IMAGE 2012 Conference

AN OPERATIONALLY BASED VISION ASSESSMENT SIMULATOR FOR DOMES

John Archdeacon, Principal Engineer and IG Architect
Dell Services Federal Government/NASA Ames Research Center
Moffett Field, CA

James P. Gaska, Ph.D
711 HPW/USAFSAM/FECO
Wright Patterson AFB, OH

Samson Timoner, Ph.D
Scalable Display Technologies
Cambridge, MA

ABSTRACT

The Operational Based Vision Assessment (OBVA) simulator was designed and built by NASA and the United States Air Force (USAF) to provide the Air Force School of Aerospace Medicine (USAFSAM) with a scientific testing laboratory to study human vision and testing standards in an operationally relevant environment. This paper describes the general design objectives and implementation characteristics of the simulator visual system being created to meet these requirements.

A key design objective for the OBVA research simulator is to develop a real-time computer image generator (IG) and display subsystem that can display and update at 120 frames per second (design target), or at a minimum, 60 frames per second, with minimal transport delay using commercial off-the-shelf (COTS) technology.

There are three key parts of the OBVA simulator that are described in this paper:

> *i)   the real-time computer image generator,*
> *ii)   the various COTS technology used to construct the simulator, and*
> *iii)   the spherical dome display and real-time distortion correction subsystem*

We describe the various issues, possible COTS solutions, and remaining problem areas identified by NASA and the USAF while designing and building the simulator for future vision research. We also describe the critically important relationship of the physical display components including distortion correction for the dome consistent with an objective of minimizing latency in the system. The performance of the automatic calibration system used in the dome is also described. Various recommendations for possible future implementations shall also be discussed.

INTRODUCTION

The work described here is part of the U.S. Air Force sponsored Operational Based Vision Assessment (OBVA) program which has been tasked with developing a high fidelity flight simulation laboratory to determine the relationship between visual capabilities and performance in simulated operationally relevant tasks. This paper describes the general design objectives and implementation characteristics of the visual system being developed by NASA to meet this requirement.

This paper is focused largely on the underlying system architecture and does not attempt to fully explore or address all of technical issues associated with constructing an eye limiting visual system.

BACKGROUND

The most significant requirement of the OBVA simulator is that it should generate observer-limited imagery. That is, visually-dependent performance measured during simulated operational tasks must be limited by the observer's visual system and not the generated imagery or the display hardware. In the spatial domain, this requires a pixel pitch of about 0.5 arc-minutes [1] and imagery that matches the display sampling rate. The temporal sampling requirements are less well defined. Although, the human visual system is not sensitive to stationary temporal

---

modulation that exceeds approximately 60 Hz, simulation of moving objects requires higher frame rates in order to avoid visible motion artifacts due to sampling [2] or hold [3] properties.

With the possible exception of the high sampling requirements discussed above, the OBVA simulator design goals are similar to those of many flight simulators. Because of the need to tile multiple projectors on a spherical screen with high accuracy, we wanted to use tools that would automate and simplify the setup for required warping and blending operations. Further, quantifying and minimizing system latency for the various subsystems and devising methods to measure total system latency within the simulator became important objectives to properly characterize the system. Finally, in order to minimize cost and leverage commercially available solutions we attempted, whenever possible, to use commercial-off-the shelf (COTS) hardware and software components.

SIMULATOR COMPONENTS

The OBVA simulator at Wright-Patterson AFB (WPAFB) is depicted below in Figure 1. A dedicated computer room (i.e., the room with blue boxes) is used to house five 42" tall 19-inch racks containing all of the computers and 3D graphics hardware, host computer, and other rack-mount electronics to drive up to fifteen (15) projectors in the initial design configuration, although only nine (9) projectors will be supported in the initial deployment in mid-2012. The dome display surface, cockpit, and projectors will be installed in a high-bay room previously occupied by an Air Force centrifuge which has recently been removed and the room renovated for OBVA simulator use.
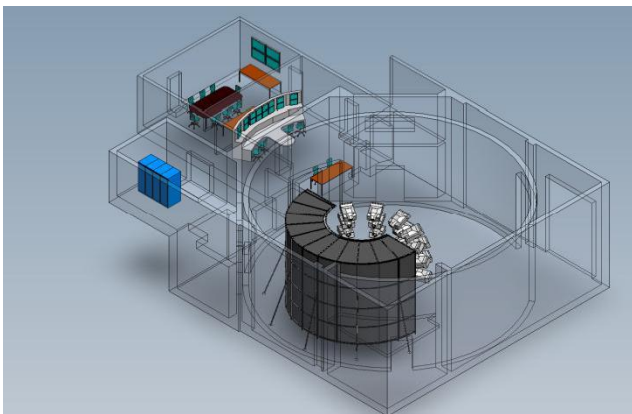

