Proceedings of the 1st Human-Centered Transportation Simulation Conference, The University of Iowa, Iowa City, Iowa, November 4-7, 2001 (ISSN 1538-3288).

GENERATION OF REAL-TIME SYNTHETIC ENVIRONMENT USING A MOBILE SENSOR PLATFORM

Yiannis Papelis
National Advanced Driving Simulator
The University of Iowa
Iowa City, IA, USA 52242-5003
(319) 335-4597
yiannis@nads-sc.uiowa.edu

Brad Bossard
National Advanced Driving Simulator
The University of Iowa
Iowa City, IA, USA 52242-5003
(319) 335-4677
bbossard@nads-sc.uiowa.edu

Submitted July 28, 2001

ABSTRACT

Traditional methods used to generate synthetic environment databases for driving simulators often utilize automated software tools for the roadways combined with a lot of manual efforts for feature placement and fine-tuning. Automated tools, although they are much more efficient than manual techniques, have several shortcomings. They produce roadways that have a very regular and repeating appearance, in the sense that they contain no faults such as pavement cracks, potholes, asphalt runs, or any other such features that are so common in real roadways. Of course such "faults" can be added manually, but doing so in large databases is extremely time-consuming. Roadways generated by automated tools are often geometrically perfect, in the sense that super-elevation and crowning is applied consistently and smoothly, again with no faults typically encountered in real roadways. Finally, automated tools are often difficult to use when building geo-specific databases because the actual three-dimensional road geometry is not readily available. Our approach to addressing these concerns is to use advanced multi-modal sensor technology mounted on a vehicle that can survey a roadway while driving through it. The ultimate goal of the project is to use various sensors to obtain multiple-level-of-detail information about the road and the surrounding area, which can then be used to automatically produce synthetic environment databases for use in a driving simulator. Sensors include Global Positioning System (GPS) sensors for centimeter-level road geometry capture, scanning lasers for millimeter-level longitudinal and lateral road profile data, and various cameras combined with image processing techniques for automatic texture generation and feature identification. This paper describes the overall goals of the project, along with some preliminary results of using GPS receivers for survey of road geometry.

INTRODUCTION

The increasing use of driving simulators for various research activities has necessitated the development of tools to accelerate the development of typical driving simulation virtual environments. Developing such correlated virtual environments [1,2] is time-consuming because of the attention and subsequent labor necessary to ensure proper feature placement, texture application, road geometry construction, and proper modeling of the road network. Automated tools can be used to facilitate the road-generation process. Tile-based tools allow a user to simply combine prefabricated tiles that contain road segments and surrounding features in very little time. Despite the dramatic reduction of development time made possible by second- and third-generation tools, we have identified three key problems that are still unresolved. The first problem is the unnatural perfection of the output of the second-generation tool. The second problem is the labor involved in producing greatly varying road geometries. The third problem is the generation of geo-specific databases that utilize ground-truth road geometry. This paper describes a process that uses various sensors mounted on a mobile platform that can travel through a road network and collect data that, when post-processed, can recreate the geometry of the road network with enough accuracy to be usable in high-fidelity driving simulation real-time correlated databases. A subset of the described process was tested by using two GPS receivers mounted on an instrumented vehicle. The collected data were processed to extract geometry used to generate an OpenFlight® [3] database and additional correlated data usable on a driving simulator.

The remainder of the paper is organized as follows. Section 2 overviews prior related work and describes in detail the motivation for the work presented here. Section 3 describes the requirements associated with surveying roads, along with relevant properties of various sensors and vehicles that were considered for this project. Section 4 describes the overall process, including the post-processing algorithms for using sensor data to produce the final databases. Section 5 describes results obtained after an initial survey utilizing GPS and video cameras as the main sensors. Finally, Section 6 provides a conclusion about the feasibility of the process based on the results.

PRIOR WORK AND MOTIVATION

Automated tools that facilitate the driving simulation virtual environment creation process can be classified in three broad categories that roughly describe three generations of tools as they have evolved through the years. First-generation tools support *generalized* modeling and provide facilities for building and manipulating geometry either for terrain or for individual models and features. Such tools do not explicitly or exclusively support driving simulator applications, although a few have special features that support driving simulation. Examples of such tools include Multigen Creator [4] and Terrex [5]. Second-generation tools are s*pecialized* modeling tools that are specifically focused on driving simulator applications and provide facilities for the construction of road networks and features customized for driving simulator applications such as traffic lights, sidewalks, etc. Such tools can be found either in stand-alone form or as add-on packages to the generalized tools, for example the Road Tools® option of Multigen Creator [3]. A key characteristic of such tools is their awareness of the road network as something more than geometry, which allows them to provide data for further construction of road network databases [6,7] that are automatically correlated with the visual representation of the database. Finally, *tile-based authoring* tools are higher-level tools that allow a user to combine pre-existing libraries of existing road network segments, often called tiles, to create larger environments. A key feature of these tools is that they are designed with ease of use in mind and target individual users who may have no experience in virtual environment modeling. To remain simple, such tools do not provide geometry or feature-editing capabilities, which only allows a user to combine existing tiles or slightly modify the tiles in a pre-specified constrained manner. When using tile-based tools, one must have access to a library of tiles with enough variation to make it possible to construct larger non-repeating environments. Usually, first- and second-generation tools are used to construct such tiles, which can then be used in third-generation tools. Note that whereas generalized tools often support the ability of reusing tiles or modules, they would not fit under the tile-authoring category because the existence of a complicated user interface would prevent non-technical users from using them. An example of a tile-authoring tool is the TMT [8] used in the National Advanced Driving Simulator (NADS).

