LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-JRNL-402821

Fast Implementation of Matched Filter Based Automatic Alignment Image Processing

A. A. S. Awwal, K. Rice, T. Taha

April 9, 2008

Optics and Laser Technology

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Fast Implementation of Matched Filter Based Automatic Alignment Image Processing

Abdul A. S. Awwal[1], Kenneth L. Rice[2], and Tarek M. Taha[2]
[1]*National Ignition Facility, Lawrence Livermore National Laboratory, Livermore, CA. 94551*
[2]*Electrical and Computer Engineering Department, Clemson University, Clemson, SC 29634*
E-mail: awwal1@llnl.gov,{krice,tarek@clemson.edu}

ABSTRACT

Video images of laser beams imprinted with distinguishable features are used for alignment of 192 laser beams at the National Ignition Facility (NIF). Algorithms designed to determine the position of these beams enable the control system to perform the task of alignment. Centroiding is a common approach used for determining the position of beams. However, real world beam images suffer from intensity fluctuation or other distortions which make such an approach susceptible to higher position measurement variability. Matched filtering used for identifying the beam position results in greater stability of position measurement compared to that obtained using the centroiding technique. However, this gain is achieved at the expense of extra processing time required for each beam image. In this work we explore the possibility of using a field programmable logic array (FPGA) to speed up these computations. The results indicate a performance improvement of 20 using the FPGA relative to a 3 GHz Pentium 4 processor.

Key word: pattern recognition, matched filtering, optical alignment, automated optical alignment, automated target recognition, parallel computing, FPGA.

1. INTRODUCTION

The National Ignition Facility, currently under construction at the Lawrence Livermore National Laboratory, is a stadium-sized facility containing a 192-beam, 1.8-megajoule, 500-terawatt, ultraviolet laser system for the study of inertial confinement fusion and the physics of matter at extreme temperatures and pressures [1]. Automatic alignment (AA) based on computer analysis of video images adjusts the laser beams quickly and accurately enough to meet stringent system requirements in less than 30 minutes. The AA system directs all 192 laser beams along the 300-m optical path to focus on a 50 micron spot at the target chamber center [2]. At the heart of this alignment technique is the image processing algorithm that determines the position of beam features that are embedded in images recorded along the beam path. Varieties of alignment fiducials incorporated in the optical system designate various beam types, such as reference beams and main beams. Many beam images have well-defined spot profiles (e.g., Gaussian beams) for which centroiding is an acceptable technique to determine positions within the required accuracy of one half pixel. However, laser beam images often exhibit intensity variation or other distortions for which the centroid-based approach may result in high position uncertainty. In these cases, matched filtering provides an excellent and stable position measurement [3,4], albeit at the expense of extra processing time required for each beam image. This paper discusses an approach to speed up these computations using field programmable logic array (FPGA). A performance improvement of 20 was achieved using the FPGA relative to a 3 GHz Pentium 4 processor.

2. MATCHED FILTER BASED POSITION DETECTION

The matched filtering technique utilizes a given object as a template, whose position is known, to find the position of a second object by detecting the template's matching position in the correlation domain. The *classical matched filter* (CMF) [5] and its variation *phase only filter* (POF) [6] has gained popularity due to its ability of detecting an object with high discrimination to the presence of strong noise and background distortions. In the CMF, the complex amplitude and phase of the reference pattern is used, whereas POF only uses the phase of the reference pattern to perform the correlation [6]. The *amplitude modulated phase only filter* (AMPOF) [7,8] was designed to further enhance filtering performance by modulating the POF by an inverse type of amplitude.