Figure 1 - The OBVA Simulator at WPAFB

Display Subsystem

Within the past several years, ultra-high resolution projectors and monitors (sometimes referred to as "quad-HD" or "4K" displays) have become commercially available (e.g., Sony, Barco, and JVC). These display devices typically have four dual-link DVI or four HDMI 1.4 inputs and provide 4x or higher pixel resolution than a standard HD display (1920x1080). These new high-end COTS display devices provide anywhere between 8.29 million pixels (QFHD with 3840x2160) and 9.83 million (4096x2400) pixels per display depending on the manufacturer and model.

For OBVA, the cockpit for the human observer involved in vision research will be placed in front of the projector subassembly as depicted below in Figure 2; the goal is to provide 20/10 visual acuity to the observer positioned at the design eyepoint location.
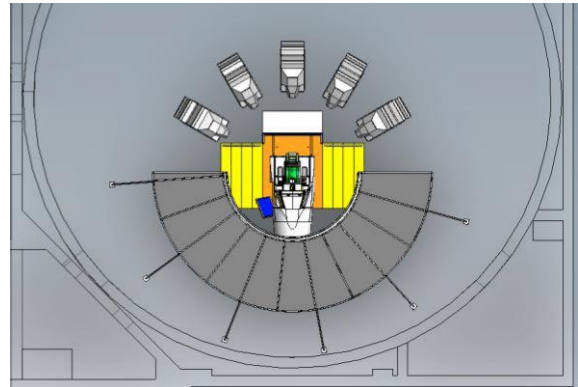

Figure 2 - Cockpit and Dome/Displays

Initially, the OBVA simulator at the NASA Ames Research Center was setup with only four 4K projectors since the USAF had not yet selected a final projector. Figure 3 shows the partial 4-meter dome subassembly used at the NASA Ames Research Center to test four Sony digital cinema projectors (two S110s and two T110 SXRD projectors).
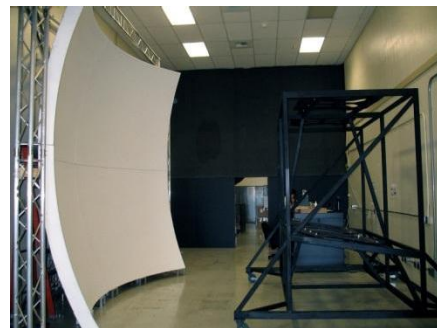

Figure 3 – Partial OBVA Dome Subassembly at NASA

When the simulator is setup at WPAFB in mid-2012, the IG will then be configured to drive nine (9) Barco Sim10 projectors (4096x2400 pixels each) arranged for projection on the four (4) meter spherical dome display surface. Growth has been designed in to support up to fifteen (15) quad-HD projectors with an approximate 160 degree horizontal field-of-view, with a future upper capacity planned for twenty-five (25) projectors.

With nine initial projectors, the human observer in the OBVA cockpit will be seeing 88.47 million pixels, believed to be the highest pixel resolution / density ever provided in a real-time simulator with planned eye limiting 20/10 visual acuity.

Target - 120 Hz Refresh Rates

Modern day quad-HD liquid crystal on silicon (LCoS) projectors generally provide very good luminance due to hold times that are approximately equal to the frame interval (~16 milliseconds for a 60 Hz frame rate). However, when an observer tracks a moving object, the retinal image corresponding to the on-screen image (which is stationary during the inter-frame hold time) is blurred due to eye movements. The magnitude of the "tracking blur" is proportional to the product of hold time and object velocity, and can be reduced using shutters or electronics at the expense of reduced luminance. A better solution is to increase the frame rate while maintaining a hold-time that is equal to the frame interval; this will reduce both sampling artifacts and tracking blur while maintaining luminance.

However, as of this writing, industry has not yet produced a quad-HD projector that supports a refresh rate of 120 Hz, so the OBVA program settled upon 60 Hz refresh rates until industry can provide at least 120 Hz or faster. Projectors that support 120 Hz refresh rates are now commercially available from companies such as Christie Digital where NASA tested a 1920x1080@120 Hz projector that significantly reduced motion artifacts [4]. However, many projectors (including the projector tested at 120 Hz) have limited luminance (often 2400 or fewer lumens) which is considerably less than the planned target of 6,000 lumens per projector or better.

