GPU ACCELERATION FOR AIRPHOTO PROCESS AUTOMATION

David Piekny, Product Manager
PCI Geomatics
50 West Wilmot St.
Richmond Hill, ON
Canada, L4B 1M5
pieknyr@pcigeomatics.com

ABSTRACT

PCI Geomatics, in response to needs in the airphoto production market, have developed a high-performance image processing system called the GeoImaging Accelerator (GXL), based on multi-core and Graphical Processing Unit (GPU) microchip advances. This system was used as a template for an ortho-mosaic extension to the UltraMap processing chain for UltraCam imagery. The workflow extension relies on six key elements to achieve high-performance, high quality throughput: 1) System automation; 2) modular components; 3) distributed processing; 4) processor optimization; 5) system sustainability; and 6) risk mitigation. Descriptions are additionally given for the hardware and architectures used for both the development and test machines. Results of an optimized ortho process are given both independently and as part of a non-optimized mosaic workflow, with an orthorectification speed improvement of 18x (9 seconds vs. over 2 minutes) for UltraCam imagery. Further improvements were identified for data management and mosaic optimization, and an overview of the distributed, cloud-computing architecture inherent to both the GXL and UltraMap extension is included.

INTRODUCTION

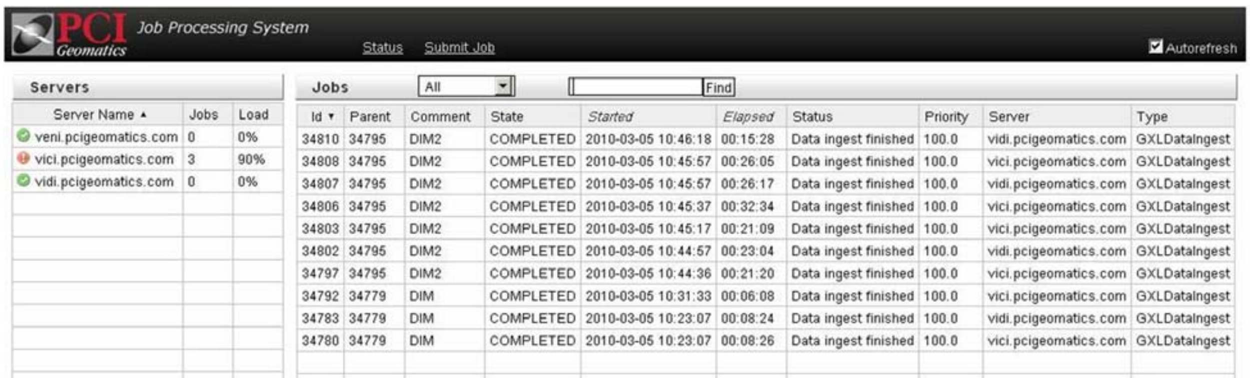
The volume, availability and diversity of digital imagery have been increasing steadily over the past five years, due mainly to launches of high-resolution optical satellites and sales of digital airphoto cameras. Concurrently, larger projects encompassing longer processing chains are becoming increasingly frequent, and the trend for precision moves towards smaller pixels and more finely tuned bandwidths. While this creates a wide and varied selection of possible products for those who need access to spatial information, it remains a daunting for those who have to process it, whether to accommodate shrinking timelines or the creation of ever more diverse products in an effort to grow or remain competitive. The result of this massive increase in data volume is a requirement for processing methods to handle them efficiently and economically. In order to address these requirements, six key areas were identified for development and commercialization of a system capable of handling the multi-terabyte projects and high-throughputs expected from the airphoto market: 1) System automation; 2) modular components; 3) distributed processing; 4) processor optimization; 5) system sustainability; and 6) risk mitigation. The result, named the GeoImaging Accelerator (GXL), is an integrated hardware and software system capable of quickly and accurately processing large volumes of imagery and optimizing advances in modern, multi-core processor architecture, as well as NVIDIA's Graphical Processing Units (GPUs). Results to be discussed herein will focus on the use of UltraCam imagery and the extension of the UltraMap pre-processing chain to include automated orthorectification and mosaicking.

