Universitat Politècnica de Catalunya (UPC)
Universitat Autònoma de Barcelona (UAB)
Universitat de Barcelona (UB)
Institut de Ciències Fotòniques (ICFO)

http://www.photonicsbcn.eu

Master in Photonics

MASTER THESIS WORK

Fast image restoration in light-sheet fluorescence microscopy with extended depth of field using GPUs

David Castillo Andreo

Supervised by Dr. Pablo Loza-Alvarez, (ICFO)

Presented on date 10th September 2015

Registered at

Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona

Fast image restoration in light-sheet fluorescence microscopy with extended depth of field using GPUs

David Castillo Andreo

ICFO-The Institute of Photonic Sciences, Av. Carl Friedrich Gauss, 3, 08860 Castelldefels, Barcelona - SPAIN

E-mail: David.Castillo-Andreo@latribu.com

Abstract. Light sheet fluorescence microscopy (LSFM) is used in many biological research experiments that require fast three-dimensional (3D) imaging up to a few volumes per second. Wavefront coding (WFC) microscopy provides LSFM with 3D real-time imaging capabilities introducing a controlled aberration that extends the depth of field (DOF) of the microscope objective. Resulting images, however, are blurred and need to be processed to recover the original sharpness in a CPU-intensive deconvolution which makes real-time visualization unattainable. We present computational tools based on GPU parallel computing to achieve real-time deconvolution and visualization of the images obtained with WFC light-sheet microscopy.

1. Introduction

Three-dimensional (3D) imaging has become key in many research fields, for example to understand the complex relations of cells cultivated in extra-cellular matrix gels or in dynamic systems. One the most successful techniques used to obtain 3D images of the samples is Light-sheet fluorescence microscopy (LSFM)[1] which is based on a light sheet that illuminates the sample from the side at the focal plane of a wide-field fluorescence microscope. A 3D data set is obtained by scanning the sample while recording the fluorescence with a camera. The out-of-focus signal is reduced due to the fact that only the imaged part of the sample is illuminated (optical sectioning) achieving results comparable with those obtained with confocal laser scanning microscopy but with the additional advantage of the reduction of photo-bleaching outside of the illuminated area.

There is however another significant constraint when imaging thick fast moving or changing samples with scanning LSFM; working at high resolution implies the use of high numerical aperture (NA) lenses which lead to a reduction of the depth of field (DoF) [2] as illustrated in figure 1. To obtain a focused 3D image with this very thin plane of focus the sample needs to move. The movement of the sample disturbs the desired conditions of acquisition and imposes a mechanical limit to the acquisition speed.

One of the methods used to overcome these limitations and extend the DoF is to introduce a controlled aberration in the optical system in what is called Wavefront Coding (WFC) [2]. The effective Point Spread Function (PSF) of the combined setup is invariant over a large range of defocus values but the images obtained are uniformly

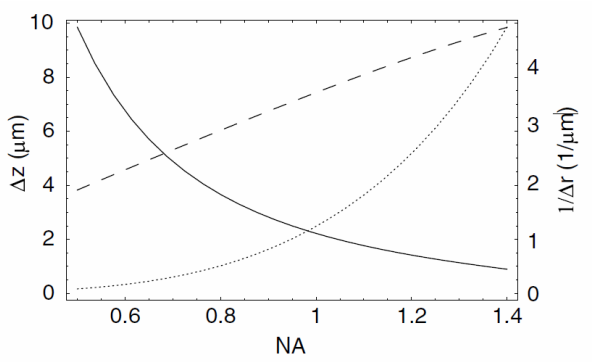
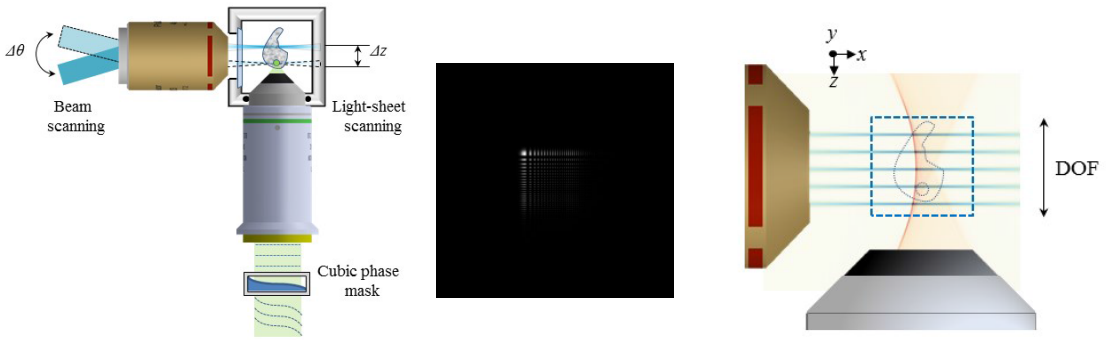


