

REAL-TIME REGISTRATION AND DISPLAY OF CONFOCAL MICROSCOPE IMAGERY FOR MULTIPLE-BAND ANALYSIS

Scott E. Budge, Anoop M. Mayampurath

James C. Solinsky

Electrical and Computer Engineering Dept.
Utah State University
Logan, UT 84322-4120
scott.budge@ece.usu.edu, anmenon@cc.usu.edu

National Security Directorate
Pacific Northwest National Laboratory
Richland, WA 99352
jim.solinsky@pnl.gov

ABSTRACT

Multiple images acquired in real-time from a confocal microscope in different illumination wavebands are used to construct a three dimensional understanding of living cell signal propagation. Registration, fusion, and display of these images is performed in real-time using Field Programmable Gate Array (FPGA) technology to allow the biological scientist to observe time critical signaling processes as they occur. The registration and display system operates at a rate of up to 15 frames/s, and performs dark offset correction and noise reduction, followed by non-linear correction for lens warp and focal plane tilt, and finally correction for spatial mismatch between the different wavebands. Parameters for the registration process are determined automatically from calibration images at the start of each experimental session in real-time. The resulting fused images exhibit significantly enhanced detail as illustrated by examples presented in the paper.

1. INTRODUCTION

Intercellular communication can be better studied if the approach deals with empirical conditions of observing signaling, i.e in 3-D. The confocal microscope aids in this aspect, as the microscope has a Z-plane added to its observation domain, thus enabling effective three dimensional reconstruction of the cell. Ultimately, the real-time observation of signaling mechanisms can be aided with the application of fluorescence resonance energy transfer (FRET) techniques to images captured during laser excitation in the confocal microscope [1].

The Pacific Northwest National Laboratory (PNNL) houses a confocal microscope that is designed to capture 800x600 pixel images from two different cameras simultaneously, one capturing at 647 nm (red) and the other at 488 nm (green)

This work was performed with support from the Pacific Northwest National Laboratory. Special thanks go to Ken Perrine and Marianne Sowa-Resat for their help.

from a cell sample excited with a laser at these wavelengths [2, 3]. These two images have to be registered and fused in order to enable FRET analysis. At present, the microscope is augmented with FPGA-based hardware to enable the real-time processing of the images and the eventual capability of real-time FRET.

This paper describes methods that were used by the authors to correct non-linear and linear distortions in the two images in real time on the Celoxica FPGA platform. While the focus of this paper is on confocal imagery only, the methods described can be generalized so as to apply to any other domain with minor changes.

2. GEOMETRIC DISTORTION CORRECTION PARAMETER DETERMINATION

Prior to registration, the two images will have to be corrected for geometric distortions. The distortions are of two kinds. The first is non-linear warping, which includes a lens warp distortion found in each of the cameras and focal-plane tilt with respect to the optical path. The second is linear distortion that occurs due to differences in the focal plane location and optical paths for the two cameras which causes misalignment of the images. The registration and display are done in real time by calculating the linearized and registered images from the parameters that model the spatial mismatch between the two waveband images.

2.1. Warp and Tilt Correction

Geometric distortion can be thought of as a mapping from the undistorted image space to the distorted image space. The mapping can be represented as the transform function

$$x = f(u, v) \quad (1)$$

and

$$y = g(u, v), \quad (2)$$

where (x, y) are the coordinates in the distorted image and (u, v) are the coordinates in the undistorted image. The mapping that was chosen for the tilt and lens warp is given by [4, 5]

$$x = a_x + b_x u + c_x u^2 + d_x u^3 + e_x v + f_x uv + i_x v^2 + j_x uv^2 \quad (3)$$

and

$$y = a_y + b_y u + c_y u^2 + e_y v + f_y uv + g_y v u^2 + i_y v^2 + m_y v^3. \quad (4)$$

The higher order terms compensate for the curvature of the warp, and the cross (uv) terms for the tilt in the images. Since we don't know the exact angle of tilt nor lens distortion present in the microscope, a method for correction was developed using control points. For this, a set calibration images from each camera was taken, which consisted of an image containing horizontal bars and a second image containing vertical bars. A set of control points was chosen from the distorted calibration images. This was compared to constructed control points from an ideal linear image and a match was found. By finding the least-squares solution for the coefficients, the distortion can be properly modeled. A different set of coefficients must be found to linearize each of the channels.

