VIRTUAL DRIVING

VR City Modeling and Drive Simulation in Real Time

#145

Abstract. This paper introduces a VR city model developed for driving simulation. The project is a part of interdisciplinary multi-year academic research grant. First, the outline of the research is explained. The process of VR city modeling is then introduced. The modeling process of creating road and intersection networks and traffic flow is explained. The system integration with a PC, modeling and VR software, and a Drive Simulator is illustrated. A case study of driving through the city with different traffic amounts using the simulator is analyzed. Computational tools to extract driving behavior data and future endeavors are discussed.

Keywords Drive Simulator, Traffic Visualization, Virtual Reality, Driving Behavior, Micro-Traffic Simulation

1. Introduction

Virtual Reality (VR) is a simulation of a specific environment, whether it be real world or fantasy. It enables an interactive experience with computer generated 3D models in a particular situation or scene. Virtual 3D cities have been recently available to the roughly 1.5 billion internet users across the world free of charge (www.internetworldstats.com). Two of these programs include Google Earth and Microsoft Virtual Earth (Google 2009), (Microsoft 2009). Both programs allow users to navigate the globe and immerse themselves in 3D terrain and virtual city environments. On a professional and academic level, these VR programs are vital to the fields of architecture, urban planning, and geography. 3D visualization is an integral part of the workflow and decision making process in these disciplines.

A drive simulator is a hardware tool that can be used for entertainment, training, education, or research purposes. It consists of a vehicle interior, with seat, steering wheel, brake and accelerator pedals. There is a screen that usually shows an out-the-windshield view of the drivable VR scene. It is an immersive environment with realistic driving conditions and controls as well as vehicle sounds. Drive simulators can monitor driver behaviors and train drivers to operate different kinds of vehicles in a safe VR environment.

This paper presents the relatively new concept of using a driving simulator in conjunction with a realistic VR model and road network of an actual city (Phoenix, Arizona). The city and road network were modeled using an off-the-shelf driving simulation package. Our goal is to create a VR city environment with traffic simulation capabilities that emulates reality as closely as possible. We aim to develop computational tools that will allow us to extract driving behavior data from the simulation for analysis and application on several other research endeavors.

2. Background

A VR city model of Phoenix, Arizona was created as part of the Digital Phoenix project. This grant project was collaboration between architecture, urban planning, design, and high speed computing at Arizona State University. The project’s goal was to visualize the past, present and future of the core downtown area while creating a tool to better understand growth patterns, policy changes and urban dynamics of the city of Phoenix and Maricopa County. A 3D model of the metropolitan area was a large part of this project. Urban planning and high speed computing used a software tool, UrbanSim, to run macro simulations of social, economic, and environmental scenarios. This data was integrated and visualized to better understand the impacts of policy choices of the growing urban area. The broader research objective of the project as a whole is to develop computational tools and digital contents to visualize and simulate several urban issues at a 3D immersive theater.

A VR city model of a one mile square of downtown was created using a variety of off-the-shelf software packages. It included approximately 500 buildings quickly modeled and textured using photogrammetry. We also created approximately 70 roads, 200 intersections, and a section of the 1-10 (a major US Interstate that runs through the heart of downtown Phoenix). In addition to the existing buildings, many large-scale construction projects scheduled for completion by 2011 were created by hand using renders from several architecture and development firms. The VR environment was visualized in a traffic simulation and civil engineering software package called UC-win/Road (Forum8 2009). Popular in Asia, this software had never used to simulate such an extensive area. A built-in function of the software is automatic traffic generation on a created road network. Each traffic element (cars, motorcycles, buses, and trucks) behave as an intelligent agent, stopping at red signals and changing speed and lanes corresponding to the surrounding vehicles. These vehicles have the function to be controlled by an exterior input device. A gaming pad, a gaming steering wheel, or a full-size driving simulator can be used to control a vehicle with an in-car perspective. The research donation of a driving simulator by Forum8 allowed us to explore the case study of real world traffic scenarios while driving through the city at various times of day. It also enabled software customization to extract driving behaviors from the simulator.