SYSTEM AUTOMATION

Production increases are frequently found through automation and one of its benefits, aptly demonstrated by Henry Ford, is consistency. Whether an orthophoto or odometer, automation provides several important benefits, including, but not limited to the consistency, such as the following: Increased levels of production (leading to increased revenues), a higher level of quality assurance, reduction in process management and operational costs, and reduction in training costs. Further, profitability generally increases in proportion to output. Another benefit of automation is more effective use of resources; both skilled operators and technological capital.

Automating a manual process allows knowledge workers to transfer their skills from menial tasks to value creation (increased productivity and profitability) by managing quality assurance or product creation.

Automation also ensures the effective use of processing hardware. Batch processing and limited automation are included in many image processing packages - the GeoImaging Accelerator is no exception, incorporating a Job Processing System (JPS) to manage multiple users, job classes, definitions and scheduling. The JPS passes parameters from the first to subsequent task as discussed below in the Modular Components section. Master job definitions are capable of spawning child jobs both for parallel processing of multiple, identical jobs (such as orthorectification of an airphoto project), and for serial processing of singular tasks (such as mosaic creation, automated seamline selection and colour-balancing). This type system is easily adapted to existing pre-processing chains such as UltraMap, as the input parameters can be linked to earlier tasks, pre-selected or altogether hidden if required, while retaining the ability to launch a QA/QC interface for manual inspection and editing of seamlines and colour-balancing. Searching, prioritization and job sorting are all features of the JPS, resulting in efficient job management (see Figure 1, below).



Server Name	Jobs	Load
veni.pcigeomatics.com	0	0%
vici.pcigeomatics.com	3	90%
vidi.pcigeomatics.com	0	0%

Id	Parent	Comment	State	Started	Elapsed	Status	Priority	Server	Type
34810	34795	DIM2	COMPLETED	2010-03-05 10:46:18	00:15:28	Data ingest finished	100.0	vidi.pcigeomatics.com	GXLDataIngest
34808	34795	DIM2	COMPLETED	2010-03-05 10:45:57	00:26:05	Data ingest finished	100.0	vici.pcigeomatics.com	GXLDataIngest
34807	34795	DIM2	COMPLETED	2010-03-05 10:45:57	00:26:17	Data ingest finished	100.0	vidi.pcigeomatics.com	GXLDataIngest
34806	34795	DIM2	COMPLETED	2010-03-05 10:45:37	00:32:34	Data ingest finished	100.0	vici.pcigeomatics.com	GXLDataIngest
34803	34795	DIM2	COMPLETED	2010-03-05 10:45:17	00:21:09	Data ingest finished	100.0	vici.pcigeomatics.com	GXLDataIngest
34802	34795	DIM2	COMPLETED	2010-03-05 10:44:57	00:23:04	Data ingest finished	100.0	vidi.pcigeomatics.com	GXLDataIngest
34797	34795	DIM2	COMPLETED	2010-03-05 10:44:36	00:21:20	Data ingest finished	100.0	vici.pcigeomatics.com	GXLDataIngest
34792	34779	DIM	COMPLETED	2010-03-05 10:31:33	00:06:08	Data ingest finished	100.0	vici.pcigeomatics.com	GXLDataIngest
34783	34779	DIM	COMPLETED	2010-03-05 10:23:07	00:08:24	Data ingest finished	100.0	vidi.pcigeomatics.com	GXLDataIngest
34780	34779	DIM	COMPLETED	2010-03-05 10:23:07	00:08:26	Data ingest finished	100.0	vici.pcigeomatics.com	GXLDataIngest

Figure 1. Screenshot of the Job Processing System (JPS), showing a set of typical jobs and parameters available to an operations manager.

MODULAR COMPONENTS

Scalability is essential for an organization to grow, either through new product creation or an increase in legacy business. What may work for one airborne sensor may not apply to another (e.g. frame vs. scanline), but all digital frame cameras produce high-resolution colour images and are usually subject to similar, if not identical processes for orthorectification and mosaic creation. Instead of a single, monolithic, stovepipe process, modules may be easily arranged and connected into logical workflows that are essentially Off-The-Shelf (OTS) products. An OTS approach reduces implementation time and increases the flexibility of any image processing system. The ortho-mosaic workflow used with UltraCam imagery is shown in Figure 2 (below). Additional workflows for pansharp and stereo-DEM extraction were also created, but not incorporated into an extension of UltraMap capability.