2.1.1. Basic Processing

Since we have two channels, the correction was applied to each independently of the other. Multiple images of the same kind, (vertical and horizontal bar), were averaged so as to reduce any noise present. Using a Sobel edge detector, the edges of the vertical and horizontal images were detected, and then the two images were combined to form a grid. The edge detection was done prior to adding the vertical and horizontal images as this was found to give better results than after the images were added together. The grid image was then filtered using 3×3 kernel, $[0 \ 1 \ 0, \ 1 \ 1 \ 1, \ 0 \ 1 \ 0]$, to find the intersection points. These form the control points in the distorted image. A threshold was set to remove noise pixels, and all points which had a size of more than one pixel were deleted manually.

2.1.2. Matching

For the construction of an ideal linear grid with line spacing similar to the calibration images, a set of four points from the calibration image representing the horizontal and vertical bar spacing were chosen and measured. Since we know that there is warp distortion present in the calibration image, only one such set of points were chosen, and the points came from the same part of the image to reduce error caused by the warping. Based upon the distance calculated, the grid was created. Using a simple neighborhood search

technique, each control point in the calibration images was matched to a grid point in the ideal image to result in the image shown in Figure 1.

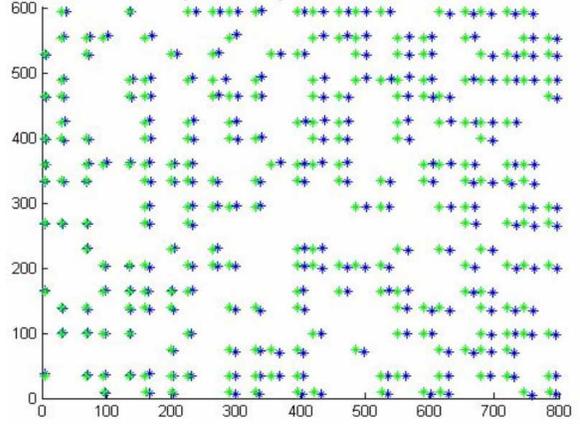


Figure 1: Calibrated image control points (blue) and matched constructed grid points (green).

The coefficients were computed as follows. Let us define

$$\mathbf{A} = \begin{bmatrix} 1 & u_1 & u_1^2 & u_1^3 & v_1 & u_1 v_1 & v_1^2 & u_1 v_1^2 \\ 1 & u_2 & u_2^2 & u_2^3 & v_2 & u_2 v_2 & v_2^2 & u_2 v_2^2 \\ \vdots & \vdots \\ 1 & u_n & u_n^2 & u_n^3 & v_n & u_n v_n & v_n^2 & u_n v_n^2 \end{bmatrix}, \quad (5)$$

$$\mathbf{k} = [a_x \ b_x \ c_x \ d_x \ e_x \ f_x \ i_x \ j_x]^T, \quad (6)$$

and

$$\mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]^T, \quad (7)$$

where \mathbf{A} contains points from the ideal linear image, \mathbf{x} contains points from the distorted calibration image, and n is the number of control points. The coefficients that model the distortion can be calculated by

$$\mathbf{k} = \mathbf{A}^+ \mathbf{x}, \quad (8)$$

Where \mathbf{A}^+ is the Moore-Penrose pseudoinverse. The solution for the y coefficients in (4) are obtained in a similar manner.

2.1.3. Determination of the Optical Center

To determine the optical center of the image, which is used as the center of rotation and scaling for the linear registration step, the shifted warping equations are given by

$$x - u_0 = (u - u_0) + \kappa(u - u_0)^3 + \kappa(v - v_0)^2(u - u_0) \quad (9)$$

and

$$y - v_0 = (v - v_0) + \kappa(v - v_0)^3 + \kappa(u - u_0)^2(v - v_0) \quad (10)$$

where (x, y) are the coordinates in the distorted image (control points), (u, v) are the coordinates in the undistorted image (grid points), κ represents a warping parameter, and (u_0, v_0) are the coordinates of the optical center of the image.