Figure 1: Depth of field (solid line), lateral resolution (dashed line) and peak intensity at focus (dotted line – arbitrary units) for an oil immersion ($n_{oil} = 1.518$) aplanatic microscope objective with a typical range of NA and $\lambda_0 = 0.53$ μm is the vacuum wavelength.

blurred over a large range along the optical axis. The sharpness of the original image can be recovered with a final deconvolution with the known PSF.

The use of WFC in LSFM is normally done by placing a cubic phase mask at the exit pupil of an objective lens as in figure 2(a) in an inexpensive upgrade [3]. The combination of LSFM and WFC provides powerful 3D capabilities with fast acquisition and reduced photo-bleaching. One of the key advantages of the WFC-LSFM in the biological field is the possibility of optical sectioning the sample without moving it. This becomes specially interesting in cases where the movement of the sample disturbs the desired conditions of acquisition. All these advantages are at the expense of the need of digital post-processing and a modest reduction of the signal-to-noise ratio (SNR) compared to the in-focus widefield image [2].



(a) (b) (c)

Figure 2: 2(a) describe the scheme of the optical setup of WFC on a light-sheet microscope. Figure 2(b) is the PSF of the system with the cubic phase mask at the exit pupil of the objective lens. 2(c) illustrates the extended DoF produced by the cubic phase mask and the optical sectioning possibilities in a LSFM

The current project focuses on the post-processing step of the WFC-LSFM. Recovering the original image sharpness requires a final CPU-intensive deconvolution process which takes to the order of seconds. In many cases, specially for large living

samples, this delay produces annoying problems to the microscopist who has to choose between watching the blurred image or wait for the deconvolution. He is obliged to wait for some seconds before moving the sample at every shot aiming for the desired lateral or axial position while dynamics are occurring inside. The work presented here provides computational tools to perform real-time deconvolution and visualization of the images obtained with a Wavefront Coding Light-Sheet microscope. To speed up such deconvolution, processing and visualization steps are performed in the GPU at a speed close to video rate.

This master thesis starts with a mathematical background of the deconvolution process including the filters implemented in the software: the simple yet efficient wiener filter and the second tikhonov regularization. Then it will describe the different technologies used in the application and its structure. Some results and figures are shown at the end of this report.

2. The deconvolution process

We can approximate the output of any imaging system as a convolution of the original image with the system's impulse response or Point Spread Function (PSF). In the case of Wavefront Coding the obtained image is severely blurred by the effect of the cubic phase mask. The term "deconvolution" used in this report refers to the process of reversing the optical distortion caused by the mentioned cubic phase mask.

There exist different software algorithms for the deconvolution of images both with the estimation of the PSF of the system and without knowing it (blind deconvolution). In our application we use inverse filter algorithms in the Fourier space. Due to its wide range of applications in image processing the Fast Fourier Transform (FFT) algorithms are already implemented and optimized in most Graphics Processor Units (GPU). For that reason, we base the implementation of our deconvolution algorithm in the FFT. We know that our obtained image is a convolution of the original image with the PSF (h) of the system and, according to the convolution theorem, it's a simple product in the Fourier space:

$$\begin{gathered} i_{image}(x,y) = \sum_i \sum_j i_{object}(x,y) * h(x-x_i, y-y_j) + n(x,y) \\ \Updownarrow \mathscr{F} \\ I_{image}(u,v) = I_{object}(u,v)H(u,v) + N(u,v) \end{gathered} \tag{1}$$

The most intuitive inverse filter that first come after looking at equation (1) is just a division of the blurred image by the Fourier transform of the PSF, called Optical Transfer Function (OTF). However, there are two main drawbacks for this simple filter to be effective: first, the OTF of the system has to be invertible and second, noise is amplified by the division operation.