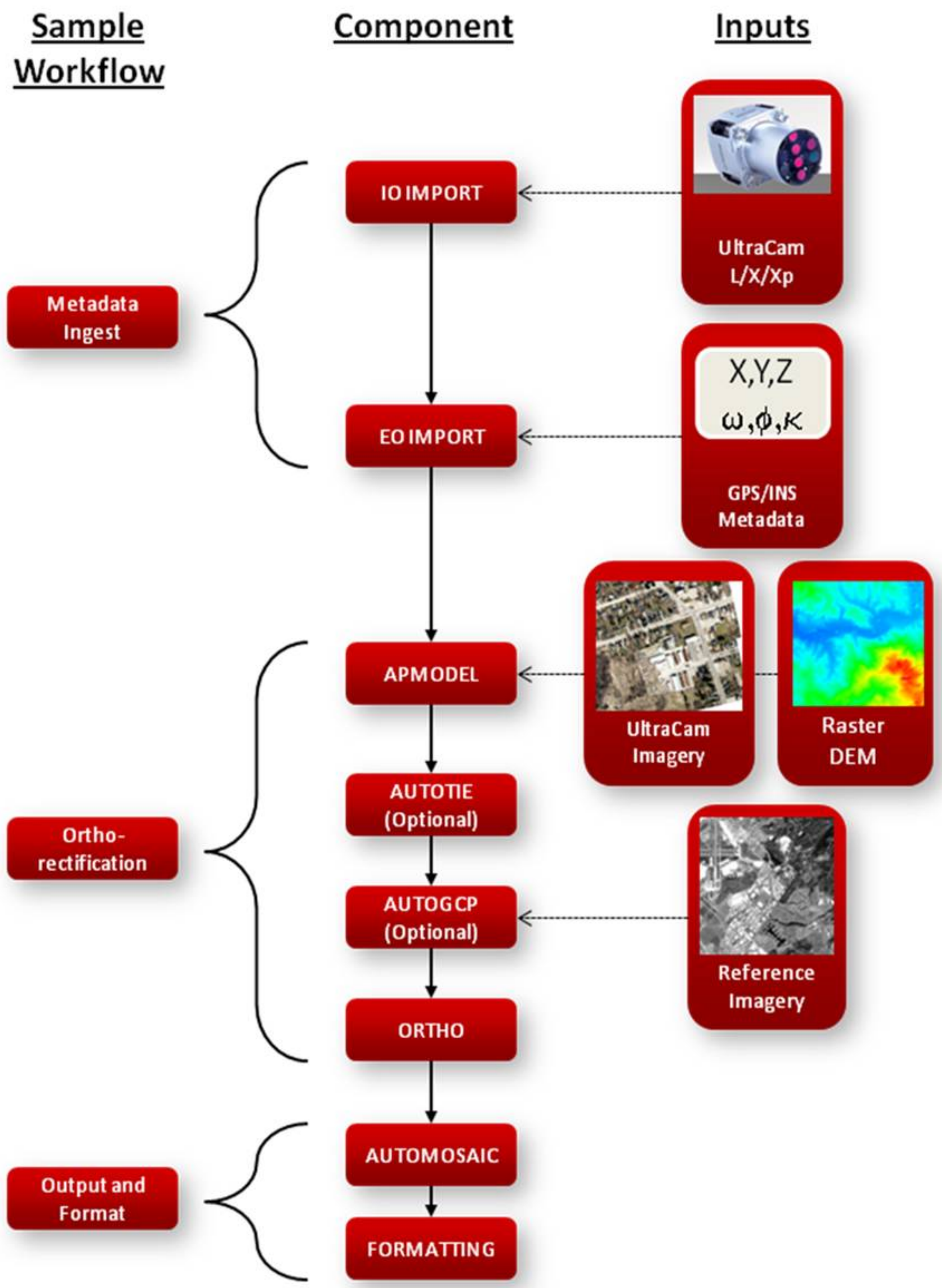


Figure 2. Modular workflow showing extension of UltraMap pre-processing into ortho-mosaic creation. Manual QA/QC can be done before or after any module.