Solving for x and y , (9) and (10) can be written as

$$x = a_1 + a_2u + a_3v + a_4uv + a_5(3u^2 + v^2) + a_6(v^2u + u^3) \quad (11)$$

$$y = b_1 + b_2u + b_3v + b_4uv + b_5(u^2 + 3v^2) + b_6(u^2v + v^3) \quad (12)$$

subject to the conditions $b_4 = 2a_5$, $b_5 = \frac{1}{2}a_4$, $b_6 = a_6$, and $b_2 = a_3$. The coefficients in (11) and (12) can then be found using the least-squares technique given in (5), (6), and (7).

After the coefficients were found, the values of u_0, v_0 and κ can be found easily from

$$\kappa = b_6, u_0 = -\frac{a_5}{\kappa}, \text{ and } v_0 = -\frac{b_5}{\kappa}.$$

2.1.4. Correction Results

The warp parameters were calculated from data extracted from the calibration images as described above. Figure 2 shows the map from the undistorted image space (green points) to the distorted image space (blue points) in the form of red points for the red channel images. As can be seen, the map is quite accurate.

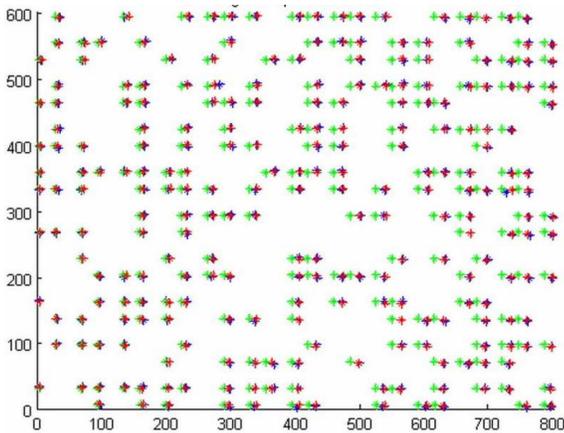


Figure 2: Map for images from channel one.

2.2. Affine Transform Correction

To register the two linearized images, it is necessary to perform an affine transform (rotation, scale, and translation) on one of the images to match it to the other image. In our design, we decided to perform the affine transform on the red channel image after the non-linear warp was removed from

both channels so that rotation, scaling, or translation differences between the frames (caused by thermal expansion, vibration, or accidental bumping of the camera set-up) could be removed at the beginning of each experimental session. (See Section 3.1.) Although this results in degraded image sharpness by interpolating the transformed image twice, it was done to allow for real-time calibration of the registration parameters through a least-squares minimization process without hard-coding the transform parameters.

The affine transform used is given by

$$x = u_T + s \cos \theta u - s \sin \theta v \quad (13)$$

$$y = v_T + s \cos \theta v + s \sin \theta u \quad (14)$$

where the transform parameters s (scale factor) and θ (rotation angle) can be replaced by the terms $k_1 = s \cos \theta$ and $k_2 = s \sin \theta$, reducing the number of parameters needed to four.

3. REAL-TIME HARDWARE IMPLEMENTATION

The data rate for the confocal microscope is 15 800x600 pixel images/s from each camera. We desired to register the images in real-time using custom-designed hardware implemented in a field-programmable gate array (FPGA) from Xilinx. The design was completed using the Handel-C programming language and compiled into gates using the DK Suite 2.1 tools from Celoxica, Inc., and then place-and-routed using ISE version 6.1i tools from Xilinx. The entire design used about 1,280,000 gates and runs at 40 MHz.

The top level design included a host-side interface program running in Microsoft Windows which captured the frames from the camera and then sent the data from the capture buffer to the FPGA coprocessor card, where the frames were registered and returned to the host for display in real-time.

The FPA coprocessor used in this design contained a single Xilinx XC2V6000 FPGA and 6 32-bit wide busses attached to external memory banks. Each memory bank was large enough to contain two complete frame buffers. One of the memory banks was reserved to hold correction offsets to remove fixed-pattern noise and another was reserved for future enhancement of the algorithm.

3.1. Concurrent Processing Design

The processing flow graph is given in Figure 3. In the figure, the circles represent processes and the rectangles represent reads/writes to memory frame buffers. Processes that run concurrently are connected by solid lines with a “C” associated, and sequential execution is represented by dashed line connections.