The benefits of higher frame rate generated a design goal of 120Hz refresh rate capability for the OBVA image generator [4]. Instead of render times for OBVA that must be computed in less than 16.667 milliseconds (60 Hz), the design update rate targets were set at 120Hz (8.333 milliseconds) for OBVA in order to be ready for the future.

Image Generator (IG)

As with other major subsystems, the OBVA image generator was purposely designed to employ proven COTS technology in order to minimize technical risk and keep development, deployment and future maintenance costs to a minimum.

Planned Technology Insertion

The image generator was purposely designed to facilitate technology insertion. Thus, NASA and the USAF can plan to take full advantage of the market phenomenon where Graphics Processor Unit (GPU) performance is expected to continue to exceed Moore's Law [5]. Historically, dramatically faster new GPUs have been coming into the market every 12 to 24 months at the same general price point as the older technology but with substantially better computational and visual performance than even predicted by Moore at Intel in 1965. Therefore, rather than relying on spare parts with today's (soon to be obsolete) technology sitting on the shelf, the general life-cycle plan for the OBVA IG is instead to refresh with "equally capable or better" compatible hardware in the future when required.

Figure 4 depicts the rack mount computer (CPU) and graphics processor unit (GPU) design for the OBVA simulator as well as for the host computer and other required electronic components / interface hardware. Since the target computer room at WPAFB had limited ceiling height, the computer rack design settled around the use of standard COTS 19" 24U racks that are only 42" high rather than the taller 42U racks.
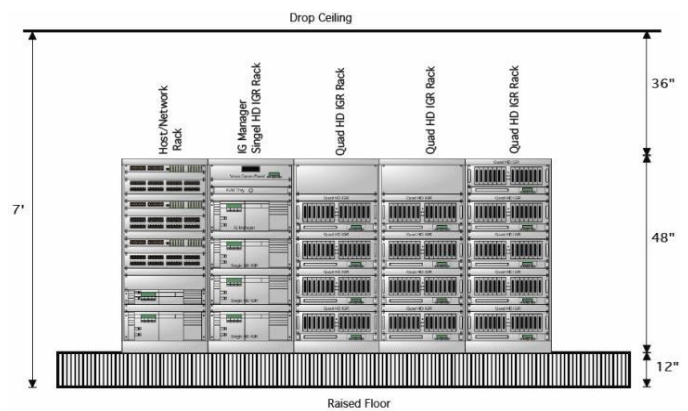

Figure 4 -Planned OBVA Computer Room Hardware

The Large Pixel Count Desktop / OBVA IG Channel
When large pixel count projectors were first introduced in the 2007 era from companies such as Sony and JVC [6], each of the four DVI (or HDMI) inputs to these new high pixel count displays were often driven by four separate GPUs and CPUs. These displays were dubbed "4K" (a possibly misleading term) since each display supports over 4,000 horizontal pixels and over 2000 pixels vertically.

With the advent of Quadro Plex® 1000 hardware from NVIDIA in that same 2007 timeframe, it became possible to drive large pixel count displays using a single computer and a single graphics subsystem containing multiple GPUs. In this case, the entire user desktop for a standard PC could then be set to an unprecedented resolution of say 4096x2160 pixels when using a new Sony SXRD 4k projector. What this meant to the application developer such as OBVA, they could conceptually create a single, full screen OpenGL (or DirectX) program rather than managing four separate programs each rendering to a fraction of the display (e.g., four programs each rendering to one of the four 2048x1080 DVI inputs at the projector).

The next generation NVIDIA Quadro Plex hardware in 2008 took this concept to a new level where two "4K" projectors (e.g., 4096x2160 each) could be driven from a single PC. With an NVIDIA dual-HIC interface board installed in the hosting PC, two Quadro Plex units could then be configured together to form a single 4096x4320 desktop when the projectors were vertically stacked one on top of the other, or 8196x2160 pixels when arranged horizontally, side-by-side.

In 2011, NVIDIA started shipping the Quadro Plex 7000 (QP7000) hardware with its latest GPU technology. In each Quadro Plex 7000, there are two (2) internal high-end Quadro GPU cards that are normally configured to generate one very large desktop or full screen 3D application. Using the standard NVIDIA rack mount option, there are actually two Quadro Plex 7000s mounted side-by-side taking 3U of rack space for the OBVA IG as shown below.