The Fourier domain treatment of the matched filter is described next. Let the Fourier transform of the to-be-detected object (template) function *f* (x, y) be denoted by:

$$F(U_x,U_y) = \left|F(U_x,U_y)\right| \exp(j\Phi(U_x,U_y)) \tag{1}$$

and that of the input scene *g(x, y)* containing the desired object to be represented by

$$G(U_x,U_y) = \left|G(U_x,U_y)\right| \exp(j\Psi(U_x,U_y)) \tag{2}$$

A classical match filter (CMF) corresponding to this function *f* (x, y) is expected to produce its autocorrelation. From the Fourier transform theory of correlation, the CMF is given by the complex conjugate of the input Fourier spectrum as denoted by Eq. 3.

$$H_{CMF}(U_x,U_y) = F^*(U_x,U_y) = \left|F(U_x,U_y)\right| \exp(-j\Phi(U_x,U_y)) \tag{3}$$

The inverse Fourier transformation of the product of *F* (U_x, U_y) and H_{CMF} (U_x, U_y) results in the convolution of *f* (x, y) and *f* (-x, -y), which is the equivalent of the autocorrelation of *f* (x, y). Moreover, when |*F* (U_x, U_y)| is set to unity, H_{CMF} becomes a *phase only filter* (POF):

$$H_{POF}(U_x,U_y) = \exp(-j\Phi(U_x,U_y)) \tag{4}$$

The correlation of input image with the template is simply:

$$C_{CMF}(\Delta x,\Delta y) = F^{-1}\left\{G\left(U_x,\ U_y\right)H_{CMF}\left(U_x,U_y\right)\ \right\} \tag{5}$$

The position of the object can be found from the position of the cross-correlation, auto-correlation, and the position of the template using Eqs. 6-7.

$$\mathrm{x}_{\mathrm{pos}} = \ \mathrm{x}_{\mathrm{cross}} - \mathrm{x}_{\mathrm{auto}} + \mathrm{x}_{\mathrm{c}} \tag{6}$$

$$\mathrm{y}_{\mathrm{pos}} = \ \mathrm{y}_{\mathrm{cross}} - \mathrm{y}_{\mathrm{auto}} + \mathrm{y}_{\mathrm{c}} \tag{7}$$

where ($x_{pos,}$ y_{pos}) is the to-be-determined position of the pattern in the image plane, ($x_{auto,}$ y_{auto}) is the position of the template autocorrelation peaks and (x_{cross}, y_{cross}) is the position of the crosscorrelation peak. The position of the cross-correlation peak was estimated using a polynomial fit to the correlation peak. The center of the template ($x_{c,}$ y_c) and ($x_{auto,}$ y_{auto}) are normally constant and may be calculated off-line, while the cross-correlation peaks move with changes in the object.

3. AUTOMATIC ALIGNMENT ALGORITHMS

The alignment system in each NIF beam line contains 26 control loops that analyze high resolution beam and reference images. A number of beam image types require matched filtering to determine the object positions. One such set of corner-cube reflected pinhole images is shown in Fig. 1. Here, the image processing algorithm exploits a template correlation to determine the pinhole centers.

A variety of distortions can challenge position finding algorithms. Examples in figure 1 exhibit a wide variety of distortions such as illumination, shade, shape, and size. A weighted, or even a binary centroid, measurement [9] will be severely affected by beam non-uniformity, intensity gradient, beam elongation or diffraction effects. The purpose of the template shown in Fig. 2 is to find the center by matching the edge of the beam. Since the beam size varies, the algorithm must search over a range of radii to determine the best matched circle [10]. The center of the circle that yields the highest correlation is chosen as the position of the pinhole image. While this template works for the majority of the beam images, a more accurate template was recently determined to represent beam images [11] that have minimal distortion.

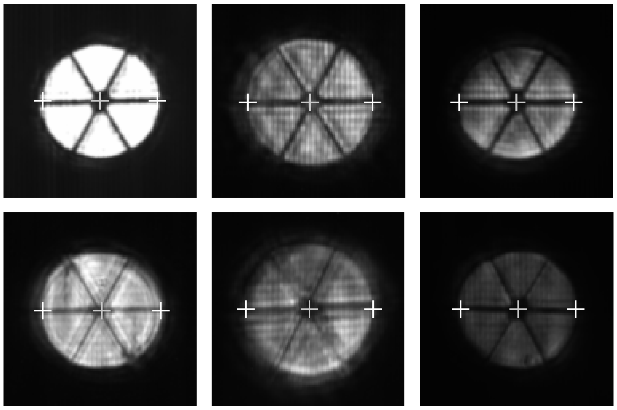

Fig. 1. A set of corner-cube reflected pinhole images of various image qualities

In the example, the radius of the template shown in Fig. 2 was varied from 33 to 42 pixels. The correlation peak at various radii is plotted as shown in Fig. 3. The peak reaches its maximum between a radius of 35 and 37. A second order polynomial fit [3] through the correlation plane provides the x and y position of the correlation peak, from which Eqs. 6-7 are used to find the center location.

