangle inequality, it can be shown that the “deconvolution set” \( C_1^m \) is convex \([4]\). Given a sequence \( \hat{f}_1, \ldots, \hat{f}_m \) in \( C_1^m \), with limiting point \( \hat{f} \in C_1^m \), the projector is given by:

\[
\hat{f}_n^* = P_1^n \cdot \hat{f}_{n-1} = O_n \cdot \hat{f}_{n-1} \tag{4}
\]

Algorithm 1 – POCS-SPIS

**INPUT:** Observed image \( g \), PSF of the system \( h \), size of the step \( s \), object support limit \( I \), residual thresholds \( \alpha \) and \( \beta \), and maximum of iterations \( N \).

**OUTPUT:** Restored image \( \hat{f} \).

**AUXILIARY:** Correction operator \( O \), residual \( \text{res} \), current iteration \( n \), current amount of performed reductions on each support direction \( r = (r_x, r_y, r_z) \).

1. \( \hat{f}_0(x) \leftarrow g(x) \).
2. \( n \leftarrow 1 \).
3. \( S_n(x) \leftarrow 1, (\forall x) \).
4. \( r \leftarrow (0, 0, 0) \).
5. Do,

   6. \( O(x) \leftarrow \frac{g(x)}{(h \ast f_{n-1})(x)} \ast h(x) \).
   7. \( \hat{f}_n^*(x) \leftarrow (O \ast \hat{f}_{n-1})(x) \).
   8. \( \text{res} \leftarrow \sum \left| f_n^*(x) \right| \ast \text{res}_n \).
   9. If \( \text{res} < \beta \), then
      10. \( L \left( S_n(x), r \right) \leftarrow \text{shrink-s}(S_{n-1}(x), r, s, l) \)
      11. Else
      12. \( L \left( S_n(x) \right) \leftarrow S_{n-1} \)
      13. \( \hat{f}_n(x) \leftarrow (S_n \ast \hat{f}_{n-1}^*)(x) \).
      14. While \( \text{res} > \alpha \) or \( n < N \).
      15. Return \( \hat{f}_n \).

The convex set that represents the support is:

\[
C_2^n = \left\{ \hat{f}_n^* : \hat{f}_n^* \cdot S_n \right\}, \tag{5}
\]

where \( S_n \) is a function designed to remove the background region, at iteration \( n \). The projector \( P_2^n \) is defined as:

\[
P_2^n \cdot \hat{f}_n^*(x) = \hat{f}_n^* \cdot S_n. \tag{6}
\]

On each iteration, the image is deconvolved using the RL iteration and constrained to the support:

\[
\hat{f}_n = P_2^n \cdot P_1^n \cdot \hat{f}_{n-1}, \tag{7}
\]

where \( \hat{f}_{n-1} \) is an estimate of the true image at iteration \( n - 1 \), and \( \hat{f}_0 = g \). Before each projection, \( O_n \) is calculated using Eq. 2 and the spatial support \( S_n \) is computed as further described in section \([5, 2]\). This method is going to be called POCS-SPIS, where SPIS stands for sequence of prototype images and supports, and is described in Algorithm 1. The projections are performed until the residual reaches \( \alpha = 0.001 \). The residual is defined as:

\[
\text{res} = \frac{\left\| \hat{f}_n - \hat{f}_{n-1} \right\|}{\left\| \hat{f}_{n-1} \right\|} \tag{8}
\]

3.2. Computing the object support

We propose a region of support \( S_n \) that changes as the restoration is performed. Since the initial observed image is severely
blurred, there is important data in regions outside the object support, that cannot be removed. However, as the image is iteratively restored, the objects outside the support are often artifacts or background objects that can be removed and improve the restoration.

The idea is to use a result from the Gerchberg-Papoulis method, to extrapolate frequencies by imposing a support constraint \( \ell \). The residual is used as a criterion to modify \( \ell \). The first support is set to the entire image. When the residual drops below \( \beta = (2 \times \alpha) = 0.002 \), i.e., when \( \text{res} < \beta \), the support is reduced by \( s \) voxels on each side.

It is necessary to define three parameters: the step size, \( s \) — how much the support is going to be reduced —, and \( \ell = (l_x, l_y, l_z) \), the object support limit, so that \( \ell \) is limited to be reduced to specific voxels at each directions \( x, y, z \). The default setting is \( s = 1 \), i.e., reduce one voxel from all sides, while defining the limit requires knowledge about the image. The Algorithm 2 describes the shrinking process.

To reduce artifacts caused by the support boundary regions, we use image edge smoothing \( \mathbf{11} \) by a 3D gaussian filter with \( \sigma = 1.5 \) and kernel size proportional to the maximum amount of reductions performed on directions \( x, y, z \), stored in variable \( r = (r_x, r_y, r_z) \) of Algorithm 2.