3. Methods

3.1. MODELING OF VR CITY

Three pieces of modeling/VR software were used in the creation of the virtual city environment: Nverse Photo, 3D Studio Max, and UC-win/Road. In order to save time in both modeling and texturing of buildings, photogrammetry was the chosen method. A pilot was hired to fly over the city of Phoenix at 6000 feet. He was directed to take one photograph straight down, and four angled shots as he inclined upward from north, south, east, and west. The negatives from these pictures were scanned in at 2400dpi, resulting in each .tiff image being larger than a gigabyte. A Precision Lightworks product, Nverse Photo, was used for the rapid 3D modeling of the urban environment (PrecisionLightWorks 2009). This is a plug-in for 3DS Max that utilizes photogrammetry for quick modeling and texturing. The five aerial images are imported into the software, and linked together by tagging the same reference points in multiple shots. Buildings are quickly and efficiently modeled through an extrusion and automatic texture wrapping process as shown in Figure 1. This allows for an extensive urban area to be modeled in a fraction of the time it would take to do this manually. A two mile by one mile area of downtown Phoenix was modeled, including over 500 buildings in a two week period by two individuals. 3DS Max was used to hand model and texture future buildings.

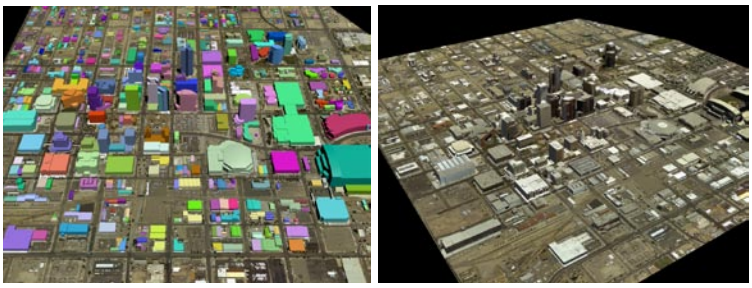


Figure1. Nverse Photo models and ground plane; models with textures

3.1.1. Frontend VR Package: UC-win/Road

UC-win/Road is a VR environment that allows users to navigate in a 3D space. The software was developed as a traffic simulation and visualization tool for civil engineering. The first step in creating the virtual city environment is to import the ground texture maps. An orthorectified satellite image is tiled and mapped onto the ground plane of the VR space. Adobe Photoshop was used to tile this image. UC-win/Road imports .3ds models into its VR space. The second step is to import the hundreds of buildings created and textured by Nverse photo. They were exported out of 3dsMax in quadrants of 24 buildings. The third step is to place the quadrants on the correct location on the ground map. UC-win/Road has built in functions to move, scale, and rotate imported objects in 3D space, which makes it easy to line up the quadrants properly shown in Figure 2.

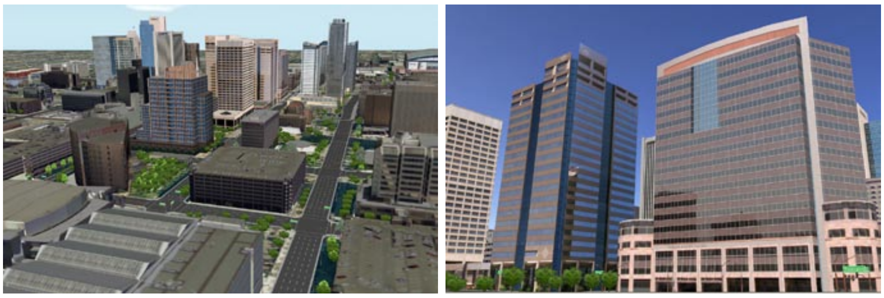


Figure2. Buildings imported in to VR environment (UC-Win/road)

3.2. ROAD NETWORK CREATION AND TRAFFIC CONTROLS

UC-win/Road has an intuitive function to create road networks and automatically generate intersections. Once a road network is created, the software will generate 3D modeled traffic on the roads. The traffic will respond to traffic signals, road hazards, and other cars as it progresses through the VR environment. In a 2D overhead view, the satellite ground map is used as a reference. Vector lines are traced over the satellite image in the approximate location of the roads. Each road can be edited to change the number of lanes, road texture, the size of sidewalk, cutting, banking, and road height. Where these lines cross, default intersections are automatically generated. Once created, the intersection has many editable features. Lines can be added to the texture to represent stop lines and crosswalks. The intersection can identify when a 3D model of a traffic light has been placed in its vicinity, and will control the light accordingly. The timing of the lights and the signal phases can all be manually set. Drive paths of vehicles through the intersections as well as stopping points are also fully controllable.