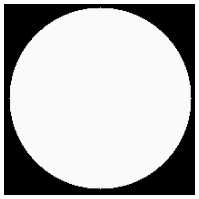

Fig. 2. Image of the template used for pinhole images

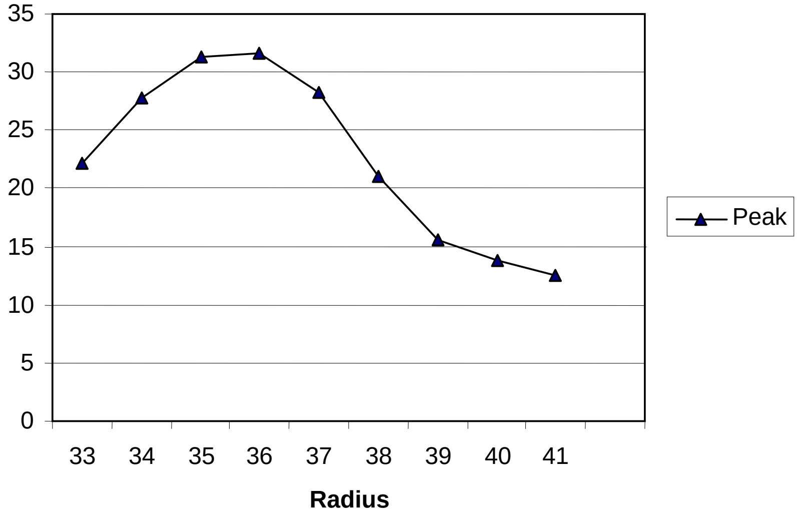

Fig. 3. Correlation peak versus template radius

In another application, where the same template is used, the pinhole images are shown in Fig. 4. Whereas the small pinholes vary from 32 to 45 pixels, these pinhole radii vary from 60 to 250 pixels. In order to reduce the processing time, instead of searching the whole range from 65 to 250 pixels, a measurement process is carried out to estimate the range to a smaller interval of 10 pixels [10].

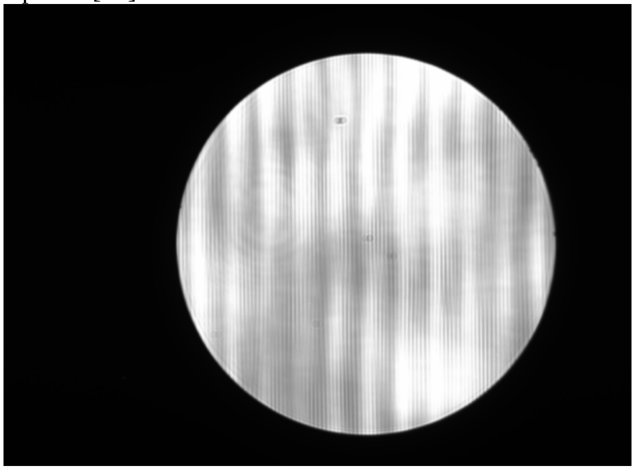

Fig. 4. Image with 160 pixel radius

In some alignment beam images, two types of fiducials (circles and squares) [12] are used to indicate the beam position and the alignment reference location. The diameter of the circle is similar to the side of the square resulting in correlation peak values that may be hard to discriminate.