### 4. EXPERIMENTS AND RESULTS

A series of experiments were carried out to show how the proposed method can improve restoration. The image rectangles was built with size \( 128 \times 128 \times 128 \) and blurred with a theoretical model for the WF PSF \( \mathbf{7} \). The image cells with size \( 256 \times 256 \times 32 \), was also blurred with a PSF and corrupted with Poisson noise. Two WF fluorescence images, \( \mathbf{RP} \), with size \( 256 \times 256 \times 32 \) and \( 20 \times \) lateral amplification \( \mathbf{4} \). All images were restored using the original RL algorithm and the POCS-SPIS. Before restoration, the background was removed using the maximum of histogram approach \( \mathbf{8} \).

The settings for the POCS-SPIS algorithm were \( s = 1 \) for all images, and \( l \) was manually defined for each image. For the real image \( l = (0, 0, 4) \), so that the support can only shrink in \( z \) direction. Evaluation was conducted by using Improvement on Signal-to-Noise Ratio (ISNR), the Universal Image Quality Index (UIQI) \( \mathbf{9} \) and by computing the image practical passband, i.e., the number of Fourier coefficients that are higher than 1% of the frequency domain peak, to assess frequency restoration.

Table 1 shows the numerical results of synthetic images restoration and Table 2 the passband for all images. Figures 2, 3, and 4 shows samples of resulting images. The proposed method achieved better numerical results and enhanced frequencies, obtaining sharper images with improved contrast. The restoration in \( z \)-axis direction was also superior as can be seen in Figure 5.

---

**Algorithm 2 — SHRINK-S: SHRINK SUPPORT**

**INPUT:** Support \( S(x) \), current amount of reductions on each direction \( r = (r_x, r_y, r_z) \), size of the step \( s \) and object support limit \( \ell = (l_x, l_y, l_z) \).

**OUTPUT:** Modified support \( S(x) \) and reductions vector \( r \).

**AUXILIARY:** Index \( i \), support size \( n \times m \times q \), maximum amount of reductions \( a \)

1. \( (n, m, q) \leftarrow \text{sizeof}(S(x)) \)
2. \( r^* \leftarrow r \)
3. If \( (r_x + 1 < l_x) \), then
   - For each line \( i = (r_x + 1), \ldots, (r_x + s) \), do
     - \( r_x^* \leftarrow r_x^* + s \)
4. If \( (r_y + 1 < l_y) \), then
   - For each line \( i = (r_y + 1), \ldots, (r_y + s) \), do
     - \( r_y^* \leftarrow r_y^* + s \)
5. If \( (r_z + 1 < l_z) \), then
   - For each line \( i = (r_z + 1), \ldots, (r_z + s) \), do
     - \( r_z^* \leftarrow r_z^* + s \)
6. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
7. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
8. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
9. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
10. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
11. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
12. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
13. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
14. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
15. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
16. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
17. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
18. \( S((n - (i - 1))[y] , z) = 0, (\forall y, z \in x) \)
19. \( a = \max(r) \)
20. \( S(x) = \text{gaussianfilter3D}(S(x), a, 1) \)
21. Return \( (S(x), r) \)

**Table 1. Synthetic image restoration evaluation**

<table>
<thead>
<tr>
<th>rectangles</th>
<th>cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISNR</td>
<td>UIQI</td>
</tr>
<tr>
<td>RL</td>
<td>5.47</td>
</tr>
<tr>
<td>POCS-SPIS</td>
<td>6.20</td>
</tr>
</tbody>
</table>

**Table 2. Practical passband of real, observed and restored images**

<table>
<thead>
<tr>
<th>rectangles</th>
<th>cells</th>
<th>RP1</th>
<th>RP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real image</td>
<td>2.120</td>
<td>3.790</td>
<td>—</td>
</tr>
<tr>
<td>Observed image</td>
<td>72</td>
<td>56</td>
<td>68</td>
</tr>
<tr>
<td>RL</td>
<td>985</td>
<td>1.001</td>
<td>704</td>
</tr>
<tr>
<td>POCS-SPIS</td>
<td>1.568</td>
<td>2.048</td>
<td>1.045</td>
</tr>
</tbody>
</table>

### 5. CONCLUSIONS

A restoration method was proposed based on a sequence of constraint support sets and prototype images using RL iterations for deconvolution. Iterative prototype images and
changing supports triggered by a residual criterion were not explored before, as far as we know. The POCS-SPIS method improved the restoration, and it is suitable for sparse images when there is knowledge about the object support.

The proposed method require some parameters to run, but we believe the results encourages the development of new methods using sequence of sets and adaptive projections. The question of whether the proposed approach is promising to other applications and methods remains open. Also, convergence and overall speed are not discussed. Those are questions left for future investigation.

6. REFERENCES