Figure 5 – Two Rack Mount Quadro Plex 7000s

When using two Barco Sim10 projectors, this means each PC desktop or full screen 3D application on the hosting PC controls 4096x4800 pixels when the two projectors are vertically stacked or 8192x2400 pixels when the projectors are horizontally arranged.

Objective - Cross Platform Application Software
A principal objective of the IG software for OBVA was to use commercially available cross platform software (C++) so the results could be deployed on a standard 64-bit operating system (OS), such as Windows or Linux variants such as Red Hat. This objective implied the underlying low level rendering software layer would be OpenGL. For OBVA, the IG software development team decided to use Windows largely because of its very strong software development and debugging tools (e.g., Microsoft Visual Studio 2010). Source control management was seamlessly integrated to the developer's Windows computers to existing NASA SVN servers at Ames and using local TortoiseSVN client software running on the developer's PC. TortoiseSVN was then seamlessly integrated into Visual Studio and successfully used for the small IG software development team.

Scene Graph Approach
A fundamental internal component of any image generator is the scene graph software which sits on top of the low level rendering software (OpenGL in the OBVA IG). The scene graph software is responsible for simplifying the complex rendering chores for the application developer with a highly optimized interface to the underlying GPU hardware. Instead of writing all of the application code in OpenGL, a developer instead uses the scene graph engine. After a brief market survey, the IG developers chose the NVIDIA SceniX scene management engine [7] which is downloadable from NVIDIA's developer site; it was the only API that properly imported a test database created using the latest OpenFlight file format at that time. SceniX operates on Windows or Linux operating systems, addresses large dataset visualization, and includes a thin, efficient interface to allow advanced GPU shader algorithms to be developed by the developer such as light point rendering or other special effects that may later be used "as is" or tweaked by an OBVA research scientist.

The standard SceniX toolkit includes a number of demo programs and framework examples to popular cross platform user interface design tools (Qt, wxWidgets or Microsoft MFC) that proved very useful to the OBVA developers for the chosen wxWidgets based graphical user interface (GUI) code used in the IG Manager component which is discussed below in the section 'IG Manager'.

SceniX also includes samples for the open source Graphics Library Utility Toolkit (GLUT) which offers an OS independent interface to the OS windowing system; since the OBVA IG software uses the more recent freeGLUT software, the GLUT examples provided with SceniX proved helpful when writing the IG Renderer (IGR) code.

As mentioned earlier, SceniX also included good support for the latest OpenFlight file format (and others). Efficient support for the latest OpenFlight format was a firm requirement for the OBVA program since the underlying terrain databases and moving models are created and stored in this popular simulation file format. OpenFlight databases are converted to a SceniX compatible, optimized runtime format using standard SceniX tools (i.e., makenbf). Cg or CgFx shader programs that are attached to select OpenFlight nodes are automatically compiled into the requisite runtime format as part of makenbf processing.

Since application programs produced using any scene management software must ultimately operate in "real-time" with minimal latency, the scene graph software must be very fast at runtime, efficient to use, and extensible to support new future graphics hardware and functions. The OBVA project goal described earlier for achieving 120 Hz frame rates for the visualization system has proven highly successful using the SceniX engine. The typical OBVA IG render times for sample pictures shown below (prior to swap buffer calls each frame) have been measured between 3.0 and 6.0 milliseconds on average for relatively complex terrain scenes (see the section 'Database Generation System' below) making it possible for the USAF to later support 120 Hz 4k projectors or displays in the future if and when they become commercially available.

Objective – Minimum System Transport Delay
The program objective for the OBVA IG is to provide no more than four frames of latency from a stimulus within the cockpit to the last pixel out on the projectors. The IG system itself needs to be responsible for no more than three frames of latency (ideally, only two) when the system is setup for synchronous operations. At a 60 Hz update rate, three frames would be nearly 50 milliseconds. With a 120 Hz update rate, the IG contribution to system latency would be ½ (no more than 25 milliseconds with three frames, and less than 17 milliseconds with two frames).

Objective – High Accuracy COTS Distortion Correction
The cross platform Scalable Display Technologies EasyBlend software along with their Windows based Display Manager software was initially acquired under a

trial use basis and tested on a subset of the entire system, a 2x2 grid of 4K projectors. A geometry warp mesh was generated using a Canon digital SLR camera (Rebel T2i) with the standard Scalable Display Manager software. Since Barco 4K projectors were not yet available during IG software development, four Sony 4K projectors were instead used (two Sony S110s and two Sony T110s). The four projectors were arranged in a 2x2 grid and were driven by two OBVA IG channels. The IG team tested both a horizontal configuration (8196x2160 pixels) and a vertical stacking configuration (4096x4320 pixels). The calibration software had to compute the blending zones in each case where light from the four (4) adjacent 4K projectors overlapped.