Once back in the 3D view, the software will generate fully functional roads with textures. The software comes built in with a vehicle library including many varieties of cars, busses, trucks, motorcycles. The amount of vehicles per hour can be set for the start and end of each road in the model. At the click of a button, traffic is simulated and will drive throughout the road network in the model (see Figure 3). This feature, coupled with the importing of 3D building data, creates a lifelike virtual urban environment. By ctrl+alt clicking any vehicle on any road, the user is given an in-car perspective of the 3D scene. If a driving simulator or gaming wheel is attached to the PC, the user can actually drive the vehicle around the model.

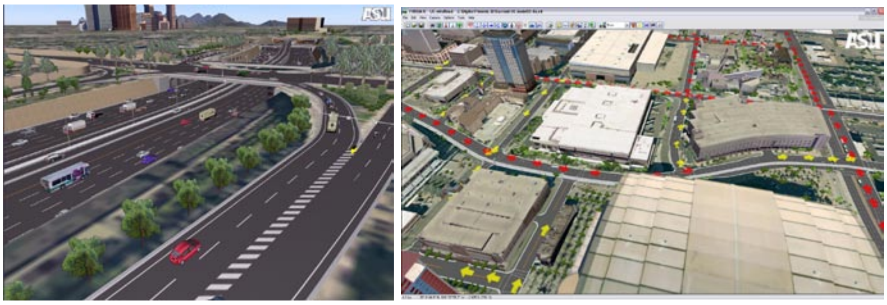


Figure3. Traffic moving along road network: using 3D automobile models (left) and with symbolic 3D arrows (right)

3.3. IMPLEMENTATION OF PLUG-INS

As described in section 2, one of the objectives is to visualize the simulation results using the VR city model. Our solution is to visualize the traffic flows using the simulation result from a micro traffic simulation tool, TRANSIMS (Los Alamos National Laboratory 2009). The tool can calculate the travel paths of all people from their homes to their job places in the future using the outcomes from UrbanSim. In order to set the traffic amount for each point in the street networks, the plug-in tool is used to assign the proper traffic amount for each street is implemented using the SDK of UC-win/Road in Delphi 2007 (CodeGear 2009). UC-win/Road is implemented in Delphi, which is an object-oriented programming language. It uses an OpenGL library in the rendering engine. The current version of the SDK allows us to access the geometrical data and traffic objects including the street and its traffic amount. The plug-in tool reads the text file describing the name of street and traffic amount per hour for each line. For example, the following lines generate 300 cars/hour at the Jefferson Street and 500 cars at the Washington Street:

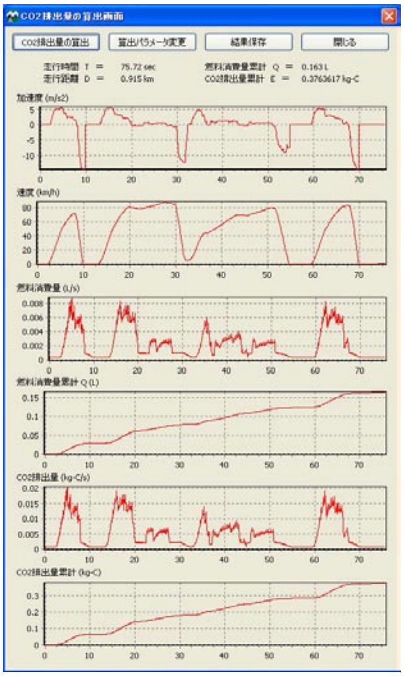
“Jefferson Street”	300
“Washington Street”	500

Once the text file is read, the traffic amounts are set in the VR environment. The default portion of cars, trucks, and busses is used for each street in the tool.

Once the traffic amounts are set for all streets, a subject can drive the proper route in the different traffic conditions in the drive simulator. UC-win/Road offers several functions to extract the driving data. According to the research by Oguchi, the CO2 emission amount can be estimated using the following formula:

$$E = K_c (0.3T + 0.028D + 0.056 \sum_K \delta_k (v_k^2 - v_{k-1}^2)) \qquad 1)$$

,where E is the CO2 amount (kg-c), T is the traveling time, D is the travel distance (meter), K is the number of speed measure points, δ k is the speed incremental value (1 or 0), and vk is the speed (m/sec) at the point k (Oguchi 2002). Figure 4 shows the CO2 calculation panel.



