The Ortho-Mosaic Process in RemoteView

Overwatch has developed and deployed geospatial exploitation software tools in RemoteView capable of very high sustained throughput rates. Originally designed to enable smooth, jitter-free roaming in high-resolution displays, this same RemoteView architecture can sustain very fast creation of ortho-rectified mosaics. File size does not adversely affect the throughput rate. Even terabyte-class image files do not diminish the throughput rates. This capability enables distributed, desktop-based workflow scenarios that can compete with more expensive, centralized processing schemes.

1. What is Ortho-Mosaicking?

Ortho-Mosaicking is the combination of two processes: **Ortho-rectification** and **Mosaicking**.

**Ortho-rectification** is the process of correcting imagery for distortion using elevation data and camera model information so that the scale variation corresponds to a map projection throughout the image. Image distortion can arise from a number of sources including terrain or feature elevation, collection geometry, and from the sensor itself.

**Figure 1** shows a very simplified illustration of the distortion that can arise from elevation. The vertical arrow represents a vertical elevation feature. The sensor takes an image of the arrow and ground plane, and projects the image onto the image plane, which is perpendicular to the sensor field of view center ray. The tail of the arrow is projected to point B on the image plane, while the arrow head is projected to point A on the image plane.
Because the sensor captures the image at an angle to the ground plane, the image will appear distorted. **Figure 2** below illustrates this distortion. The arrow appears to be “layed over” on its side. Points A and B illustrate how these points appear on the distorted image. The arrow also appears to be “shortened” from its true length.
Figure 3 illustrates how the image should look if the sensor captured every point in the image directly from overhead every point. If the sensor truly collected every point from a perpendicular collection angle, points A and B would overlap. The arrow would appear as a point. This kind of transformation is known as "true" ortho-rectification.

Simply put, the process of "true" ortho-rectification is the transformation to get from Figure 2 to Figure 3.

Mosaicking is the process of taking two or more separate images and "stitching" them together into a single image. Figure 4 illustrates an image generated by a mosaic process over four images.
The **ortho-mosaic process**, logically, combines these two processes. It creates a single image from many images, and corrects the resulting mosaic for distortions.

2. **Why do I need it?**

Users today often need to monitor broad areas and bring other georeferenced data, such as GIS layers, into their analytical environment. Most users are not experts in photogrammetry or pixel manipulation. Their primary skills are in analyzing features and their timelines, and in quickly drawing accurate conclusions. Since areas of interest often span multiple images, analysts need tools that will quickly and effortlessly stitch together multiple images to create the broad area coverage. They also need tools that will take the work out of placing GIS layers in their proper locations on the broad area images. Proper layer alignment is crucial for drawing accurate conclusions.

2.1. **Ortho-rectification**

As illustrated above, when a sensor collects imagery, especially in a non-vertical (oblique) collection geometry, there is some inherent distortion present in the collected imagery. This is especially true when the imaged area is hilly or mountainous. The collected imagery will have a number of distortions. One type of distortion is called layover, and is due to the higher elevation being closer to the sensor. **Figure 1** illustrates the layover phenomena.
Layover can also produce pronounced shifts in a feature’s location. Without ortho-rectification, layover will be present, and GIS layers will not line up. **Figure 5** below illustrates the shift in the position of a road vector that layover can produce. In this figure, the red line shows the true road position. The yellow line shows what can happen when a user extracts a road vector from a map or previously ortho-rectified image, and overlays that vector on a non-ortho-rectified image.

![Figure 5](image)

Without correcting for terrain effects, the vector will not line up with its true position. In this case, the offset is over 29 meters.

### 2.2. Mosaicking

Areas of interest often span significant geographic regions. This is especially true in the analysis of infrastructure such as communications, power and transportation. These infrastructure networks can cover significant portions of entire countries. Creating mosaics of these networks from multiple images can involve hundreds of gigabytes of image data. Analysts cannot afford to wait hours or even minutes for a tool to create a mosaic before the analysis can begin. Even modest areas of interest, such as a city, can require many gigabytes. For example, **Figure 6** below represents almost 6 GB of imagery from several commercial imaging satellites.
3. What ortho-mosaicking does RemoteView do?