Pixel placement mismatch in the overlap zone after the Scalable setup process was not visually evident. Lines across projectors appeared to land squarely on top of lines and pixels from neighbouring projectors. Previous perceptual test conducted by the USAF using EasyBlend software and Sony SRXD projectors established that discrimination of the orientation of a triangle outside the blend zones required a 2.52 arcminute triangle base size (5 pixels at 0.5 arcmin pitch). Although the triangle base size required to make the same discrimination within two-projector and four-projector blend zones were larger (2.76 and 2.73 arcmin, respectively) the small loss of resolution was deemed acceptable. This test has not yet been performed using the Sim10 projectors.

The distortion correction alignment process for the four Sony SXRD projectors at Ames was done on a partial section of the actual OBVA dome surface (see Figure 3); it took less than an hour to complete which included the blend zones. Color balancing of the four projected images has not been done at this time in the program because the USAF selected the Barco Sim10 projectors which were not available at the time this paper was written. Color balancing in general can be a difficult task and it is not discussed in further detail in this paper. Ideally, this issue is best addressed with automation if available since manual adjustments can be tedious and time consuming.

Objective – Zero Latency Distortion Correction
Historically, dome distortion correction has often been accomplished using separate hardware subsystems where the image generator pixels are adjusted to account for the inherent distortion of the spherical display surface. These external subsystems often add at least one frame of latency and usually more. Further, at least some of the hardware based distortion correction subsystems could not accept the large pixel counts in the video being produced by the

Quadro Plex GPU to drive two Barco Sim10 or two Sony T110 4K projectors. The Barco Sim10 projectors do support built-in warp correction but this feature could not be tested or compared to the Scalable software approach since the OBVA IG development team did not have access to these new projectors.

The goal for OBVA geometry distortion correction was "zero" frame latency; that is, no additional frames being introduced by the visual system as a result of dome (geometry and blending) correction. This proved successful as the overhead for performing the Scalable geometry distortion correction was measured and preliminarily found to be less than 0.5 milliseconds when integrated to our SceniX based application software. Since there is sufficient render time available and remaining in the target 8.333 millisecond frame, the EasyBlend SDK did not affect the update rate in the system, and therefore introduced no additional latency as desired.

The IG Renderer SceniX based application software has since been adapted to directly accept a standard Scalable .ol warp mesh file (per channel) which includes the blend zone and distortion correction "warp mesh" definitions produced by the Scalable Display Manager software. This allows the IG end user to easily setup and control the distortion correction on any dome or non-uniform display surface by simply pointing to the setup file for each IG channel. It can also be easily disabled for comparison with hardware based solutions at a later date.

Objective – Use COTS Synchronization
Another objective for the OBVA IG was to use commercially available software and hardware for synchronizing all of the dome projectors and channels. The NVIDIA Gsync2 hardware was chosen for this task because on paper, it had all of the salient characteristics to facilitate synchronization across all dome channels:

- Framelock,
- Swaplock, and
- Support for Stereo synchronization

With framelock synchronization, the vertical retrace for each projector can be synchronized using an external "house sync" signal supplied over a standard 75-ohm BNC video cable. This same house synchronization signal is also supplied to the host computer. This allows all IG dome computers and the host computer to be operating on the same system "heartbeat" pulse (when integration is complete) which will also help minimize system latency.

With the swaplock capability present in the standard Gsync2 hardware, each of the dome IG renderer swap buffer calls can also be fully synchronized.

The QP7000 graphics hardware already includes Gsync2 hardware. All dome channels are then easily connected to the synchronization chain using short CAT5e network cables which then pass along the framelock and swaplock Gsync2 signals. The OBVA IG developers then used industry standard OpenGL driver extensions, notably `WGL_NV_swap_group`, to implement framelock and swaplock logic on the dome channels.

The OBVA IG does not currently have a requirement for the stereo Gsync2 synchronization signal, but this capability is felt to be a potentially useful feature for future vision research or simulator use at NASA or the USAF.