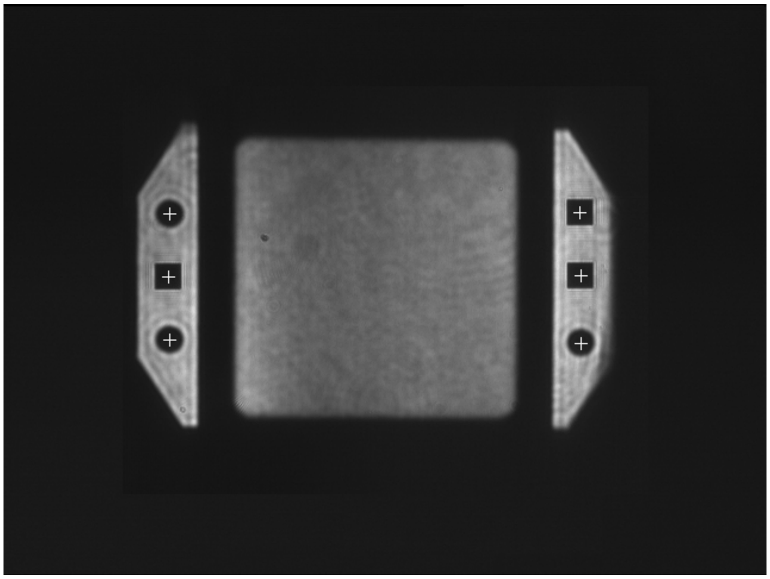

Fig. 5. Two classes of fiducial patterns with positions identified

To enhance the discrimination, and hence the detection accuracy, of the to-be-detected objects, features such as object edges are used as shown in Fig. 6. Instead of using circle templates, the circle edge is used for the filters. The resulting correlation cross-section from the right side of the wings is shown in Fig. 7. Note from Fig. 7 that the circle autocorrelation is higher than crosscorrelation with the squares exhibiting a 2:1 discrimination between the two. Based on the normalized autocorrelation value, a dynamic threshold (as a percentage of the maximum peak) can be selected to reject the non-circles correlations. After selecting the circles, the image is correlated with a second template consisting of a square mask. Now using Eqs. 6-7, the position of the objects can be found from the position of the crosscorrelation peak, the autocorrelation peak, and the template.

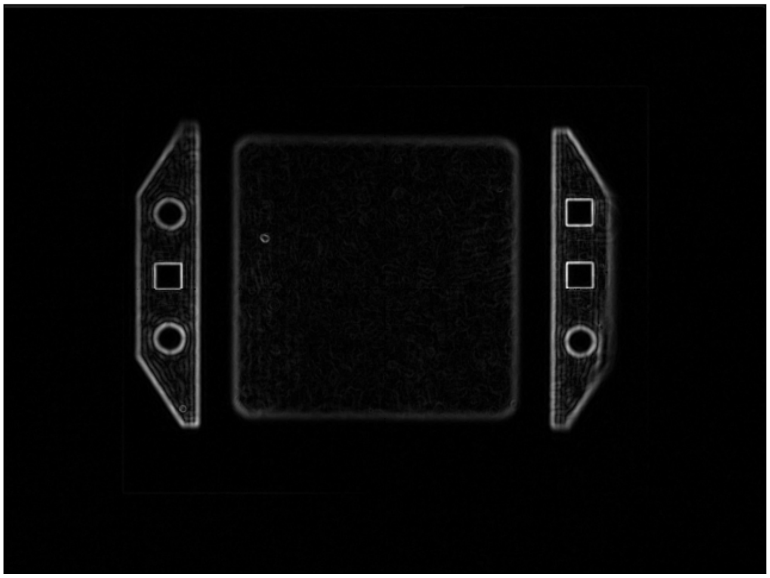

Fig. 6. The edge of the image in Fig. 5

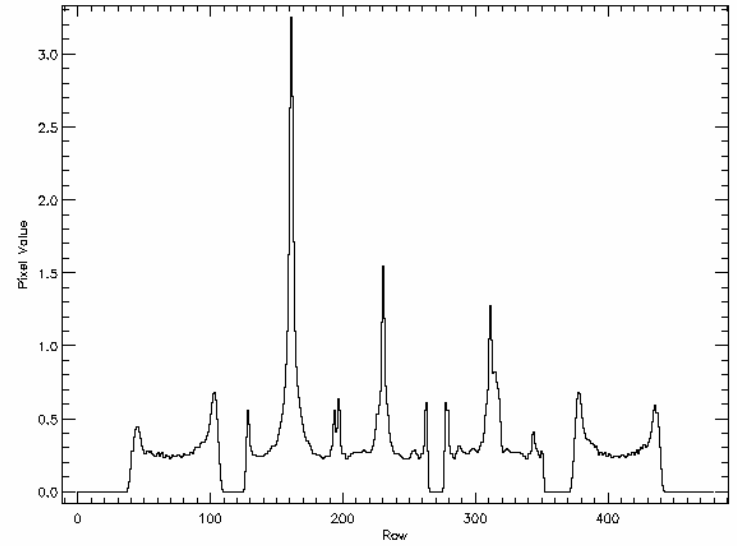

Fig. 7. The correlation with circle of the image in Fig. 5 (the cross-section through the right wing shown)

In all these applications, the basic operation performed is a matched filtering via Eq. 5.