Second- and third-generation tools have dramatically reduced the time it takes to construct driving simulation virtual environments. During a recent development phase of a research study at the NADS, researchers provided a rough specification of the required database to the development group. Based on prior experience using first- and second-generation tools, we estimated it would have taken three to five weeks for a modeler to construct a database that included about 30 miles of highway and suburban roadways and about 20 intersections, one of which was a highway trumpet interchange. This estimate was based on existing libraries of models that included most cultural features and intersections, including the trumpet. Instead designing the database from the ground up, the NADS TMT was used to build it. No new tile development was necessary. It took a modeler slightly less than five days to construct the database, during which time a review by the researchers led to adding another trumpet interchange to the final version. It is also important to note that almost 50% of the total time was spent dealing with various hardware and interface problems of the NADS Image Generator (IG), something that is to be expected for a new IG installation. Similarly, about 20% of the time was spent fixing visual anomalies that could be traced to the original tiles and represent a one-time investment in improving the overall tile quality. Although it is unclear whether this example represents typical timesavings, the TMT resulted in a five-fold reduction in time. We anticipate that, as issues with the IG are resolved and visual tile anomalies are corrected, timesavings will increase by an order of magnitude when necessary tiles are available. For larger databases, timesavings will further increase.

Despite the dramatic reduction of development time made possible by second- and third-generation tools, we have identified three key problems that have a significant impact on the quality of the final visual database. The first problem is the unnatural perfection of the output of the second-generation tool. The second problem is the labor involved in producing greatly varying road geometries. The third problem is the generation of geo-specific databases that use ground-truth road geometry. Each of these problems is discussed below.

Unnatural Perfection

By default, road-generation tools produce road geometries that are completely free from real-life faults, such as pavement cracks and disturbed geometry caused by erroneous construction, ground sinkage, and weather deterioration. Road tools are generally provided with lane and road widths, a lateral profile and curve parameters that may include super-elevation, entry/exit spiral lengths, and radius of curvature. Alternatively, a sequence of 3D points can be provided, and the tool automatically determines the set of parameters to build a road that interpolates the control points. Using mathematical formulas, the tools then produce brutally accurate geometry dressed with a repeated texture perfectly matching the underlying polygons. The final result possesses perfection that is the dream of most drivers and civil engineers, yet is never to be found in reality. Granted, one can always disrupt the output of the tools to add disturbances, either in geometry or in texture, but such work is very labor-intensive, and we have found it extremely difficult to recreate realistic imperfections, especially in geometry. An additional complication is that any changes in geometry necessitate changes to the corresponding correlated road network representation, further adding to the labor involved in this process. Texture disturbances are easier to apply because, after roads are generated, it is reasonably straightforward to replace the texture image in selected parts of the road with a new image containing potholes or other surface markings. The aspect that makes this straightforward is that the tools have already calculated the texture application parameters so the modeler does not have to. The problem of manually correlating the new images still remains, however, especially if the disturbances are expected to produce corresponding cues in other modalities (i.e., a bump when going over a pothole represented on the texture).

Overall, the problem is greatly amplified in high-fidelity simulators that provide high-resolution visuals and rich motion and vibration cues. A visual system that provides increased detail allows the driver to observe texture details and thus expect corresponding motion and/or vibration and audio feedback.

Road Network Variability

Road-generation tools that produce one road segment at a time allow the modeler to specify a segment that is among a known set of segment types (hill, turn, straight, etc.). Although possible, it is very labor-intensive to create roadways that contain numerous variations in turn radius or hills by successively building consecutive segments. Often, it may not be possible to model a particular part of a road if it is not part of the pre-existing segment type. For example, consider a curve whose entry spiral has suffered smooth sinkage, creating a smooth dip over its length. Such disturbances are not infrequent, especially in cold climates where the winter freeze combined with frequent heavy truck traffic continuously taxes the structural integrity of the roads. However, they are either impossible or extremely labor-intensive to create using second-generation tools. Another example is a curve whose radius changes throughout the curve. A tool that supports only constant radius curves cannot readily produce such a road segment.

Geo-specific Ground-truth Roads

Geo-specific databases represent an actual location. Typically, geo-specific databases are constructed by using existing terrain elevation data to construct the surrounding terrain followed by the generation of a road network that matches the two-dimensional profile of the road as extracted from a map or topographic picture. The road can be stitched to the terrain, and textures extracted by photographing the actual location are used to provide very realistic roads. However, the actual three-dimensional geometry of the road is only an estimate based on the surrounding terrain and two-dimensional imagery. For example, for a given curve, top-down photography can be used to determine the radius of curvature but not the super-elevation (if any). Extensive survey of the site can provide such data [9], but generally such activities are too costly and labor-intensive. Similarly, small dips or bumps (in the 0.1- to 0.3-meter range) and higher frequency vertical variations within the 0.1-meter range cannot be recovered through traditional surveys.

ROAD NETWORK SURVEY REQUIREMENTS

Our approach to the aforementioned problems involves the use of a mobile sensor platform that can survey the roadway environment while traveling through it and collect data that can be post-processed to rapidly create a driving simulation database that reflects the surveyed environment. The focus of the work is on the geometry and properties of the road network itself, not the surrounding terrain or cultural features. A few key goals were established early in the project's lifetime:

1. All shortcomings of existing techniques should be addressed by this process.
2. The mobile platform should be able to travel at a reasonable speed, even in the presence of other vehicles, so that surveys can take place without having to close a particular road.
3. The system's accuracy and sampling rate should collect enough information to re-create low spatial frequencies (less than 2.5 Hz) of the road surface.
4. If possible, the system's accuracy and sampling rate should collect enough information to re-create high spatial frequencies (over 2.5 Hz) of the road surface.
5. The resultant databases should be produced reasonably soon after data collection.
6. The resultant databases should be compatible with databases produced by older-generation tools.
7. The overall process should be modular so that additional sensors can be used to augment the surveyed information or improve on the accuracy of other sensors.

Based on these goals, an ambitious research project is underway at the NADS at the University of Iowa to investigate platforms, sensors, and algorithms that, when properly integrated, will allow the construction of geo-specific databases containing roads with ground-truth geometry. There are two key sets of apparatus involved in the survey project. The first includes the various sensors, and the second includes the platform carrying the sensors.

GENERAL APPARATUS PROPERTIES AND CAPABILITIES

Sensors

This section provides an overview of sensors that were considered for the project, focusing on sensor capabilities. Later sections describe how sensors are used within the overall framework of the project.