The two concurrent processes on the left side of the figure, “Read From Host” and “Coadd,” represent processes

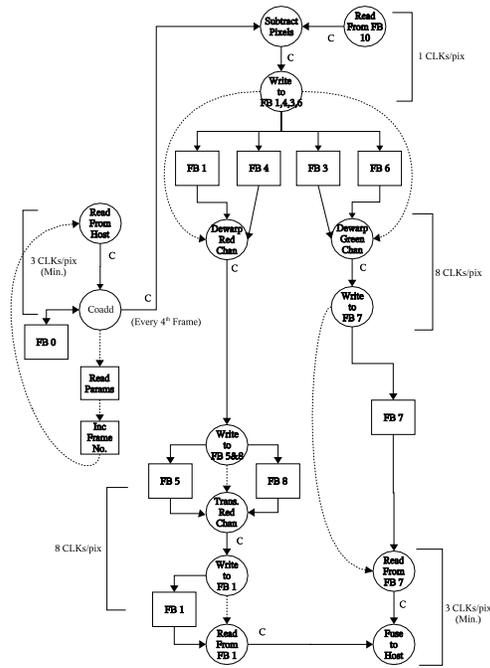


Figure 3: Processing flow graph.

running each frame time. In order to reduce the effects of noise from the cameras, every four frames are coadded (averaged) before the resulting frames are registered for display. This is done at a rate of 3 CLK/pixel minimum. Every four frames, the following processing steps are completed:

1. Dark current offsets (obtained from dark image calibration each experimental session) are read from memory and subtracted to remove fixed-pattern noise, and the resulting images are stored in memory in duplicate copies of each of the channels. This takes 1 CLK/pixel.
2. The polynomial transforms ((3) and (4)) are performed concurrently for each channel and four pixels/channel are read and used to bilinear interpolate the transformed image pixels. The transformed images are stored to memory. This takes 8 CLKs/pixel to perform in a highly pipelined design.
3. The red channel image is transformed using the affine transform given in (13) and (14), bilinear interpolated, and stored to memory. This takes 8 CLKs/pixel.
4. The two channels are read from memory, fused together pixel-by-pixel, and written back to the host. This is done at a rate of 3 CLK/pixel minimum.

Thus, the entire registration process can be completed at a rate of 20 CLKs/pixel. The performance-limiting step in this implementation was the interpolation process. This is because of the need to read four pixels (two per access) from the input image for each interpolated pixel, requiring two memory access times.

3.2. Multiplierless Polynomial Implementation

One of the goals of the design was to eliminate the dependency of the computations in (3), (4), (13), and (14) on multipliers. This was done by modifying the method described in [6] to apply to our polynomials, as shown in Figure 4 for the implementation of (3). In the figure, the A registers rep-

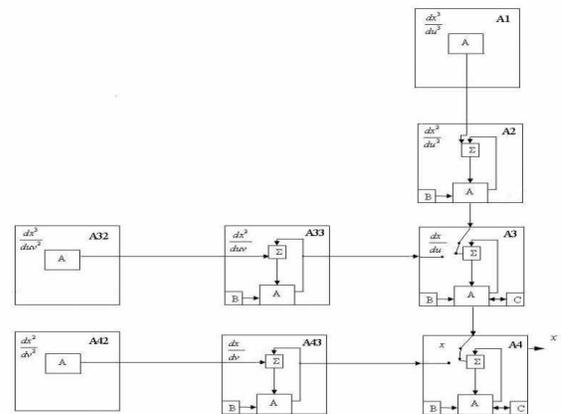


Figure 4: Multiplierless polynomial computation.

resent accumulators. The B registers contains A's value at the start of a frame, and the C register contains A's value at the start of an image-line. The switches in the right column blocks are switched once at the start of a line to connect the blocks on the left side of the figure. This implementation requires no multipliers and a total of 14 adders to compute (3) and (4) in a single clock on row pixels and two clocks on the left edge pixels. For more detail on the process, see [6].

4. REGISTRATION RESULTS

To test the performance of the design, a calibration grid slide was mounted on the confocal microscope stage and the camera system was set to capture pairs of images at approx. 15 frames/s. A pair of fixed-pattern noise frames was then captured by eliminating all lighting and averaging 32 frames together. An FPGA calibration configuration similar to the one described above was then used to find the least-squares optimal affine transform parameters by matching, in real-time, the grid images from both channels and storing them

in memory interfaced to the FPGA. The parameters for the non-linear warp were hard-coded into the FPGA configuration.