5. FPGA Acceleration of Image Correlation

The most computationally intensive portion of the image processing is the 2-D image correlation. Thus to shorten the alignment time, one can reduce the image processing time. For continuous high performance alignment operation such as may be required in a laser fusion power plant, faster methods of beam alignment will be necessary. One advantage of these computations is a significant amount of parallelism, thus enabling hardware acceleration. We evaluated the potential of hardware acceleration by implementing the correlation computations on an FPGA. The test system utilized was a Cray XD1 with 864 2 GHz AMD Opteron processing cores and 144 Xilinx Virtex II Pro FPGAs. In this system, only one FPGA and AMD processor was utilized for the testing. The AMD processor sends the input image to be processed to the FPGA and receives back the location and peak value in the correlation output. A more practical approach for FPGA acceleration would be to utilize an FPGA accelerator card in a desktop computing system (such cards average about $2500 per FPGA at present).

5.1 Hardware design

Fig. 8 presents a system overview of the FPGA implementation. Input data and intermediate values are stored in buffers (shown as the shaded boxes). These are on-chip memories on the FPGA. The inputs to the system, *f* and *g*, represent the filter and source image in Eqs. 1 and 2, respectively. Up to 16 filters can be loaded into the FPGA (in the buffers labeled *f0* to *f15*) and applied to each image. The two-dimensional FFTs in Eqs. 1 and 2 are performed using two consecutive one-dimensional FFTs. Similarly, the inverse FFT in Eq. 5 is implemented with two one-dimensional forward FFTs. The FFT units were built using Xilinx-supplied library components.

To enable high-throughput computation, the system is pipelined into four phases as shown in Fig. 8. Each phase works on a particular image–filter combination. Since the same set of filters is used for each image, the filters are preloaded in on-chip buffers. This allows high speed access to the filters that accelerated the system performance. Note the time to load each filter onto the FPGA is longer than the pipeline phase computation time. Since each phase requires multiple cycles to compute (about 22.4k cycles), two buffers are needed between consecutive phases. For example, in Fig. 8, the upper buffer (F_1) between phases 1 and 2 holds the output being generated by phase 1. The lower buffer (F_2) holds the completed output previously generated by phase 1, for use by phase 2. Switches pipe data to the appropriate buffers.