走行時間 T = 75.72 sec
走行距離 D = 0.915 km
燃料消費量累計 Q = 0.163 L
CO2排出量累計 E = 0.3763617 kg-C

Figure4. Screen Shot Image of CO2 Emission Calculator

4. System Integration

Because of the vast amount of texture data associated with the VR city model, a higher end PC is required to run it smoothly. The PC currently running the model and simulator has dual Xeon processors, 4 GB ram, an Nvidia Quatro fx4500 video card (512 MB dedicated memory), and operates on the Windows XP platform.

The drive simulator connects to the PC with two USB cables for data transfer. The 45” LCD display connects to the computer with a 30ft VGA cable. The drive simulator has built in 5.1 surround sound speakers, with a subwoofer located under the seat to feel vibration. An external Creative soundcard also connects to the PC via USB 2.0, and to the simulator with three 3.5mm jack speaker cables. The simulator components and the speaker amplifier plug into a standard outlet with two universal PC power cables.

Once both the computer and the drive simulator are powered up, UC-win/Road is opened. This connects the two devices, and the simulator turns the wheel automatically to calibrate it. The user sits in the driver’s seat. Buttons on the dashboard let a user scroll through options for driving. A predesigned scenario can be ran with constraints to a given area and vehicle, or the user can freely drive on any road with any vehicle. Once the selection is made, a first person view of the roadway appears on the LCD screen. The user must start the vehicle with the key by turning the ignition. The user then depresses the brake pedal and puts the vehicle in ‘drive’ using the gear shift. Once in drive, the vehicle performs operates true to life. The simulator is built with the seat, pedals, and steering column directly out of a Hyundai so it looks, feels, and responds like a real car (see Figure 5). The software still has complete modeling and editing functionality using mouse and keyboard even while connected to the simulator.

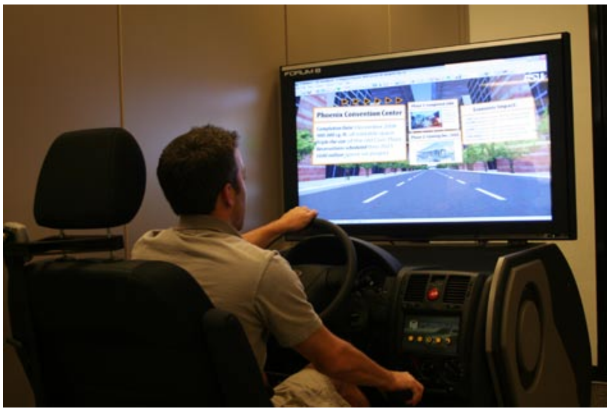


Figure5. Drive Simulator in use.

5. Other Applications

Components of the VR city model have been explored through collaborations at Arizona State University. One such application of the model is running a VR environment in a 3D immersion theater. The Decision Theater consists of an interactive 3D immersive environment built with cutting edge graphics technologies. The core component, called the Drum, is a 260-degree faceted screen that can display panoramic computer graphics or 3D video content. The Drum has seven screens that seamlessly integrate into one gigantic display (see Figure 6).

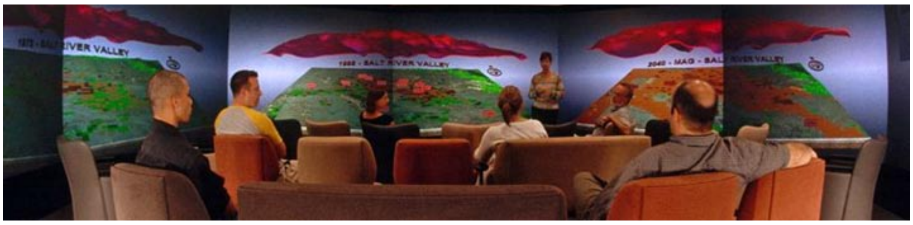


Figure 6. Decision Theater’s 3D immersive environment

Using a customized version of Google Earth, the theater can surround its guests in VR content. Since the original building models were created in 3D Studio Max, they can be exported in COLLADA format and imported into Google Earth as shown in Figure 7. This allows buildings to be tagged with information such as carbon footprint, approximate jobs, historical information, height restrictions, etc. This information can be catalogued in a database and loaded into the software on cue.