A key development that facilitated the approach was the availability of dual-frequency Global Positioning System (GPS) receivers that operate in Real-Time Kinematic (RTK) mode. Although a detailed description of the operation of accurate GPS is beyond the scope of this paper, it is important to specify the types of receivers that are necessary to satisfy the project goals. Additional information about GPS is provided by Parkinson et al. [10] and in numerous locations on the World Wide Web. GPS receivers operate by calculating the distance between the receiver's antenna and multiple satellites on known orbits. Once the range to a few satellites (four or more) has been obtained, triangulation is used to calculate the receiver's position. The problem then is in accurately calculating the distance between the receiver and each satellite. This is achieved by detecting the travel time for radio transmissions between the satellite and the receiver. Each satellite transmits a continuous stream of pseudo-random numbers whose transmission is aligned to a specific time. Most commodity GPS receivers use the delay between when the number was transmitted by the satellite and when it was received by the receiver to determine the range, a technique known as Code-Phase. Differential GPS [11] uses two different receivers, one of which broadcasts the error between its actual location (which is provided) and the location it calculates using the satellites. L1/L2 receivers use two receivers to detect the phase shift of the signal's carrier frequency. The carrier's frequency is at 1.56 GHz, yielding a wavelength of approximately 0.2 meters, so a receiver that can match the carrier frequency to 10% will give a less than 2-cm accuracy. Combined with a differential correction signal, this accuracy can be improved even further. Such receivers operating in RTK mode can deliver accurate position samples one or more times per second. Using two such receivers on the platform can provide information that can be used to recreate the road's geometry with very good accuracy.

Cameras are the second type of sensors used for the project. Cameras have multiple uses, including the detection of lane deviation or the capture of images that can be used to extract textures. Cameras are also used to capture forward-looking footage used for documentation purposes. Various algorithms also exist that use camera data to extract road elevation information, although there are no immediate plans to use cameras for that purpose on the project. Whenever possible, it is important to use cameras that can be synchronized to an external source so as to maintain time correlation with the remainder of the surveyed data.

Distance measuring equipment (DME) is another type of sensor that is used in the project. DME provides the distance along a line between a point on the vehicle and the road. Both laser- and sonar-based DME equipment was considered. Generally, laser-based DME equipment is more expensive but also more accurate and can produce more samples per second than sonar-based DME. As a result, laser-based DME is used when high accuracy is required or when multiple measurements (multiple thousands per second) are required. Lasers that measure time of travel of either direct reflection or diffuse reflection are available, but with different constraints on the actual laser and return light sensor placement. Lasers can easily provide an accuracy of less than 1 millimeter. Sonar-based DME, due to lower cost, can be used to create sensor arrays that can simultaneously sample multiple paths in parallel, in fairly high-density configurations. However, cross talk across sensors can be a problem, so sonar pings must be serialized in time to reduce the likelihood of cross talk.

Accelerometers and rate gyros were also considered for the project. Usually, accelerometers and rate gyros are used to extrapolate the position of the vehicle among sample points. When appropriately post-processed, linear and rotational accelerometers can provide a more detailed trajectory than low frequency samples obtained by a GPS receiver or other positioning sensor. However, this requires a double integration operation, which causes the accumulation of errors as time elapses. We decided that for the first phase of the project, accelerometers would not be used for augmenting position information, primarily because of the ability of RTK GPS receivers to provide samples frequently enough to re-create the low spatial frequencies. Properly placed accelerometers can still be used, however, to capture the higher spatial frequencies that cannot be captured by GPS alone.

The final sensor considered for the project was sweeping laser-based DME equipment. In this type of sensor, a laser beam is reflected over a rotating mirror that allows multiple readings along a lateral section of the platform. Such a sensor mounted slightly ahead of the platform, scanning to a direction lateral to the motion of the platform, can capture the depth profile of the entire road surface. Sweeping laser sensors are rather costly, and their use is still under investigation for this project.

Platform

Although airborne platforms were considered, we decided to proceed with a ground vehicle for the initial aspects of the work, primarily because of cost and risk factors. Therefore, although an airborne sensor has not been ruled out for the future, most of the work has focused on mounting various sensors on a ground vehicle. The initial feasibility study used an instrumented passenger vehicle with the GPS sensors mounted on the roof rack. Figure 1 illustrates two views of the vehicle. The picture on the left illustrates the mounting hardware that can be used for mounting a sole DME sensor and a downward-looking camera with visible access to the front of the vehicle. The picture on the right shows the mounting of the GPS and radio antennas for receiving the differential GPS correction. One problem with using this vehicle was the requirement for three people inside the car during data collection: one to drive, one to operate the GPS, and one to operate the remaining electronic data collection equipment. For future studies, we will use a larger vehicle with a dedicated control station that will allow controlling of all onboard sensors from a single station. Another problem is the lack of enough space for all the equipment necessary to perform the survey, especially as additional sensors are utilized. A final vehicle decision has not been made at this point.

FIGURE 1 Two views of the instrumented vehicle used for the first data collection.

THE OVERALL SURVEY AND POST-PROCESSING PROCESS

After extensive research and consideration of the available sensors, their cost, and their cost-benefit ratios, we formulated an overall survey process that depends on a pair of RTK L1/L2 GPS receivers as the key position sensor and cameras for lane placement, texture acquisition, and documentation purposes. The process is designed so it can easily be extended to incorporate downward-looking laser DME sensors for correcting vehicle suspension effects, accelerometers for high-frequency spatial frequency capture, and, eventually, sweeping laser DME sensors for lateral road profile capture. However, the focus of the initial work was to formulate a process that can be used to determine the feasibility of the overall survey approach. Figure 2 illustrates the block diagram of the initial survey process.