In RemoteView, the process of removing sensor and terrain-induced distortions in an image is fully automated. Users are never required to enter sensor model information such as the type of sensor or the parameters of the sensor. This is handled automatically as shown in Figure 7.
When a user opens an image in RemoteView, the software automatically determines the best method of georeferencing the image based on its metadata, which is data in the header of the file that determines how it is laid out, as well as its other characteristics. This sensor model is "attached" to the image. Now, when the user moves the mouse, RemoteView "reads" the cursor location and, using the sensor model, converts the pixel coordinates to ground coordinates (Latitude, Longitude and Height or some other ground coordinate system such as MGRS, depending on what the user has selected in the viewer’s coordinate readout control). RemoteView also calculates statistical estimates of the accuracy of the ground coordinate when it is able to from the metadata. These values are called the "Circular Error 90%" and the "Linear Error 90%", abbreviated CE90 and LE90. These two numbers define the radius and the height of an imaginary cylinder located with it's base centered on the ground point. In a statistical sense, 90% of the true values of the ground point will fall within this cylinder. Of course, the other 10% of the time, the true values lie somewhere outside of this cylinder. The basis of this calculation depends on the imagery, but for images with Rational Polynomial Coefficients, the errors are based mostly on the ERR_RAND an ERR_BIAS tags in the metadata. These are interpreted as 1-sigma random and bias errors respectively, in the plane normal to the look vector, and not including terrain errors. Terrain errors are accounted for separately in the LE90 value on single image error propagation.
The results of geolocation, including accuracy are presented in the geo reporting control in the viewer as shown below.

![Geo Report](image)

This process of determining a ground point from pixel coordinates is one of the fundamental applications of Photogrammetry. The reverse process is also critical, that is, the computation of image coordinates from a given ground point. In fact, the mathematics of the sensor model make this reverse process more straightforward, since no external data (elevation) is required. A simple example of where this process is used is in the RemoteView Geo-Marker function shown below.

![RemoteView Geomarker Application](image)

In this function, the user enters ground point coordinates and RemoteView calculates the correct image pixel location which corresponds to that ground point and moves the user's view to center on that point. It will also optionally place a marker with a symbol and text at that point. To do this computation, RemoteView uses the sensor model in the reverse sense to that explained above.
for Geo Location. It converts from ground space to image space (pixel location) and moves the viewer to center on that point.

4. How is RemoteView ortho-mosaicking different?

As early as the mid 90’s RemoteView engineers have architected solutions that exploit converging COTS hardware technology and quality image metadata. This approach enables even novice users to benefit from performance increases offered by the most recent hardware advances that soon enter the commercial mainstream. It is a common sense approach to achieving increased performance at reasonable (COTS) costs.

That approach continues today, and gives RemoteView an ortho-mosaic capability unique among geospatial exploitation solutions.

4.1. Ease of Use

As already mentioned, RemoteView removes the burden of removing sensor and terrain-induced distortions from the user. Users are never required to enter sensor model information such as the type of sensor or the parameters of the sensor. This eliminates the need to understand the details of photogrammetry. Users with average computer skills can correct imagery for distortion so that the scale variation is the selected map projection throughout the image.

4.2. Leveraging 64-bit, Multi-core Advantages

The RemoteView architecture in conjunction with 64-bit, quad core, dual processing hardware offers new capabilities that enable very fast creation of ortho-mosaics of entire countries at high resolution.

A 64-bit OS running in a multi-core, multi-CPU environment has the necessary processing power and memory address space to address the processing demands that a mosaic of this scale would place on the processing system. This is simply not possible to do in a 32-bit environment.