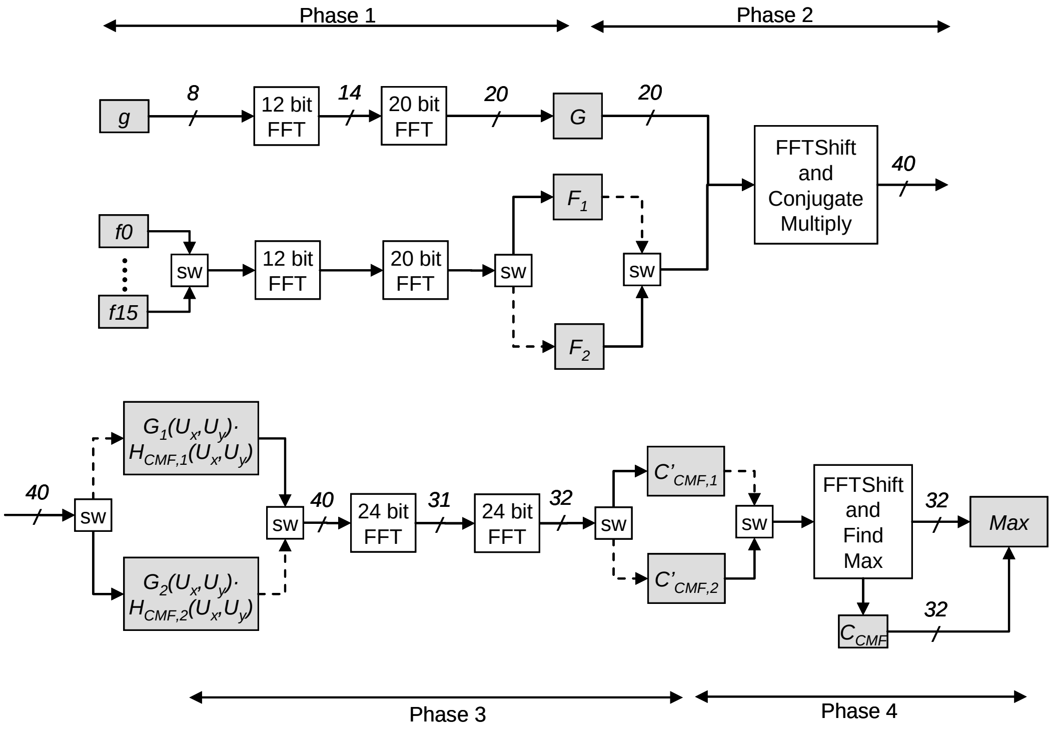

Fig. 8: The block diagram of the FPGA operations. The boxes labeled "sw" are switches.

The four phases in the architecture perform the following functions:

Phase 1: Complex Fourier transform represented by Eqs. 1 and 2 are computed. These two computations can be carried out in parallel. The inputs to this phase are unsigned 8 bit values. Since an 8 bit FFT unit would treat the inputs as signed values, a larger bit width FFT unit is needed. Therefore a 12 bit FFT unit is used for the first stage. As the maximum output value is 14 bit, a 20 bit FFT unit is used for the second stage. The second stage 20 bit FFT outputs are stored in buffers labeled *F* and *G*.

Phase 2: Part of Eq. 5 is evaluated. Here the output of Eq. 1 is conjugated and multiplied by the output of Eq. 2. An FFT shift operation is executed in parallel with the multiplication in order to center the image. The 40 bit output is stored in a buffer.

Phase 3: The inverse FFT in Eq. 5 is evaluated. Since the inverse FFT is implemented with two 24-bit forward FFT units, they use only the most significant 24 bits of the inputs. This introduces round-off error as the computations take place in the integer domain.

Phase 4: Location of the peak in the output of Eq. 5 (C_{CMF}) is determined. The coordinates and amplitude of the peak along with the amplitude of the four surrounding locations are stored and returned to the processor.

5.2 Hardware Performance

The system above was implemented on a Xilinx Virtex II Pro FPGA (part number XCVP50) on a Cray XD1. The FPGA synthesized system ran at 158 MHz. The logic utilization was 74% while the block RAM utilization was 93%. The algorithm was also run on a 3GHz Pentium 4 processor based desktop computer using Matlab version 6. We tested both systems with 64x64 images and 16 filters per image. The overall runtime of the FPGA system to process an image through 16 filters was about 3.32 ms, while the desktop system required 66.40 ms. This is equivalent to a speedup of 20 times. Newer generation FPGAs with larger resources and higher clock speeds would allow multiple pipelines to analyze more images in parallel, thus resulting in greater speedups. Fig. 9 shows the output error between the Matlab and the FPGA implementations for both auto and cross correlation examples. Only the peak and it neighboring four locations are shown. The average absolute error for these values is 0.33%.

Matlab	**FPGA**	**Error (%)**
4.9806	4.989	0.17
4.9806	4.9452	-0.71
4.8208	4.822	0.02
4.7942	4.796	0.04
4.7409	4.748	0.15

(i) Cross correlation between a square and a circle

Matlab	**FPGA**	**Error (%)**
6.2324	6.198	-0.55
5.8862	5.885	-0.02
5.8862	5.889	0.05
5.753	5.784	0.54
5.753	5.784	0.54

(ii) Auto-correlation of a square

Matlab	**FPGA**	**Error (%)**
4.9806	4.948	-0.65
4.6344	4.635	0.01
4.6344	4.635	0.01
4.5278	4.5591	0.69
4.5278	4.561	0.73

(iii) Auto-correlation of a circle

Fig. 9: Output comparison between Matlab and FPGA implementations for the peak and surrounding four locations (output values are to be multiplied by 10^{10}).