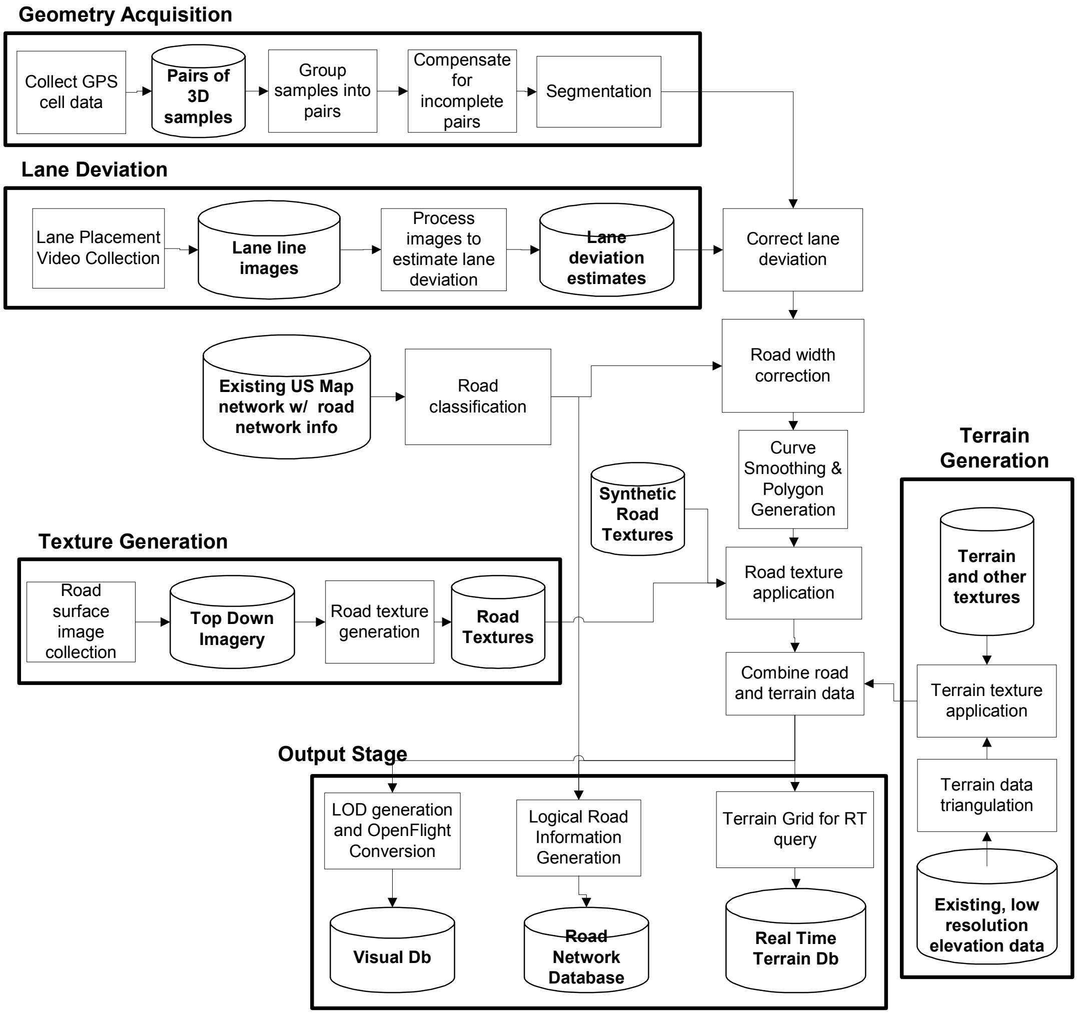

FIGURE 2 Ground Truth Acquisition System Block Diagram.

In summary, the process involves acquiring continuous video for lane placement correction and documentation purposes, and GPS sample points for road geometry reconstruction. After data collection, various post-processing steps are required to compensate for shortcomings in the data collection process introduced by human driving, vehicle suspension effects, and lack of additional information needed to fully describe the road network. The eventual output of this process is a set of correlated virtual environment databases that can be used for driving simulation applications.

The first data collection was performed on March 18, 2000. Three test runs were performed and have been used to develop and test the post-processing software. The actual equipment used was a pair of Trimble 4700 Receivers mounted on the vehicle as shown in Figure 1. An additional Trimble 4700 stationary receiver was used to broadcast the necessary differential correction.

The remainder of this section describes each processing step in more detail, along with the lessons learned in utilizing the equipment.

Collect GPS Cell Data

This process involves the acquisition of the GPS data points. As the vehicle drives through the road to be surveyed, the GPS receivers collect data at 1 Hz. The rated accuracy of the Trimble 4700 receivers is ±1 cm horizontally or ± 2cm vertically with 1 part per million (ppm) variation. This accuracy was validated by looking at the variation of the distance between the two GPS coordinates. Since the antennas were mounted at a fixed distance from each other, the Cartesian distance between pairs of GPS samples should be identical to the real distance. Any variation provides a measure of the relative error between the two receivers. In fact, it is the relative error that we care most about because the absolute error does not make any perceived difference on the road geometry. TABLE 1 contains statistical information on the distance between corresponding samples as computed by the GPS samples. The maximum error row is calculated for each run, assuming that the actual distance is the average of the sampled distances. The eight runs represent eight distinct data collection intervals taken over a two-day period. Each run consists of several hundred pairs of points. All units are in meters.

TABLE 1 GPS Relative Error Analysis

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8
Min Dist	1.147	1.139	1.155	1.147	1.157	1.156	1.162	1.156
Max Dist	1.186	1.181	1.180	1.191	1.180	1.177	1.175	1.189
MeanDist	1.163	1.165	1.164	1.167	1.165	1.168	1.168	1.169
StdDev	0.00526	0.005255	0.00608	0.00546	0.00496	0.00379	0.00416	0.00454
MaxError	0.023	0.026	0.016	0.025	0.015	0.011	0.007	0.019

As shown in the table, the relative error between the two receivers is significantly lower than the rated absolute error. In fact, the error is so small as to be considered noise when compared with other errors induced by the vehicle's suspension. This is consistent with GPS operation if one considers that the majority of errors are caused by atmospheric effects that affect both GPS receivers consistently. The maximum error variation across runs was harder to explain until we observed that often when driving by buildings or dense forest areas one receiver would lose complete synchronization while the other operated fine. Overall, however, the relative accuracy of the GPS samples was impressive and validated the premise that these units can be used for very high-resolution roadway surveys.