2.1. The wiener filter

The first filter used in our deconvolution software is the wiener filter [4] defined in the Fourier space as:

$$G(u,v) = \frac{1}{H(u,v)}\left[\frac{|H(u,v)|^2}{|H(u,v)|^2 + \frac{N(u,v)}{S(u,v)}}\right] = \frac{H(u,v)^*}{|H(u,v)|^2 + \frac{1}{SNR(u,v)}} \tag{2}$$

If we have an estimation of the PSF and the signal-to-noise ratio (SNR) of the system, the implementation of the wiener filter is straightforward and it requires only one single step. Although the mentioned simplicity the wiener filter is quite efficient as it minimizes the mean square error.

In our application we will consider that the SNR is approximately constant all over the image reducing the term $\frac{1}{SNR(u,v)}$ to a constant term.

2.2. The second order tikhonov regularization

Generalized solutions in image restoration like the wiener filter are not optimal under some conditions. The fact of constraining the solution to the minimum error above all else leads to unstable solutions in noisy images. To obtain stable solutions in the presence of noise we use regularization methods. The purpose of the regularization is to introduce prior knowledge in the restoration of images to identify meaningful estimations and avoid the oscillations produced by the noise in the generalized methods [5].

One of the most used methods in the image restoration field is the Tikhonov regularization. Tikhonov regularization introduces prior knowledge of the original image by including a term in the original least square function used in generalized methods. The filter is defined in the Fourier space as:

$$G(u,v) = \frac{H(u,v)^*}{|H(u,v)|^2 + \alpha^2|\mathcal{L}(u,v)|^2} \tag{3}$$

where α is the regularization parameter and $\mathcal{L}(u,v)$ is a matrix which in the case of the second order is the discrete Laplacian operator. Different orders of the tikhonov regularization use other matrices instead of the Laplacian operator. The second order tikhonov regularization favors smooth solutions by penalizing curvature.

3. The deconvolution software

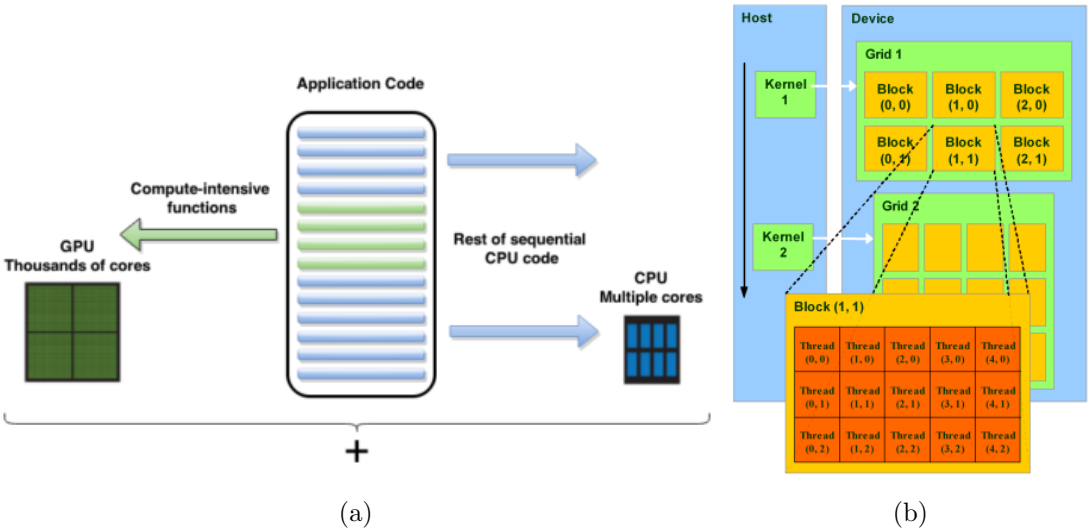
3.1. General-purpose computing on graphics processing units

The main purpose of the project is to use parallel computing to speed up the deconvolution process and obtain speeds as close as possible to real-time. Here is where general-purpose computing on graphics processing units (GPGPU) comes to the rescue. GPGPU is the use of a graphics processing unit (GPU) together with a CPU to accelerate applications that are computationally intensive. GPUs are present in every

modern computer, tablet or mobile phone and its main purpose is to perform intensive graphics computations which are intrinsically parallel.