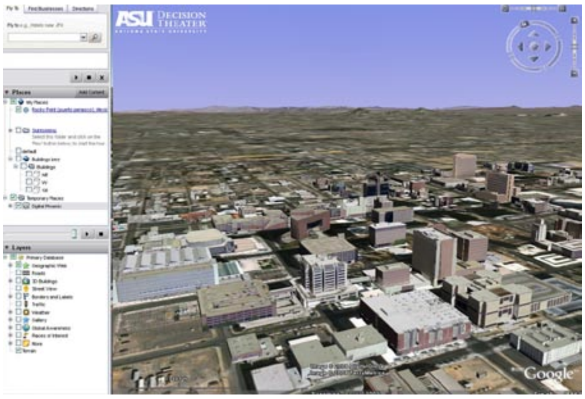


Figure 7. Screen Shot of Google Earth with Phoenix City model

Another Decision Theater hardware tool used was the Imove® 360° camera. This vehicle roof mounted device has 6 cameras recording video as the car is driven. Included software stitches the videos together, allowing for 360° rotation during video playback. The camera was used to record major roadways throughout the core of downtown Phoenix. This video was then played side by side with our UC-win/Road VR city environment in the Decision Theater. In both the video and the model, the same road was simultaneously navigated and the view rotated all around. This tool was essential in validating the accuracy of the VR model to the real world environment without having to actually go downtown (see Figure 8).

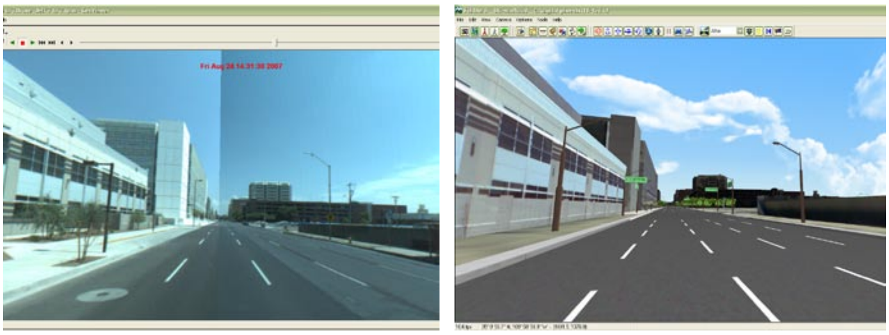


Figure 8. Imove® camera view (left) and
VR model view (right) at the same location

6. Case Study

A case study was developed using the VR city model and the drive simulator. The goal of the study was to see if highway traffic flows could be accurately modeled and a drive simulator used to replicate travel from the I-10 to a given landmark in a timely fashion.

First, the modeling team created the multi-lane interstate, including lane conditions, tunnel section, overpasses, street signs, embankments, and foliage. Major intersections between the highway exit and the downtown corridor were accurately modeled with working street lights, drive paths, and textures. Video was taken from an overpass of the Interstate 10 on the section of the highway that passes through the VR model. This video was analyzed, and the amount of traffic was manually counted in both directions of the highway at two given time periods. They were during morning rush hour (8am) and mid afternoon (1:30 pm). Once the traffic was analyzed, the same traffic numbers were entered into UC-win/Road to accurately represent these given times. The drive simulator was then used to navigate through the traffic from point A (start of I-10 in the model) to point B (new convention center in downtown). This is shown in Figure 9. Alternate routes through the city were explored to see if time could be cut off the journey at peak traffic hours. Using a high definition video camera, each simulator session was recorded and analyzed.

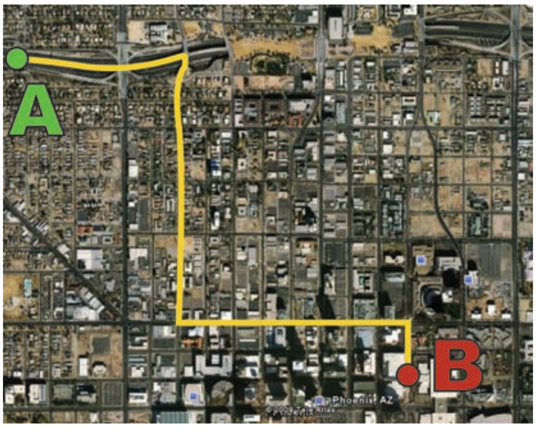


Figure 9. Route Map from Point A to Point B

Table 1 shows the case studies with two different traffic conditions. CASE 1 is with regular (midday) traffic, and CASE 2 is with heavy (rush hour) traffic. Table 2 shows the time taken while driving from point A to point B (three trials for each case). The average time in CASE 1 was 4 minutes and 31.6 seconds, and it in CASE 2 was 11 minutes and 37 seconds.