The RemoteView architecture is able to take full advantage of a 64-bit, multi-CPU environment because of the focus the RemoteView engineers have placed historically on smooth, jitter-free display in dynamic imagery exploitation. In an interactive exploitation environment, data displayed dynamically on a screen must keep pace with a reasonably high refresh rate in order to guarantee fast response times and minimize user fatigue. For some applications, this refresh rate is as high as eighty-five new display frames every second, or about 12 msec per frame. Any pixel processing required to render pixels on the screen must finish within that time to avoid rendering “stalls” (also known as jitter). Jitter occurs when the processing required to render a single frame exceeds the interframe time interval. To avoid stalling the rendering pipeline, the pixel data must be queued or pre-staged into memory in a manner that accounts for I/O and processing latency. This queuing of pixel data requires significant amounts of memory, and large address spaces to support today’s higher-resolution monitors and more complex data sets.

By applying the power of a 64-bit OS running in a multi-core, multi-CPU environment to these demands, new capabilities become possible. Some examples of these new capabilities are listed below:
4.2.2. Soft Copy Search over TeraByte-class Mosaics

Soft Copy Search capabilities allow users to easily and quickly build large area mosaics from high resolution imagery and interactively search (roam, pan, zoom) and manipulate (contrast, brightness, DRA, etc.) the mosaic image. In order for the roam, pan and zoom to be truly interactive and responsive, the system must “queue up” all the necessary display tiles “next in line” to be pushed to the display. A 32-bit OS soon runs out of memory address space needed to keep track of all the display tiles ready to be pushed to the display in this interactive mode. Users in a 32-bit OS therefore experience a soft limit of about 500 GB on the size of an image mosaic. The soft limit in a 64-bit OS may be on the order of 100 TB. Thus, a 64-bit OS running in a multi-core, multi-CPU environment can create high-resolution mosaics of entire countries.

4.2.3. New “on-the-fly” processing possibilities

Besides the need to keep track of all display tiles ready to be pushed to the display, a truly interactive system also has the need to process the display pixels fast enough in order to keep up with the desired refresh rate. An additional advantage of “on-the-fly” processing is that it eliminates the need to store multiple processed copies of essentially the same data set. Only one copy of the native data needs storage space. “On-the-fly” processing creates all other requested processed products on demand.

Single CPU systems have offered limited ability to handle display pixel processing in an interactive manner. For example, RemoteView has taken advantage of this ability for more than ten years and offered users the ability to perform processing stages like pan-sharpening or ortho-rectification “on-the-fly”. A single core, single CPU system can achieve some performance gain by overlapping I/O and processing, but such a system must limit the number of processing stages they can support interactively before the CPU became the performance bottleneck.

Multiple CPUs and multiple cores can significantly reduce this bottleneck and improve performance. Multiple processing cores enable many of these stages to take place at the same time. This in turn means more processing stages can be ready for display in the same time in which previously only one stage could be ready. Some examples of concurrent processing stages that multiple CPU/multiple core systems may enable in an interactive (i.e., “on-the-fly”) manner are:

- Ortho-rectification terrain correction
- Mosaicking
- Re-projection
- Pan-sharpening
- Multispectral composition
- Edge detection
- Multispectral processing (i.e., NDVI, tasselcap transformations, etc.)

A 64-bit OS running in a multi-core, multi-CPU environment will support interactive mosaics of multi-stage products.

5. Summary

Overwatch has taken the RemoteView pixel processing architecture originally designed to enable smooth, jitter-free roaming in high-resolution displays and applied it to “on-the-fly” exploitation processes, including the Photogrammetric process of generating ortho-mosaicked images.
When installed on a 64-bit, multiple-CPU PC, this architecture can easily scale to terabyte-sized mosaics. With such a configuration it is now feasible to consider creating daily refreshed mosaics of entire countries in a desktop environment, realizing a considerable savings over more expensive, centralized processing schemes.