As mentioned previously, the workflows can be fully automated, but allow breakpoints for QA/QC and process control. Image processing jobs can be chained together, run with multiple parameter sets, and components re-used to reduce migration and update costs as well. The modules and workflows are managed by a Job Processing System, which directly handles both data and metadata ingest and output formatting modules, as such, it is configurable to handle the varying data and product requirements expected as new imagery becomes available. The individual modules are described below:

IO IMPORT: Camera information, focal length, chip size, lens distortions (if applicable), etc.
EO IMPORT: GPS/INS information (XYZ/ωφκ), model calculations in Bingo format, etc.
AP MODEL: With a DEM, optimized code for GPU block model calculation for thousands of images
AUTOTIE: Optional image-to-image refinement based on overlap along or between flightlines
AUTOGCP: Optional image-to-image refinement between airphoto and reference mosaic
ORTHO: With a DEM, optimized code for GPU ortho-processing of UltraCam, etc. (at 1:1 sampling)
AUTOMOSAIC: Including preview generation, automated seamline selection and colour-balancing.
FORMATTING: Tiling, clipping, reprojecting, reformatting, etc.

Additional modules can extend this workflow further, for instance semi-automate fiducial collection for scanned airphotos, or linescan sensor input, while existing modules can be easily upgraded to include new functionality without altering the overall workflow or job management. The key module advances due to GPU processing include large block bundle adjustment on the order of thousands of images and tens of thousands of points, as well as fast orthorectification for each pixel with no interpolation or resampling.

DISTRIBUTED PROCESSING

Just as OTS modules provide flexibility and reusability, so does a distributed processing environment in which any number of computers at various locations can be managed and controlled in a coordinated fashion from a simple interface. This is critical as greater throughput is only of value if the increased number of projects can be effectively managed. The capabilities of the Job Processing System (JPS) (discussed previously), include load balancing and node availability for new tasks. Job classes are given not only a priority, but also a resource flag indicating relative ease of processing. For example, UltraCam orthorectification can be assigned to specific machines for data-transfer optimization or released into the greater resource pool for the first available processing core (CPU or GPU). The JPS overhead is minimal and this component is installed on any node, including laptop, desktop and rack configurations, with or without Graphical Processing Units (GPUs). The result is ease of scaling to enterprise processing levels by taking advantage of multi-core and GPU-based processing, where each node reports the availability of each processing core contained within. This two-fold approach allows management of a cloud of physical nodes with diverse specifications, as it handles both inter-node parallel processing and intra-node multi-core threading.

PROCESSOR OPTIMIZATION

Beyond a distributed environment and node management, further speed gains can be found in processing-core optimization, specifically GPU-based processing. Using nVidia Tesla microchips, which are dedicated to performing graphical operations, the speed of tasks such as orthorectification can be increased tenfold or more. Graphical Processing Units (GPUs) are uniquely suited to complex mathematical transformations with greater speed and precision than traditional CPUs, which are typically optimized for memory management of multiple, disparate tasks. For instance, producing an ortho from a 330MB UltraCam scene takes less than ten seconds using a combination of GPU and multi-core processing. The advantages of this acceleration are of course multiplied as the hardware scales up.

During development of the ORTHO module (see above), numerous techniques were used to increase the processing speed, and the achieved performance figures reflect the combination of all of these things. The purpose of this test was to investigate how much impact each different optimization had on the final result.

In general, the sequence of algorithmic improvements proceeded as follows:

1. The code was rewritten so that the same grid spacing is not required for the map transform and math model transform.
2. The raster I/O library was improved.
3. To minimize disk contention, different hard drives were used for input and output data.
4. The code was compiled in multithreaded mode, enabling the interleaving of I/O and processing as well as loop parallelization
5. Most of the per-pixel processing steps (except math model calculations) were moved to the GPU
6. Math model calculations were moved to the GPU.

The tests yielded some interesting observations:

1. Removing any grid-spacing constraints allowed separation of sub-tasks in the AP MODEL module.
2. Improving the raster I/O library reduced the runtime, even in single-threaded mode. If there is no concern about running out of disk space, it is not necessary to initialize the full output file (i.e., write zeroes to the whole file) before processing, which saved time in the implementation. It is also very important to have a library that does not access the disk when it doesn't need to! For instance, if the image is stored in tiles and we wish to overwrite an entire tile, there is no need to first read it from the disk.
3. Multithreaded processing is helped by using different hard drives for input and output. This is because input and output operations happen simultaneously in multithreaded mode, potentially causing disk contention.
4. Even while processing entirely on the host, performance is helped enormously by intelligent use of OpenMP multithreading. For an average scene, processing time was reduced by over 90%.
5. Moving all of the operations possible onto the two GPUs further reduced processing time by greater than 10%. We found during implementation that it was best to combine several successive processing steps into one function so that memory copies to and from the GPUs were minimized; this made a difference of about 15% in runtimes with respect to the non-combined processing steps (i.e., datasets that were running at about 2.0 GB/min were improved to 2.3 GB/min).
6. It does help to have two GPUs rather than one, but the improvement in performance are hampered because the system is partially limited by I/O performance; some blocks of the image (not all) are delayed while waiting for I/O to finish. Having a faster or more processors does not help in that case.