TABLE 1. Traffic Amount Setting

Street Name	Traffic Amount in CASE 1	Traffic Amount in CASE 2
I-10	5,000	15,000
5th Ave.	400	1,100
Van Buren	500	1,000
3rd Ave.	600	1,200
… Other streets has the same traffic amounts		

TABLE 2. Driving Time from Point A to Point B

	Driving Time in CASE 1	Driving Time in CASE 2
Test1	4’ 51”	11’ 08”
Test2	4’ 15	12’ 01”
Test3	4’29”	11’ 42”

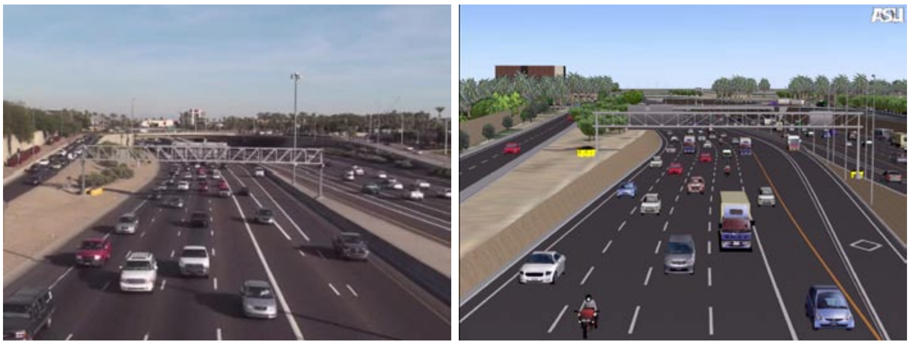


Figure10. Video of I-10 traffic flow (left) and VR simulation of I-10 traffic flow (right) at the same position

7. Conclusion and Future Works

We introduced a VR city model developed for driving simulation. The VR city model covered 1 mile x 2 mile area of downtown area with all buildings, all intersections with controllable traffic signals, and all streets with the exact lane conditions. The process of making buildings, street networks, and traffic data was described. We also developed a plug-in tool to assign traffic amounts for each street from a text file referring to the street name in order to visualize the result from the micro traffic simulation seamlessly. We developed a system to extract driving data in the VR space using a drive simulator. By using the VR city model and plug-in tool, we analyzed the time of driving in the different traffic conditions.

The future works is to develop more flexible system to extract any kind of driving data in VR environments using the drive simulator. For example, 1) the evaluation system of good driver, safe driver, or eco-friendly driver; 2) the cognitive system to analyze the vision in driving by integrating with the eye tracking systems; and 3) the interactive traffic control system to change the traffic amount interactively without restarting.

Acknowledgements

The authors wish to acknowledge the support received from Forum8 Co. Ltd. This project is sponsored by the Digital Phoenix project at Herberger Center for Design Research at College of Design, Arizona State University.

References

CodeGear: 2009, *Delphi 2009: The fastest way to build native Windows applications*, available at http://www.codegear.com/products/delphi/win32 (accessed 02/06/09)

Forum8:2009, *UC-win/Road product information*, available at http://www.forum8.com/english/uc-win/ucwin-road-e1.htm#30403 (accessed 02/06/09)

Google: 2009, *Google Earth: Explore, Search and Discover*, available at http://earth.google.com/index.html (accessed 02/06/2009)

Los Alamos National Laboratory: 2009, *TRANSIMS: transportation analysis simulation system*, available at http://www.ccs.lanl.gov/transims/ (accessed on 02/06/2009)

Oguchi, T., Kattakura, M., and Taniguchi, M.: 2002, *Carbon dioxide emission model in actual urban road vehicular traffic conditions*, Journal of Civil Engineering No.695, IV-54, 125-136.

MicroSoft: 2009, *Microsoft Virtual Earth: The integrated Mapping, Imaging, Search, and Location Platform*, available at http://www.microsoft.com/virtualearth/ (accessed on 02/06/09)

PrecisionLightWorks: 2009, *NversPhoto product information*, available at http://www.precisionlightworks.com (accessed on 02/06/09)