IG Computer Architecture
The IG software is implemented across multiple PCs in a master-slave relationship. Unlike most past IG implementations, the external host computer for OBVA is purposely permitted to transmit data at runtime on the internal, private IG local area network (LAN) via a UDP multicast mechanism to help minimize system latency. The OBVA host computer is a Concurrent iHawk PC running the real-time RedHawk OS; one of the host's network interfaces is used to transmit IG commands using the JIGI data packet protocol (see the 'Host Computer Interface (JIGI)' section below).
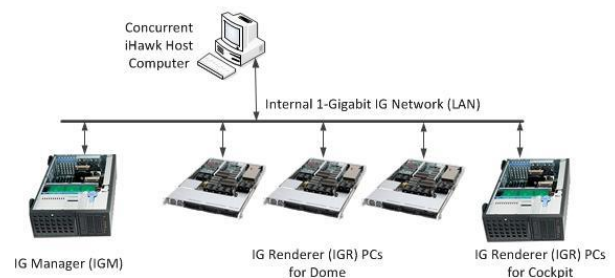

Figure 6 – Top Level IG Architecture

The OBVA host computer interfaces to a reconfigurable cockpit built by NASA. The single man cockpit includes a 24-inch touch screen and reconfigurable inceptors, either an F-16 stick and throttle subassembly or helicopter inceptors.

Unlike most host computers, the OBVA host computer not only includes the typical real-time aerodynamics and six degree of freedom (6-DOF) equations of motion math models (e.g., F-16) for manned flight, but the OBVA host also includes a real-time software interface to MATLAB for experiment design. Thus, vision scientists can design, develop, and operate research experiments using MATLAB to first setup the experiments which then automatically interface to the real-time OS using application software developed by NASA for the USAF. This makes it possible for the vision research scientist to develop their experiments without having to write complex C or C++ code.

The master IG computer, referred to as the IG Manager or IGM for short, allows the user to manipulate the slave IG computers using a straightforward graphical user interface. IGM takes care of chores for launching or terminating remote IGR applications, booting, shutting down, or even restarting the network IGR Slave computers. The IGM is also responsible for synchronizing user specified folders on all IGRs so they remain identical to the contents on the IGM.

Each slave in this architecture is referred to as an IG Renderer, or IGR for short. The IGM is aware of the IGR computers in its local IG domain and automatically starts up and terminates applications when required by the IG user. A simple double click on a desktop icon for example can startup all remote applications across the IG network.

The Concurrent iHawk host computer to IG data packets on the internal IG network have precedence over the IGM at runtime; the IGM GUI automatically reflects changes (if any) made by the external host computer to the IG on its appropriate pages (tabs).

IG Manager (IGM)
The IG Manager is a 4U rack mount PC (Colfax FX7400) that runs the Windows Server 2008 R2 64-bit operating system. The IGM computer has three network interfaces; one for the internal private IG network, another for the gateway to the internet when available, and a third for the internal development network. The server OS was chosen for IGM (rather than Windows 7 64-bit) so the IGRs could in the future be more easily setup for remote boot operations from the IGM over the internal IG network if desired.

The IGM graphical user interface (GUI) software was implemented using the wxFormBuilder open source software; wxFormBuilder produces C++ code for the

wxWidgets software layer used in IGM. Consistent with our cross platform development philosophy, wxWidgets software is also open source software and is available for download on the web.

The IGM includes a small solid-state drive for the boot / system volume and a separate RAID-5 4-TB hard drive subassembly for OBVA IG specific data and programs.

IG Renderer (IGR)
The IGR PCs are slaves to the IGM. Unlike traditional PCs, they are intended to be wholly controlled by the IGM although in practice for the current version, OS updates to the IGRs are done locally on each computer at this time.

There are two types of renderer computers:

1. 1U rack-mount computers with NVIDIA Quadro Plex 7000 graphics (3U form factor) for driving the dome projectors and other 4K displays

2. 4U rack-mount computers with dual NVIDIA Quadro 6000 GPUs for driving cockpit displays and HD resolution out-the-window displays in the conference room and operator control room

1U Out-the-Window Dome Renderers
Each dome renderer is responsible for driving two 4K projectors. Each PC (Colfax CX1050) for dome rendering duties is mechanized in a 1U form factor, and each is a server class PC. Each PC includes a single NVIDIA PCI-Express x16 Generation 2 dual-HIC interface card. The CX1050 has been certified by NVIDIA as a supported PC for the demanding dual-HIC interface card. The dual-HIC interface board drives two Quadro Plex 7000 rack mount graphics subsystems; the rack mount dual QP7000s are implemented in a 3U form factor that is then configured with NVIDIA's standard configureMosaic program to setup a single, very large desktop as described earlier.