There were, however, several challenges. One was the simultaneous initiation of data collection for both receivers. It takes a significant number of keystrokes on the portable control unit to set up the GPS receiver to begin collecting. Once it begins collecting, it does so at regular 1-second intervals. However, we required both receivers to collect at the same time instance. Unfortunately, given the time constraints, we had no way to initiate collection on both receivers at exactly the same time other than by pressing the "Start" buttons on both control units simultaneously. After a bit of practice, it was possible to start both receivers with a less than 0.05-second drift (as verified by looking at the timestamps of the samples). In the future, however, it will be necessary to automate this process.

Other problems that occurred during data collection were dropped points and loss of signal. It is well known that GPS requires line-of-sight access to enough satellites, and as a result, it cannot be used under thick tree canopy or among multiple tall buildings. However, L1/L2 receivers are more sensitive than Code-Phase receivers and are more likely to lose the signal due to obstructions. During collection, we observed two types of signal-loss-induced failure modes: single point drop and complete loss of signal. In single point drop, either or both GPS signals would skip logging a data point for one or two samples but continue immediately after that. Single point drops did not require stopping data collection because the receivers continued logging data after the drops. When a complete loss of signal took place on either receiver, tracking of the satellites was lost and the receiver quit collecting data. This necessitated stopping the vehicle and remaining stationary until the receivers reacquired the satellites. This process took as little as 30 seconds or as long as 5 minutes, depending on time of day and location.

A final issue that needs to be addressed in the future is compensation for the effects of vehicle suspension on the acquired geometry. During turns, braking, or acceleration, or when the road is not perfectly flat, added 3D displacements are introduced to the sampled data, caused by the motion of the vehicle chassis relative to the ground because of the vehicle suspension. Depending on the vehicle, such displacement can be as high as 0.2 meters, something that is an order of magnitude higher than the GPS error. In the first study, we ignored this error; however, it is clear that in the future there needs to be compensation for this error. One potential approach to compensating for suspension motion is to use a DME sensor that measures the distance between the GPS antenna and the road surface. Since most vehicle motion is along the vertical axis, such a measurement could offset the majority of suspension-induced errors.

Group Samples into Pairs

Each GPS receiver stores its data on a separate portable data-logging collector, assigning sequential numeric tags to each sample. The tags, which are user-selectable at startup, allow an arbitrary prefix to be assigned to each point so the tags may look like this: Left001, Left002, Left003, etc. for the left unit and Right001, Right002, Right003, etc. for the right unit. The goal of this step is to correlate the tagged points of one receiver with the corresponding points on the other receiver. As long as both receivers do not drop any points, then correlating the left and right channels is simply a matter of correlating the numbers of both samples. However, the GPS receivers operate independently of one another, without any real-time check to ensure that the points are properly paired. So, for example, Left005 may correspond to Right006 if the right receiver dropped a sample. This process compensates by checking the distance between sampled points on the two receivers and then pairing the appropriate samples together.

Compensate for Incomplete Pairs

After grouping samples in pairs, this step compensates for missing data by filling in the points. The points before and after the missing point(s) are used to create a cubic spline whose perpendicular distance to the matching point is equal to the known distance between the two GPS receiver antennas. Figure 3 illustrates an example of how missing points are filled. In the figure, missing points are indicated with a hollow point.

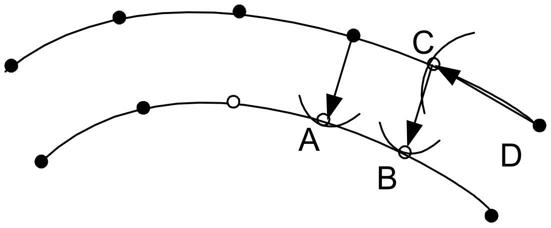

FIGURE 3 Missing Sample Point Recovery Illustration.

In cases where both points in a pair are missing (as are B and C in Figure 3), prior points on the same side (in this example, D) are used to extrapolate the position of the missing point, which then can be used to fill in the position of the other point in the pair. In the data obtained during data collection, this technique didn't have to be used at all because no pairs of points were lost; instead, most dropped points were in isolation. Figure 4 illustrates sample points drawn from a top-down view from one of the actual data collection runs. Note that in this example, only two points are dropped on the left side of the picture, one on each side.

Segmentation

Data collection proceeds while the vehicle is driving through roads and intersections. Also, a vehicle will often have to drive over prior paths (as shown in Figure 4) when navigating. The purpose of the segmentation process is to isolate segments of data samples that belong to a road segment between two intersections. This allows the use of the samples for road geometry alone and provides a clean separation between the issues involved in surveying roads and the issues involved in surveying intersections.

At this point, segmentation is implemented manually. We are investigating the use of a graphical tool to allow interactive selection of segments.

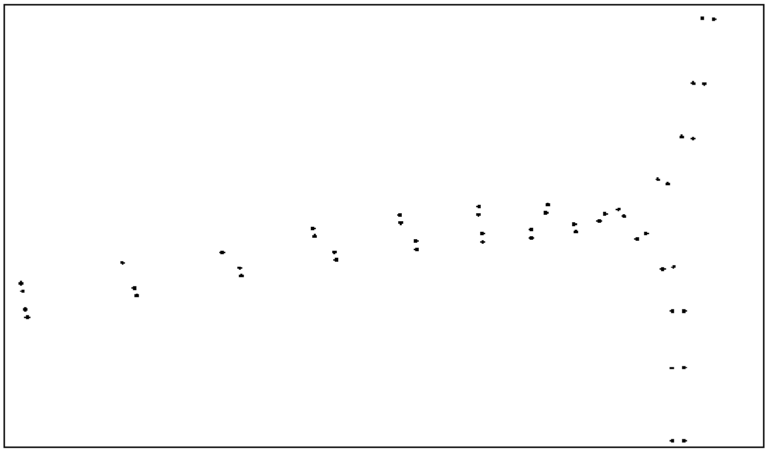

FIGURE 4 Actual collected points illustrating point drops.