A set of low-levels operations to be executed in the GPU is called a kernel. A single kernel call in the host (CPU) runs the kernel multiple times in parallel on the GPU. Each independent, concurrent execution is called a thread. Threads are grouped in equally-sized blocks, and blocks are distributed in grids as illustrated in figure 3(b).

There exist two main platforms for GPGPU programming: CUDA (Compute Unified Device Architecture) created by GPU manufacturer NVIDIA and OpenCL which is the standard for cross-platform (even working in NVIDIA devices).



(a) (b)

Figure 3: 3(a) illustrates how CPUs and GPUs work in conjunction to accelerate computing. CPUs have a few cores optimized for sequential serial processing while GPUs have thousands of smaller and efficient cores in a massive parallel architecture. 3(b) shows the grid/block/thread structure of GPUs

3.2. Software architecture

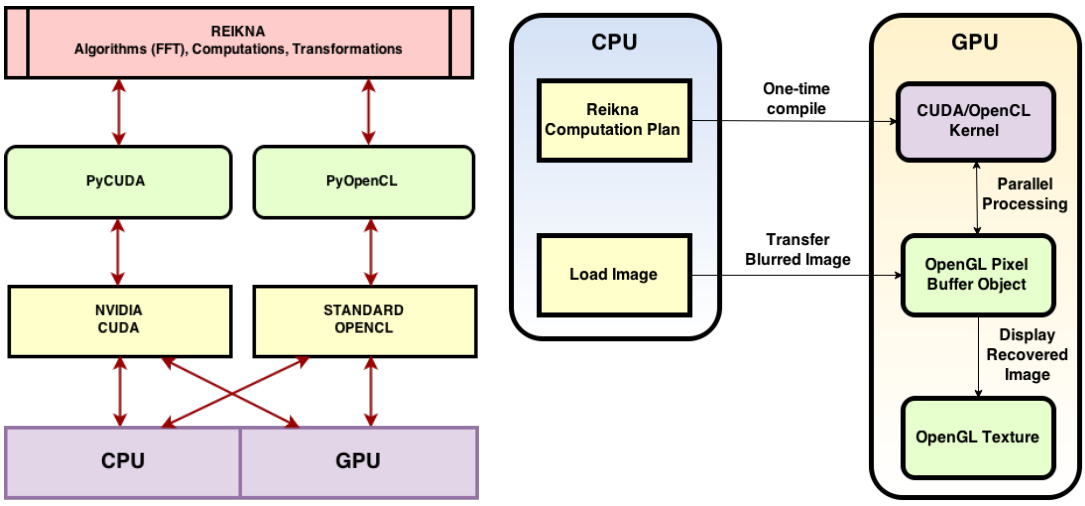
The programming language we have selected for the deconvolution application is python, a high-level programming language widely used in the scientific community due to its mature scientific libraries and readability. In particular we employ reikna as the highest abstraction layer to generate CUDA and OpenCL optimized kernels as shown in figure 4(a).

Reikna is a python library containing various GPU algorithms built on top of PyCUDA and PyOpenCL. PyCUDA is a python wrapper for the NVIDIA's CUDA api for parallel programming. It allows to generate and run CUDA kernels from python. PyOpenCL is PyCUDA equivalent for standard OpenCL api. One of the algorithms implemented in reikna is the Fast Fourier Transform (FFT) which as mentioned before is widely used in image processing. Reikna provides as well an abstraction layer in the preparation of the computation using a modular approach that does not compromise the

execution performance. Finally it allows the application to run in both main competing technologies CUDA and OpenCL, detaching the choice from the underlying hardware.

For visualization we use OpenGL libraries in order to interface directly with the GPU without passing through the CPU memory. By integrating reikna with OpenGL we accelerate the rendering of the resulting images avoiding unnecessary transfers from the GPU to the CPU.