6. SUMMARY

Automatic alignment of the NIF laser is dependent on computationally intensive image processing. One important component of the image processing is matched filtering. This paper describes an approach to speed up this computation using low cost parallel computing hardware. The results indicate a speedup of 20 using an FPGA over a 3 GHz Pentium 4 processor. Other applications that can benefit from the speed enhancement include associative recall of extremely large databases of medical or other images using a partial queue.

ACKNOWLEDGEMENT

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. Kenneth Rice acknowledges the summer student support at Lawrence Livermore Laboratory. Kenneth Rice and Tarek Taha acknowledge grants from the Air Force Research Laboratory (including the AFRL Information Directorate) and a National Science Foundation CAREER award. This work was also supported in part by a grant of computer time from the DOD High Performance Computing Modernization Program at the Naval Research Laboratory. Abdul Awwal acknowledges insightful comments provided by Paul Van Arsdall.

REFERENCES

1. E. Moses, et al., “The National Ignition Facility: Status and Plans for Laser Fusion and High-Energy-Density Experimental Studies”, *Fusion Science and Technology*, Vol. 43, p. 420, May 2003.
2. K.C. Wilhelmsen, A. A. S. Awwal, S. W. Ferguson, B. Horowitz, V. J. Miller Kamm, C. A. Reynolds, “Automatic Alignment System for the National Ignition Facility” presented in the international Conference on Accelerator and Large Experimental Physics Control Systems, Knoxville, TN Oct 14-20, 2007.
3. A. A. S. Awwal, Wilbert A. McClay, Walter S. Ferguson, James V. Candy, Thad Salmon, and Paul Wegner, “Detection and Tracking of the Back-Reflection of KDP Images in the presence or absence of a Phase mask,” Applied Optics, Vol. 45, pp. 3038-3048, May 2006.
4. J. V. Candy, W. A. McClay, A. A. S. Awwal, and S. W. Ferguson, “Optimal position estimation for the automatic alignment of a high-energy laser,” Journal of Optical Society of America A, Vol. 22, pp. 1348-1356,2005.
5. VanderLugt, “Signal Detection by Complex Spatial Filtering,” IEEE Trans. Inf. Theory IT-10, 139-145, 1964.
6. J. L. Horner and J. Leger, “Pattern recognition with binary phase-only filters,” *Applied Optics*, Vol. 24, pp. 609-611, 1985.
7. A. A. S. Awwal, M. A. Karim, and S. R. Jahan, “Improved Correlation Discrimination Using an Amplitude-modulated Phase-only Filter,” *Applied Optics*, Vol. 29, pp. 233-236, 1990.
8. M. A. Karim and A. A. S. Awwal, Optical Computing: An Introduction, John Wiley, New York, NY, 1992.
9. W. A. McClay III, A. A. S. Awwal, H. E. Jones, K. C. Wilhelmsen, W. Ferguson, M. McGee, M. G. Miller, “Evaluation of laser-based alignment algorithms under additive random and diffraction noise,” in *Photonic Devices and Algorithms for Computing VI*, edited by K. Iftekharuddin and A. A. S. Awwal,, Proc. of SPIE 5556 (SPIE Bellingham, WA, 2004), pp. 243-248.
10. A. A. S. Awwal, “Automatic identification of templates in matched filtering,” in *Photonic Devices and Algorithms for Computing VI*, edited by K. Iftekharuddin and A. A. S. Awwal,, Proc. of SPIE 5556 (SPIE Bellingham, WA, 2004), pp. 102-109.
11. A. A. S. Awwal, K. Rice, R. Leach and T. Taha, “Higher accuracy template for corner cube reflected image,” to be presented at the Optics and Photonics for Information processing(II), San Diego, August CA 2008
12. A. A. S. Awwal, “Multi-object feature detection and error correction for NIF automatic optical alignment” in *Photonic Devices and Algorithms for Computing VIII*, edited by K. Iftekharuddin and A. A. S. Awwal,, Proc. of SPIE Vol. 6310 (SPIE Bellingham, WA, 2006), 63100Q, 2006.