Lane Placement Video Collection

This step is necessary to compensate for the fact that the driver of the survey vehicle does not generally maintain a fixed lane placement. This is either because of human error or because the vehicle has to veer to avoid stationary objects (i.e., a parked car). As a result, the collected data do not accurately reflect the road geometry. To remedy this problem, a camera mounted on the passenger side of the car records digital images of the lane line. These images can later be used to ascertain how far the driver drifted from the side of the road. Video data collection did not pose any major problems.

Process Images to Estimate Lane Deviation

This step uses still images extracted from the passenger-side video stream to detect the lane deviation. This information can then be used to correct the collected data by providing a lane deviation number corresponding to the time each pair of samples was collected. At this point, no work has been done toward this goal, but it is part of the planned future work.

Correct Lane Deviation

This processing step is responsible for using the lane deviation correction produced in the prior step to correct the acquired 3D samples. Figure 5 graphically illustrates this correction process.

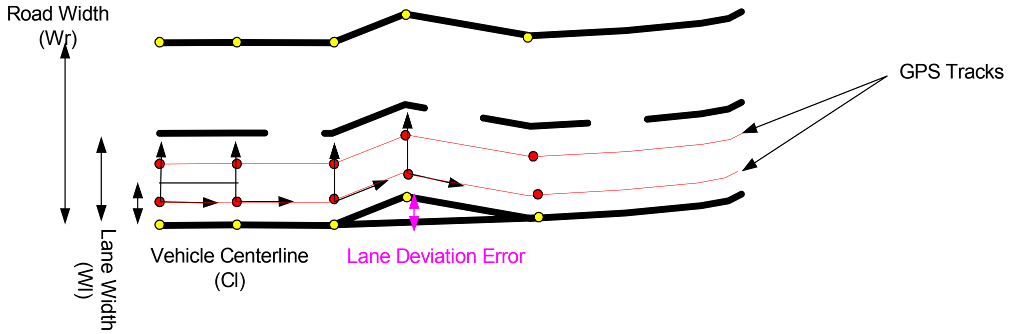

FIGURE 5 Lane Deviation Correction.

A point near the beginning of data collection is used to establish a baseline lane deviation. From that point forward, a lane deviation error is computed for each sample point, and that error is used to laterally offset the pair of sample points along the direction of a line formed by these two points. Although no work was done on the automatic lane deviation estimation extraction, some data was coded manually to test the correction algorithm, which appears to be working as expected. However, unless the automated extraction process completes, it is not possible to estimate the error sensitivity of this approach to erratic driving.

Road Classification

The increasing use of GPS receivers for in-vehicle navigation has caused several vendors and organizations to develop highly accurate road network databases that can provide in real time information about the current road, the type of road, and its connectivity to adjacent roads through intersections. Whereas the resolution of the geometry is not high enough to be used for this project, the attribute information about a road is useful in that it can be used to determine the approximate width of the road, the number and direction of lanes, and numerous other logical attributes such as speed limit that can be used for autonomous driving applications. This processing step uses the GPS signal to query geographical databases and obtain various pieces of information. At this point, the primary interest lies in obtaining the total width of the road and the number and direction of lanes.

Road Width Correction

Since the GPS receivers were mounted on the top of the car a fixed distance apart, the data at this point represents lanes that are approximately two meters wide. Using the road width information obtained from the previous step, the road can be widened to the correct width by extending the sample points along the line crossing both samples. A key assumption that facilitates this calculation is that the survey vehicle is always driving on the right-most lane of the road.

Curve Smoothing and Polygon Generation

The GPS receivers collect samples at fixed time intervals, independent of the speed of the survey vehicle. As a result, the spacing between successive points varies depending on the speed of the vehicle. This presents a problem regarding the roughness of road geometry created by simply triangulating the sampled points. If too few samples are taken as the car corners a tight curve, the road reflected in the database appears “blocky.” To some degree, this problem can be addressed by driving slower; however, when driving among traffic there is a minimum speed that has to be maintained to ensure safety. An alternative is to have the GPS receivers sample faster. The GPS receiver used for this study had a maximum sample rate of 5 Hz; however, we only managed to configure it for collection at 1 Hz. According to vendor specifications, a newer receiver, the Trimble 5700, has a maximum sampling rate of 10 Hz, and future plans for this project include procuring the newer receiver and sampling data at 10 Hz. Even at that rate, however, moderate speeds may yield points that produce visibly discontinuous road geometry.

To alleviate this problem, a cubic interpolating spline is fit along the collected points. Additional points on the spline are computed as follows: the tangent vector is calculated at each control point and the angle between consecutive tangent vectors is measured. If this angle is greater than a threshold angle, an extra point pair is added between the two control points. The process repeats until no angle differential between successive points exceeds the threshold. Following that, triangles are generated to represent the road surface.

Figure 6 illustrates the effect of adding new control points on a road network. The road produced by utilizing the sampled points alone is shown on the left, whereas the road produced after augmenting the original data with additional points along the spline is shown on the right.

Road Surface Image Collection and Texture Generation

One of the most ambitious components of this project is the automatic extraction of roadway textures based on actual imagery. To achieve this, we plan to collect video footage from a camera facing straight toward the road pavement. Individual images can then be processed to obtain varying textures that reflect the actual roadway. No work has been done on this process yet, but one of the anticipated challenges is mounting the camera as high as possible, yet ahead of the vehicle so as to minimize warping of the image. At the same time, it is very difficult to create structural supports, and there are safety issues to contend with when driving a vehicle with equipment extruding far beyond its approximate bounding box.