Each 1U IGR runs the Windows 7 64-bit Ultimate OS, and like the IGM, each contains a solid state drive for the boot / system volume, and a separate RAID-5 4-TB hard drive subassembly for OBVA IG specific data and programs.

4U Cockpit Display Renderers
The less expensive NVIDIA Quadro 6000 GPU boards are used for HD or lower resolution displays (e.g., less than 2560x1440). Unlike the 1U PCs, the 4U IGR computers contain one or two Quadro 6000s each; they do not include the dual-HIC cards (although they likely could). The NVIDIA Quadro 6000 GPUs are used for all non-dome displays, including but not necessarily limited to the cockpit heads-down displays and for displaying lower resolution out-the-window scenes. Each 4U IGR runs the Windows 7 64-bit Ultimate OS, and like the IGM, each contains a solid state drive for the boot / system volume, and a separate RAID-5 4-TB hard drive for OBVA IG specific data and programs.

Atmosphere & Clouds Visual Simulation
Simulating reality such as clouds that we see every day in the real-world is not easy to do. There is often a careful balancing act between performance and appearance. Even with state-of-the-art QP 7000 GPUs, it is relatively easy to tax the 'fill rate' of the graphics hardware, especially when flying through a series of cumulus type 3D clouds that automatically affect the observer's visibility with a high degree of realism.

The OBVA IG uses a commercially available software development toolkit called Silverlining™ from Sundog Software [8]. The Silverlining SDK implements 3D cloud layers and is responsible for rendering the background skydome (sun, moon, stars) with an active ephemeris model.

Cockpit Display Simulation
The OBVA IG uses the popular human interface simulation tool, GL Studio [9] from DiSTI to implement simulated cockpit displays for the OBVA IG. The GL Studio developer tool produces OpenGL compatible C++ source code from their developer GUI.

Host Computer Interface (JIGI)
The host computer to IG bi-directional interface is a data packet protocol intended for implementation over a local area, high speed Ethernet network. The private IG network is a one gigabit per second network interface. The legacy open source Common Image Generator Interface (CIGI) softwares was strongly considered for adoption for the OBVA IG, but a number of required features were not available in the current CIGI 3.x feature set, at least without extensive use of custom data packets.

Instead, a 'JIGI' interface protocol was adopted for OBVA that includes general purpose data packets required by the research team. JIGI is an informally adopted name describing the host-IG and other interfaces, and it was catchy, so the abbreviation 'stuck'.

The JIGI interface control document is part of the OBVA IG documentation and defines all data packets between the host and the IG as well as between the IGM and slave IG Renderers. JIGI provides a highly customizable data packet protocol that can be more easily extended in the future to implement nearly anything the research scientists dream up, including hooks to control existing shader functions implemented within the IG rendering software.

Database Generation System (DBGS)
What good would an image generator with eye limiting resolution be if the synthetic world details were not at least partially eye limiting? Even with the world's most powerful GPUs, it remains impossible to visually simulate every detail found in 'reality' at a 120 Hz update rate. The best we can still do in this environment is to approximate the details of the real-world by preserving only the most important features required for the mission (in our case, vision research where eye limiting detail must be available at least in designated select areas).

The database generation system designed for OBVA is built on top of two popular and commercially available software products:

1. Google Earth Pro, and
2. Presagis Creator Pro

The OBVA DBGS COTS solution allows government end users to generate visual databases for non-commercial use by automatically capturing imagery and terrain elevation data from Google Earth Pro (or other compatible servers) and then automatically converting these to the OpenFlight format using the Presagis Creator Pro editor.

Cultural 3D models (e.g., hangars, control towers, and other important man-made structures) can also be included in the visual database from select buildings extracted from Google Earth Pro, or hand modeled in Creator.

At this time, a visual database environment for the current IGR application software design must fit into the available 6GB of GPU memory present in the QP7000 graphics hardware. This is not a sufficient approach for databases that exceed GPU memory. Real-time database paging remains a challenge for future work.

RESULTS

Below are some screen captures that depict the first target database for OBVA, Sheppard AFB where each pixel in the airport area is about one foot resolution. This 300 x 300 mile database area was selected because it was one of the USAF pilot training facilities.

All of the runway surface markings and other airport details in the screenshots below were generated from the Google Earth Pro source imagery and not hand modeled as would typically have been done for past image generators. High resolution imagery for this database is currently only present around the immediate airfield area; lower resolution images are used elsewhere.

The airfield area in the pictures below is modeled with over one hundred 4096x2048 pixel texture maps and over three hundred 2048x1024 texture maps elsewhere, all stored in the DXT1 format. The OBVA IG team is currently investigating possible use of larger texture map files (i.e., 8192x8192 pixels); the hardware supports up to 16384x16384 pixels per texture map.