Figure 4(b) illustrates the flow of execution and how CPU and GPU are combined to accelerate the computation. On run-time the CUDA or OpenCL kernel is compiled one time and loaded into the GPU. Then, on each iteration the input blurred image is first loaded to the RAM memory and transferred to an OpenGL pixel buffer object (PBO) in the GPU. The kernel operates with the data applying the deconvolution and leaving the result in the same PBO. Before the next iteration the PBO is used to fill an OpenGL texture which displays the result. Both PBO and texture remain in the GPU avoiding unnecessary transfers of the image back to the CPU.



(a) (b)

Figure 4: 4(a) is a block diagram representing the different software layers of the deconvolution software. 4(b) illustrates the execution flow in CPU and GPU.

4. Results

Two main versions of the software have been developed: real-time single image and volume deconvolution. Although both versions share similar computations, the first one gives real-time deconvolution of a single input image coming from the microscope camera while the second one gives information of a series of images which are part of a volume. They are designed to work in the two major GPU development platforms: CUDA and OpenCL, ensuring compatibility with almost all GPU brands. The filters discussed in section 2 have been developed in the computations and the user has the

possibility to modify its parameters ($\frac{1}{SNR(u,v)}$ in the wiener filter or α in the second order tikhonov regularization) while the deconvolution is in process.

4.1. Real-time single image deconvolution

Figure 5 shows the main graphical interface of the real-time single image deconvolution software.

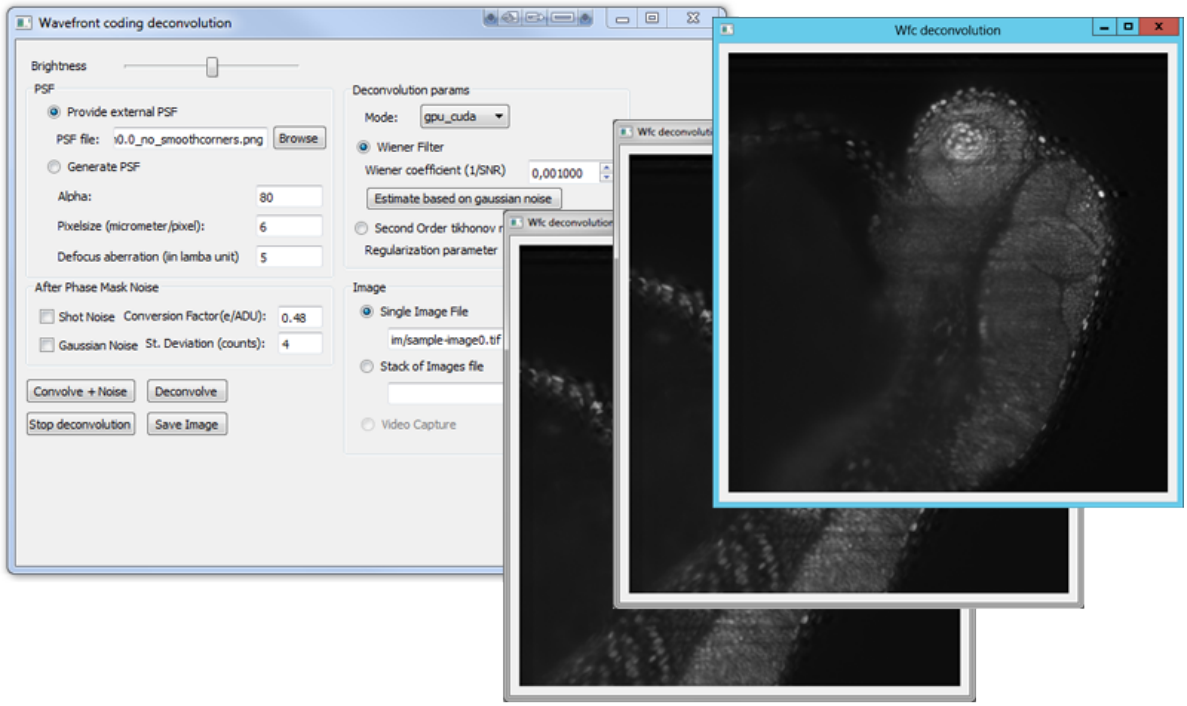


Figure 5: Main interface of the real-time single deconvolution software.