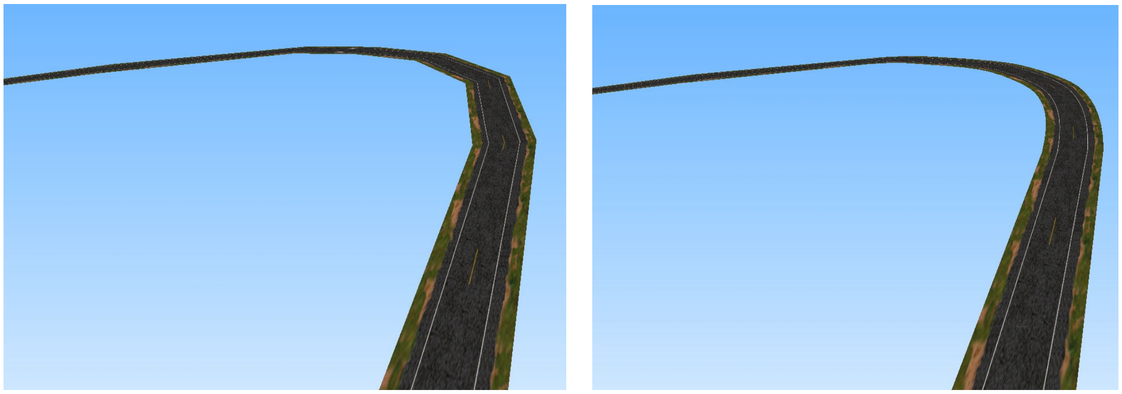

FIGURE 6 Illustration of smoothing sampled points.

Road Texture Application

Independent of the techniques used to create the textures, they must be properly applied on the road polygons. Since the road polygons vary in size and shape, a texture application scaling factor must be calculated for each road polygon. This step also calculates an application offset to ensure that textures blend seamlessly over adjacent polygons. Figure 7 illustrates the road polygons before and after application of a road texture. Note that for this particular illustration, an existing texture was used, not one obtained from the actualThis road texture was created by hand.

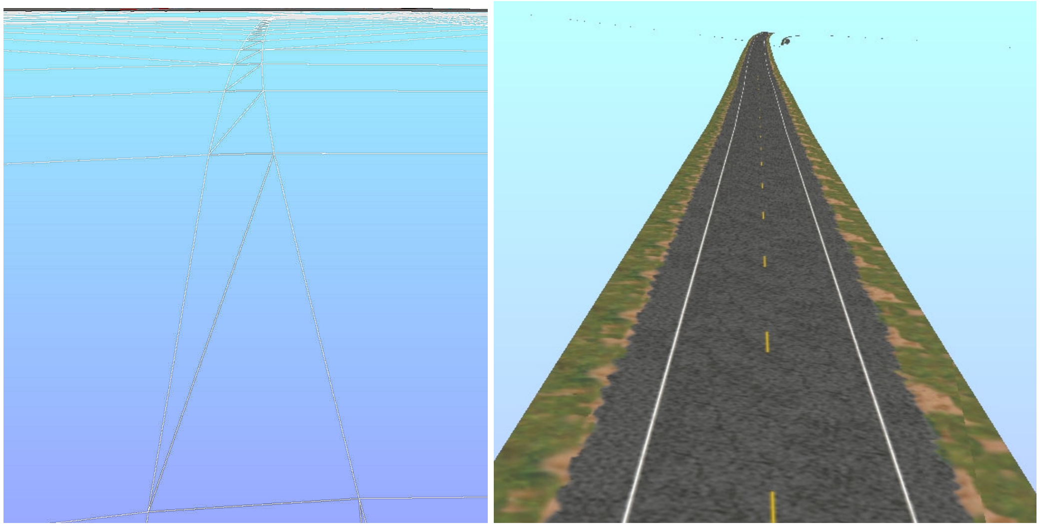

FIGURE 7 Texture application illustration.

Terrain Generation

In order to be usable, a driving simulation database cannot only consist of road network visual elements; at a minimum, terrain and cultural features are necessary. The terrain-generation process is responsible for producing terrain information that can be merged with the high-resolution road network and provide a usable final database. Extensive handling of terrain is not a major goal of this project, and in fact, there are several commercial tools [4,5] that do an excellent job of generating terrain polygons. Nevertheless, it is useful to have the ability to produce a usable database that contains terrain without having to use commercial tools. To support this goal, a simple process was devised to obtain existing geo-specific elevation data that can be triangulated to produce a set of polygons that can be combined with the road network. The developed module utilizes grid data to produce triangles representing the terrain. These triangles can then be stitched with the road network to provide a seamless geometrical transition between the road and the terrain. Following the stitching step, externally specified textures can be applied to the terrain polygons. Figure 8 illustrates the stitching process. The illustration on the left shows the road polygons and overlapping terrain polygons. The illustration on the right shows how the terrain polygons are re-triangulated to match the edge of the road network geometry. This approach does increase the number of polygons in the final scene, but it is especially useful for Z-buffered image generators because it eliminates hidden pixels that have to be processed by the Z-buffer algorithm.

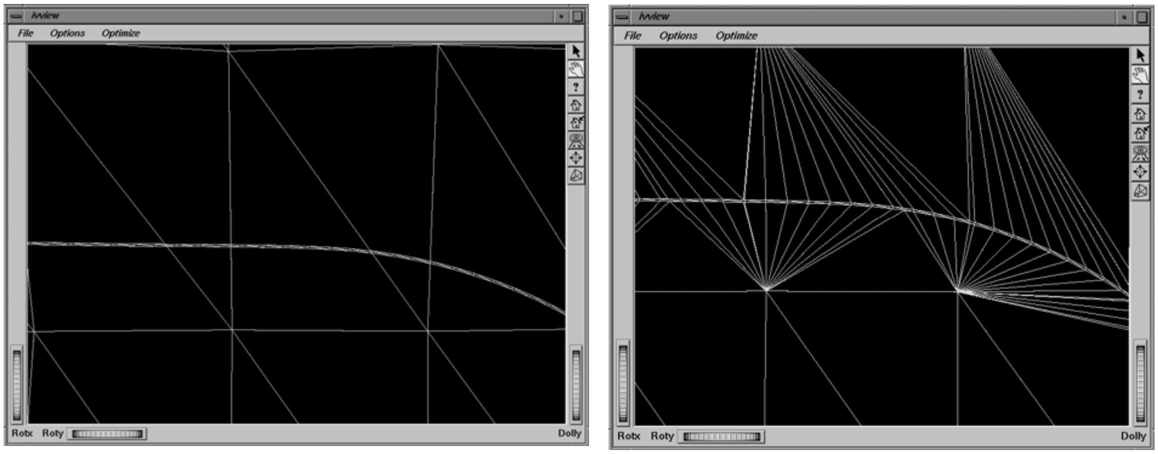

FIGURE 8 Effect of terrain road stitching.