SYSTEM SUSTAINABILITY

From interviews and vendor-experience, massive throughput is attractive to the airphoto market. Processing 50TBs weekly will become ever more common. Until then, hardware infrastructure must be flexible enough to accommodate new frameworks and upgrades as easily as it does new sensors and processes. This was recognized and taken into account for the GXL design. Any extension to UltraMap or other processing chains adopts standards of mainstream IT and hardware communities, for instance OpenMP for multithreading and CUDA for GPU optimization. The hardware and software components and architectures have guaranteed development roadmaps outside of product development and as these industries change, new technologies will be introduced to provide a smooth transition for geo-image processing. Like the modules discussed above, multi-core and GPU-based hardware is OTS, resulting in lower total cost of ownership and fast implementation. Both the computing architecture and interface to the GPUs, called CUDA (Compute Unified Device Architecture) has been adopted as an industry standard.

Development Environment and Tools

For development of the orthorectification software, openSuSE Linux 10.3 (64-bit) was installed on the workstation. All development was done using the Intel C++ compiler (version 10.1), using the OpenMP 2.5 libraries for multithreading and NVIDIA's CUDA SDK release 2.0 for programming the GPUs.

openSuSE Linux 10.3 (64-bit)
Intel C++ compiler (version 10.1)

OpenMP 2.5 libraries for multithreading
NVIDIA CUDA SDK release 2.0 for programming the GPUs

Development System Hardware

The system purchased was an HP Blackbird 002 LCi machine, pictured in Figure 3. It features an Intel® Core2™ Extreme Quad-Core 3.0GHz QX9650 CPU, 8 GB RAM, one 7200 RPM and two 10,000 RPM SATA hard drives, and two NVIDIA GeForce GTX-280 GPUs with 1GB of GDDR3 SDRAM. The total cost of the system was $7600 USD.

HP Blackbird 002 LCi
1x Intel® Core2™ Extreme Quad-Core 3.0GHz QX9650 CPU
2x 4GB RAM
1x 7200 RPM SATA HDD
2x 10,000 RPM SATA HDD
2x NVIDIA GeForce GTX-280 GPU with 1GB of GDDR3 SDRAM

Figure 3. Orthorectification development workstation.

Because the ortho-mosaic extension of the UltraMap processing chain will be installed on either and pre-existing or new-and-compatible hardware configurations, the original GXL design philosophy of sustainability will apply for the commercialization of this and future workflows.

RISK MITIGATION

The final requirement for any system to be packaged with workflows from other sources is mitigation of risk from both a technical and business perspective. This is true for both the extension of the UltraMap capability and additional workflows for other uses and sensors, especially given the burgeoning amount, type and complexity of geo-image availability the upcoming years. Meeting increasing processing demands requires a stable foundation for predictable growth, so any derivative of the GeoImaging Accelerator must address these demands. As previously discussed, the foundation is built foremost on a combination of automated processing, which increases consistency of product quality and reduced operational costs, and speed, meaning faster product availability. Additionally, modular components can easily be built and configured without the need for complex development or extensive training. A Job Processing System allows distributed processing and load balancing for large batch processes of orthorectification and mosaic creation. Distributed processing allows processing power that is flexible and cost-effective, both of which are absolute requirements. Finally, multi-threading and GPU processing yield huge speed increases for geo-image processing and allow all projects to be completed more quickly, whether in a desktop, rack or distributed environment. While this addresses the technical perspective, mitigation of risk from a business perspective can only be achieved from the company responsible.

For over twenty-five years PCI has provided expertise to the geomatics industry as a knowledge based company with broad and deep IT experience. Profitability, global diversity and active industry participation at all levels provide the stability required to support long-term growth and development of the high-performance GXL system, and extensions to UltraMap processing.