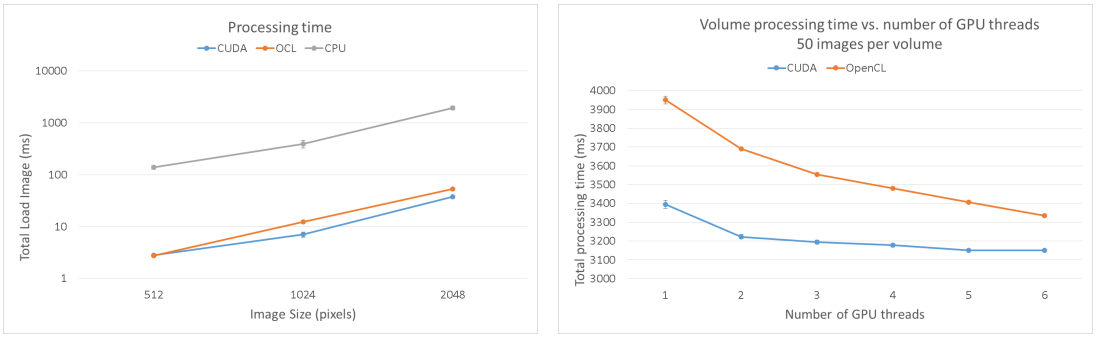
Although its main purpose is to work directly attached to the microscope's camera, the tool includes other options intended to generate figures for this report, like the possibility to save the resulting image, or to evaluate the performance of the deconvolution process, like the noise generation. The noise of the WFC-LSFM microscope has been modeled as a contribution of Gaussian noise and shot noise according to the specifications of the camera. There exists the possibility of generating an estimated PSF to use in the deconvolution process [6] or select it as an external image.

Using this GPU-based application we have reduced the deconvolution time of a 2048x2048 image from approximately 2s in a standard† CPU processing down to around 40ms (up-to 50 times faster) as illustrated in figure 6(a). OpenGL provided an additional speed up to the visualization that allowed us to present fully deconvolved images close to real-time performance.

† Intel Xeon Processor E5-1650 v2, 3.5GHz, 6 cores.

4.2. Volume deconvolution

Figure 7(a) shows the main graphical interface of the volume deconvolution software. The application has two modes of operation to provide information of the volume of images: the maximum intensity projection (MIP) and the three planes projection. The MIP produces images that are projections along one direction of the pixels with highest intensity throughout the volume. As seen in figure 7(b) and 7(c), three cartesian projections are generated: axial, lateral and frontal. Each line of the lateral and frontal projections is expanded for better visualization. In the plane projection mode a single point of the volume is selected and the images of the cartesian planes crossing that point are shown. The selected point can be changed during deconvolution allowing the user to visualize with great detail the desired 3D position of the sample.



(a) (b)

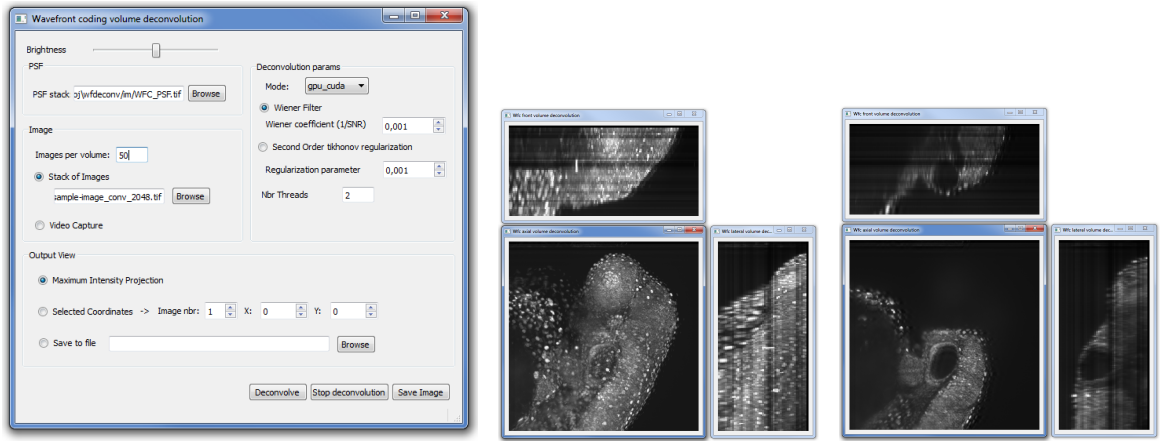
Figure 6: 6(a) shows the processing time reduction attained by using a GPU-based deconvolution. Figure 6(b) illustrates the reduction in time by using multiple GPU threads in the deconvolution of a volume composed by 50 images. Experimental data corresponds to a NVIDIA Quadro K2000 GPU device with 384 CUDA Cores.