Figure 8 – Sheppard AFB Approach on OBVA IG

Figure 9 – F-35s below the 3D Clouds on Approach

CONCLUSIONS

The utilization and selection of the right COTS tools and components (hardware and software) for the OBVA Image Generator have proven successful thus far. Two (2) software developers designed, developed and integrated the OBVA IG in approximately one year; one (1) visual database development engineer helped design the processes to construct high resolution visual databases using COTS technologies in ways that have never been attempted before.

As this paper is being written, the OBVA team is readying the simulator for initial delivery to the USAF at Wright-Patterson AFB, Ohio from the NASA Ames Research Center. The delivery and setup for the OBVA simulator is planned to be no later than the mid-June 2012 timeframe.

Further results and experiences to date will be highlighted and discussed at the IMAGE 2012 Conference.

ACKNOWLEDGMENTS

- Dr. Barbara Sweet
  NASA Ames Research Center, Code TH

  Barb is our NASA task leader and is one of the original OBVA visionaries who helped forge the way!

AUTHOR BIOGRAPHIES

Mr. Archdeacon received an M.S. degree in Computer Science from Wright-State University and a B.S. degree in Computer Science from the University of San Francisco. He has been working with real-time computer image generator (IG) technology for the Department of Defense for nearly 30-years. He joined NASA Ames and the OBVA team as a federal contractor in 2009 to lead the OBVA image generator proof-of-concept activities, then the design, development and implementation of the IG once the Air Force approved development of the OBVA simulator. Prior to working on the OBVA program, Mr. Archdeacon was a founder and principal at Quantum3D, Inc. and before that, with Gemini Technology Corporation where the OpenGVS™ software was developed as the world's first cross-platform scene graph software for simulation using a PC in the Windows 95® era with 3Dfx Interactive, Inc. graphics boards (Voodoo™) while also supporting high end OpenGL 3D UNIX workstations from companies such as Silicon Graphics.

Dr. Gaska received a Ph.D. degree in Biopsychology from Syracuse University and from 1981 to 2000 worked at UMASS Medical School where he used single cell electrophysiological and computational techniques to explore and model the representation of the world in the visual cortices of primates. From 1995 to present, Dr. Gaska worked as contractor to the Air Force and used vision science and visual performance models to aid in the design of camouflage, laser eye protection and flight simulation technologies. He is currently a Research Psychologist in the USAF School of Aerospace Medicine at WPAFB supporting the OBVA program.

Dr. Timoner is a founder of Scalable Display Technologies and serves as the company's Chief Technical Officer (CTO). Dr. Timoner has a Ph.D. in image processing from MIT. He has been delivering automatic calibration software to the Navy, Marines, Air Force, Coast Guard and others for more than 6 years.

REFERENCES

[1] J.P. Gaska, M. Winterbottom, W. Sweet and J. Rader 2010, "Pixel Size Requirements for Eye-Limited Flight Simulation", *Proceedings of the IMAGE 2010 Conference*, The IMAGE Society, pp. 123-130.

[2] A.B. Watson, A. J. Ahumada, Jr., and J. E. Farrell, 1986, "Window of Visibility: a psychophysical theory of fidelity in time-sampled visual motion displays", *JOSA A*, Vol. 3, Issue 3, 300-307.

[3] M.D. Winterbottom, G.A. Geri, W.D. Morgan and B.J. Peirce, 2004, "An integrated procedure for measuring the spatial and temporal resolution of visual displays", Proceedings of the interservice / industry training, simulation and education conference. ASA430109, Paper Number 1855.

[4] B. Sweet, 2012, "120 Hz – the New 60 for Flight Simulation?", Proceedings from Image 2012 Conference, The Image Society

[5] K. Akeley, Microsoft Research, "GPU Gems 3", Foreword from book, August 12, 2007, ISBN-10: 0321515269 | ISBN-13: 978-0321515261

[6] Siggraph 2007, UC San Diego, California
http://ucsdnews.ucsd.edu/newsrel/general/07-07SIGGRAPHSuperHighDefinitionVideoDR-.asp

[7] http://www.nvidia.com/object/scenix.html
NVIDIA Scene Management software engine

[8] http://sundog-soft.com/sds/features/real-time-3d-clouds/, Sundog Silverlining SDK

[9] http://www.disti.com/Products/glstudio/
GL Studio human interface design software from DiSTI Corporation