Once the calibration process was completed, a slide containing animal tissue was mounted on the microscope. The system was then used to perform continuous real-time registration of the two image channels. An example of the results before applying the warp removal and registration algorithms is given in Figure 5, and the results after registration, captured from the image stream, is given in Figure 6. Particularly note the section of the image marked by the red line.

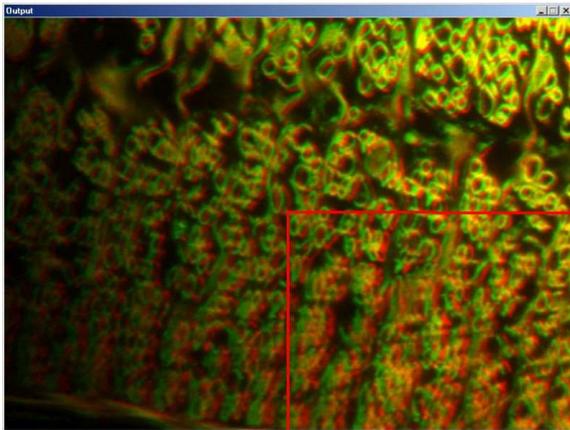


Figure 5: Fused image before registration.

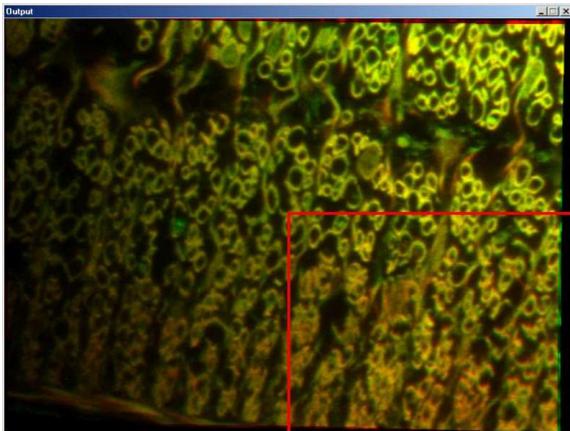


Figure 6: Fused image after registration.

5. CONCLUSIONS

On-going research into fundamental biological processes at PNNL requires the ability to perform real-time FRET. To

develop this capability, it was necessary to design hardware able to register, in real-time, images captured simultaneously for immediate display. This allows the biologist to search quickly through a test sample and focus on cell signaling processes as they occur.

The system reported in this paper was successful in accomplishing the goals above. The design fits easily into the Xilinx XC2V6000 FPGA available to the project, and performs at a rate of 20 CLKs/pixel at 40 MHz, or 2 Mpixels/s. The performance of the system is not limited by the design; the 40 MHz clock rate was the lowest necessary to perform the registration in real-time. As demonstrated, the registration was successful in removing the effects of focal-plane tilt and lens warp in both channels, and effectively registered the two channels for clear visual evaluation.

6. REFERENCES

- [1] G. W. Gordon, G. Berry, X. H. Liang, B. Levine, and B. Herman, "Quantitative fluorescence resonance energy transfer measurements using fluorescence microscopy," *Biophysical Journal*, vol. 74, pp. 2702–2713, May 1998.
- [2] J. C. Solinsky, S. E. Budge, P. D. Majors, and B. Rex, "Real-time image analysis of living cellular-biology measurements of intelligent chemistry," in *Sixth International Conference on Quality Control by Artificial Vision* (K. Tobin, ed.), vol. 5132, (Bellingham, WA), SPIE, 2003.
- [3] M. S. Resat, J. C. Solinsky, H. S. Wiley, K. A. Perrine, T. A. Seim, and S. E. Budge, "3-d multispectral monitoring of living-cell signaling using confocal-imaging and FPGA processing," in *IEEE International Symposium on Biomedical Imaging (ISBI 2004)*, (Washington, D.C.), pp. 680–683, 2004.
- [4] J. C. Russ, *The Image Processing Handbook*, ch. 2, pp. 94–95. CRC Press, 1992.
- [5] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, ch. 9, pp. 456–457. PWS Publishing, second ed., 1999.
- [6] A. G. J. Nijmeijer, M. A. Boer, C. H. Slump, M. M. Sampson, M. J. Bentum, G. J. Laanstra, H. Snijders, J. Smit, and O. E. Herrmann, "Correction of lens-distortion for real-time image processing systems," in *Workshop on VLSI Signal Processing VI*, pp. 316–324, IEEE, Oct. 1993.