In this version of the software we include the possibility of using several independent GPU threads of computation. The use of those concurrent threads assures the full utilization of the GPU capacity and allow us to reduce even more the deconvolution time per volume as shown in figure 6(b).

4.3. Image restoration

To observe the effect of the WFC mask and the deconvolution process we use the synthetic test image of figure 8(a) which has a steady decaying frequency spectrum. According to the specifications of the camera a low Gaussian and shot noise has been added to the test image to simulate the microscope acquisition (mse‡ = 0.0018). Then, we simulate the WFC mask by convolving the image 8(a) with the cubic phase mask PSF to obtain figure 8(b) (mse=74.66). We finally use the deconvolution software with the wiener filter and the parameter $\frac{1}{SNR(u,v)}$ set to 0.001 to restore the image blurred

‡ Mean squared error (mse) $= \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$.



(a) (b) (c)

Figure 7: 7(a) is the main interface of the volume deconvolution software. 7(b) illustrates the deconvolution of a volume with 50 images in the (Maximum Intensity Projection) MIP mode. 7(c) illustrates the deconvolution of a volume with 50 images in the three planes projection mode.

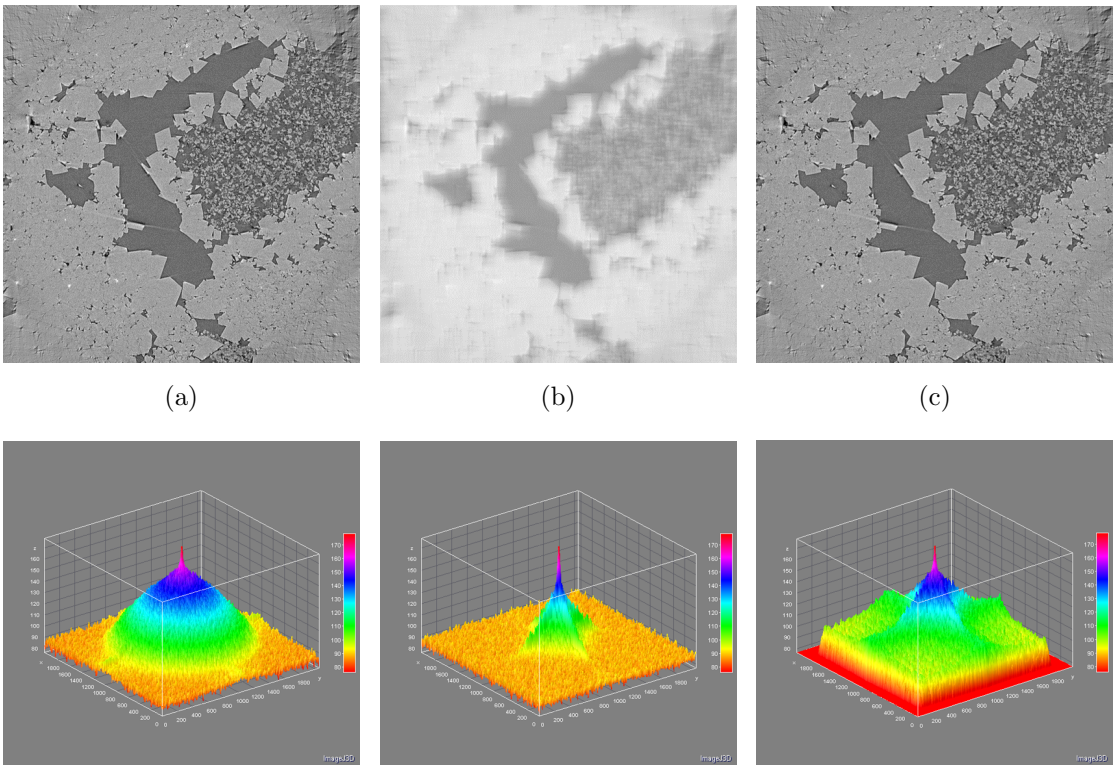
by the mask. The result is shown in figure 8(c) (mse=4.86). As discussed in the Introduction (1), in the deconvolution process there is a loss in fidelity compared to the in-focus widefield image. The effect is better seen in the frequency domain. In figure 8(e) we observe that the effect of the WFC mask is an attenuation of high frequencies as compared with the original frequency spectrum 8(d). The deconvolution tries to recover the original image but while some frequencies are over attenuated specially out of the axis others containing noise are increased as illustrated in figure 8(f).