PROCESSING RESULTS

Test Hardware

Operating system:	RED HAT Enterprise Linux 5, Server
GPU System:	NVIDIA Tesla GPU S1070-500 Server System – Barebone 1U BB Dual&Quad-Core Xeon (5000/5100/5300 series) Intel Xeon E5410 2.33G Quad-Core Processor 667MHx ECC FB-DIMM x42 300G 10000rpm Velociraptor
Processing Servers (2):	Quad Core Xeon E5420 Processor2x6Mb Cache, 2.5 GHz, 1333MHzFSB 4GB 667NHz (4x1GB), Single Ranked Fully Buffered DIMMs
SAN data drive:	Dell AX4-5F DP (Dual Storage Processor) 12 1T 7200 RPM SATA 3Gpbs 3.5 in HotPlug Hard Drive

Test Dataset

Source:	UltraCam Imagery
Filesize:	330Mb
Image Size:	11500 pixels, 7500 lines
Resolution:	0.1m
Bit-Depth:	8-bit
Channels:	3 (RGB)
Number:	267 Images

Process Overview

Four modules were used in the generation of the mosaic tiles for the airphoto UltraCam data. The following are the modules used, including the process used within those modules:

Airphoto Data Ingest Tasks

link: Creates a live link to a non-PCIDSK imagery file of a supported format
crproj: Creates an OrthoEngine-style ASCII project file from a list of input images or GCP segments
camimport: Imports camera calibration parameters from an XML file into an OrthoEngine project file
eoimport: Imports the exterior orientation data from a text file into an OrthoEngine project
apmodel: Performs a bundle adjustment on the camera position and orientation values associated with a set of images

Orthorectification Tasks

ortho: Performs orthorectification or geometric correction on raw image data where a math model exists

Mosaic Preparation Tasks

mosprep: Performs all of the necessary pre-processing of the scenes that are to be mosaicked. It computes reduced resolution files, determines hotspot or adaptive correction coefficients, performs tonal balancing, and computes cutlines

Mosaic Creation Tasks

mosdef: Generates a mosaic definition XML file for the given source image list and also output a tile definition in the form of a vector polygon layer
mosrun: Generates a mosaic tile or tiles based on the information stored in the scene list and mosaic definition xml file
scale: Maps the grey levels of input image to the grey levels of output image
fexport: Exports a window of selected data layers in a PCIDSK file or GDB supported file format
imerge: Automatically merges a set of geocoded images into a single image

Performance Metrics

The baseline test results were gained by running the original code on the test hardware – no optimization, multi-threading or GPU processing was used. The GPU timings reflect the results of the optimization described in this paper. At the time of writing, the mosaic tasks had not yet been optimized.

Table 1. Performance Metrics (h:mm:ss)

Process	Average time per scene - baseline	Total processing time – baseline	Average time per scene - GPU	Total processing time – GPU
Data Ingest:	0:00:11	0:50:19	0:01:01	4:30:00
Orthorectification:	0:02:38	11:46:12	0:00:09	0:40:02
Mosaic Preparation:	0:49:51	0:49:51	0:49:51	0:49:51
Mosaic Generation:	0:08:02	19:10:15	0:08:02	19:10:15
Total Elapsed Time:	1:00:42	32:36:37	0:59:03	24:10:08

Performance Summary

Throughput Metrics – Full ortho-mosaic workflow

Overall Processing Time on GPU: 0.25x Speed Increase (no ingest or mosaic optimization)
Ortho Processing Time on GPU: 18x Speed Increase

Throughput Metrics – Orthorectification and Data Ingest

Orthorectification GB/Hour: 17.0 GB/Hour
Orthorectification GB/Day: 407 GB/Day
Orthorectification Area km²/Day: 1065 km²/Day
Orthorectification Scenes/Day: 9600 Scenes/Day

The throughput tests were carried out to answer the questions of how much UltraCam imagery can be processed in a given period of time. The key points identified are that orthorectification was increased eighteen-fold over non-optimized code running on the same high-performance hardware. It also indicated the overall importance of data management and mosaic improvements in the context of total workflow management. These recommendations have been addressed during the development of the ortho-mosaic extension to the UltraMap processing, with preliminary results suggesting an overall workflow speed increase of 2.5x, up from 0.25x. This represents 12 hours faster completion of the project described in Table 1.