5. Conclusions and future work

We have developed a tool for both deconvolution and visualization of volumetric microscopy images obtained with a wavefront coding microscope. The use of GPUs allowed us to perform the required tasks at a speed close to video rate. Further optimization on the code and the GPU hardware would boost the processing speed over the video rate limit.

Currently, the software is under active development to include the control of the microscope in the imaging pipeline. Particularly, the software for scanning the light sheet and its synchronization with both the acquisition and the volumetric deconvolution is being implemented. This synchronization will ensure the correct temporal and spatial assignment of the dynamic structures within recorded volumes. I hope this work will help to advance the field of 3D microscopy of fast biological dynamics. The findings of this work have resulted in two communications in major conferences of the field:

1. Poster presented at: Focus on Microscopy 2015, Göttingen (Germany), March 29 - April 1, 2015. "GPU-based deconvolution and visualization for wavefront coding microscopy", D. Castillo-Andreo, O.E. Olarte & P. Loza-Alvarez.



(a) (b) (c)

(d) (e) (f)

Figure 8: 8(a) is a synthetic test image. 8(d) is the Fast Fourier Transform of 8(a). 8(b) is the convolution of 8(a) with the cubic phase mask PSF. 8(e) is the Fast Fourier Transform of 8(b). 8(c) is the deconvolution of 8(b) using the deconvolution software presented in this report with a wiener parameter of 0.001. 8(f) is the Fast Fourier Transform of 8(c).

2. Abstract submitted to: BiOS - Photonics West 2016, San Francisco (USA). February 13 - 18 2016. "Use of a GPU for fast deconvolution in wavefront coding light sheet imaging", D. Castillo-Andreo, J. Licea, O.E. Olarte & P. Loza-Alvarez.

References

[1] Swoger J, Pampaloni F, Stelzer EH. 2014. Light-sheet-based fluorescence microscopy for three-dimensional imaging of biological samples. Cold Spring Harb Protoc 2014:1–8.

[2] M. R. Arnison, C. J. Cogswell, C. J. R. Sheppard and P. Török, "Wavefront coding fluorescence microscopy using high aperture lenses," in Optical imaging and microscopy: techniques and advanced systems, P. Török and F.-J. Kao, eds., Springer-Verlag, Berlin, 143-165 (2003).

[3] Omar E. Olarte, Jordi Andilla, David Artigas, and Pablo Loza-Alvarez, "Decoupled illumination detection in light sheet microscopy for fast volumetric imaging," Optica 2, 702-705 (2015)

[4] Lim, Jae S.. Two-Dimensional Signal and Image Processing. Englewood Cliffs, NJ: Prentice-Hall, 1990.

[5] W. Clem Karl, "Regularization in image restoration and reconstruction," in Handbook of Image and Video Processing, Second Edition, (Elsevier, 2005).

[6] Tingyu Zhao and Feihong Yu, "Point spread function analysis of a cubic phase wavefront coding system with a circular pupil," Opt. Express 20, 2408-2419 (2012).