Figure 9 depicts the same geometry using a 3D perspective and with textures applied to both the road network and the terrain.

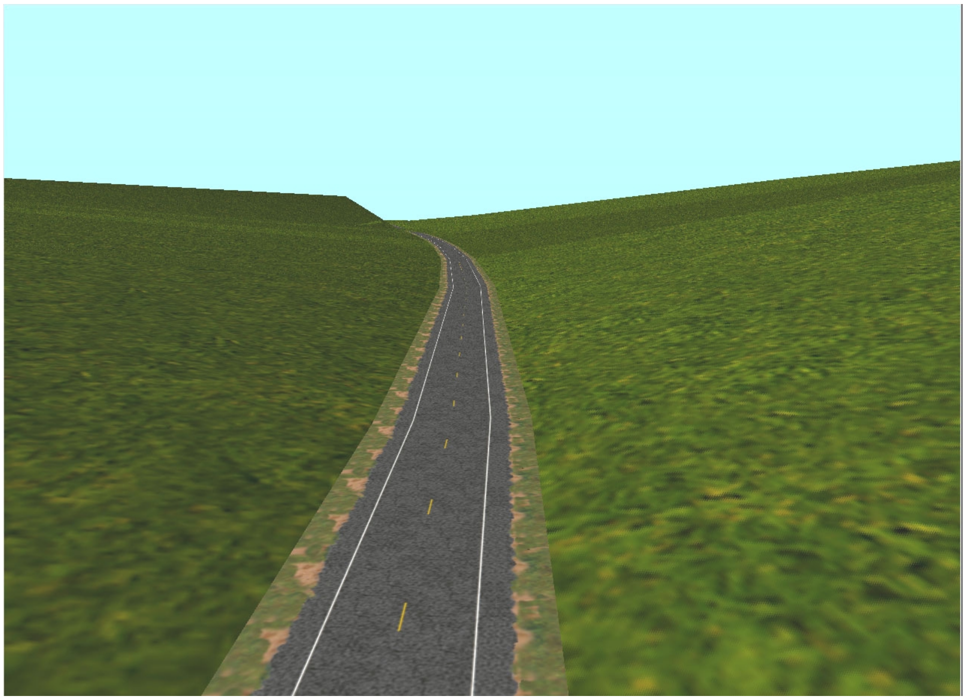

FIGURE 9 Textured and stitched road and terrain.

Output Stage

During the output stage of the process, the various data are combined and translated into the necessary real-time correlated databases necessary for driving simulation applications. Without loss of generality, we have focused on databases whose format matches the requirements of the NADS. Specifically, the visual database is produced in OpenFlight® format. This real-time graphical format allows the use of level of detail (LOD) nodes that help manage the processing load during rendering. In fact, without effective LOD, databases that are too large for the IG to render in real-time are unusable. The LOD generation process separates the overall database into square regions, and within each region it sub-samples the road network points and re-stitches the terrain. Lower polygon count versions are used for the low LOD while maintaining consistent matching between LOD regions.

In addition, the code produces a Logical Road Information (LRI) format [6] that is usable by the NADS scenario definition and control software. Finally, to support off-road applications, the code can also produce data usable by the NADS off-road terrain interrogation software [11].

CONCLUSION AND FUTURE WORK

This paper described a process for producing geo-specific driving simulation databases from data obtained from surveying the actual road with sensors mounted on a vehicle. An overall process was defined to utilize various sensors and how to combine the sensor data to produce the final outputs. An initial feasibility study was described where a subset of the process was exercised with very encouraging results.

Future work on this project includes the procurement of latest-generation GPS receivers and the incorporation of DME equipment to help compensate for suspension effects introduced in the sampled data, as well as further development of algorithms to extract additional information from the data, such as lane deviation, lateral profile, and high-frequency spatial content of the road.

ACKNOWLEDGMENTS

The authors would like to acknowledge the work of several individuals who have directly or indirectly contributed to the project. The initial data collection and development of the early prototype code was done by Ben Wehrle as part of his senior design project. Professor Tom Schnell of the University of Iowa Department of Industrial Engineering provided access to his instrumented vehicle for the first data collection. Shawn Allen and his group have often provided consulting advice on database modeling issues.

This research was also partially supported by the US Army Tank Automotive Research Center through the Automotive Research Center (Department of Defense contract number DAAE07-94-C-R094).

REFERENCES

1. Evans, D.F. Correlated Database Generation for Driving Simulators. *Proceedings of the 1992 IMAGE IV Conference* (pp. 353-361). The IMAGE Society, Tempe, Arizona, 1992, pp. 353-361

2. Papelis, Y.E., and S. Bahauddin. Development of Domain Specific Correlated Databases Using Parametrized Tiles. *Proceedings of the 1998 IMAGE Conference*. The IMAGE Society, Scottsdale, Arizona, 1998.

3. *Multigen OpenFlight Scene Description Language Format Specification*, Multigen-Paradigm, Inc., 2000.

4. *Multigen Creator User's Manual*. Multigen-Paradigm, Inc., Addison, Texas, 2000.

5. *TERREX TerraVista User's Manual*. Terrain Experts Inc., Tucson, Arizona, 2000.

6. Papelis, Y.E., and S. Bahauddin. (1995). Logical Modeling of Roadway Environments to Support Real-Time Simulation of Autonomous Traffic. *First Workshop on Simulation and Interaction in Virtual Environments*. The University of Iowa, Iowa City, Iowa, 1995, pp. 62-71.

7. Bayarri, S., M. Fernandez, M. Perez, and R. Rodriguez. A Database Model for Urban and Motorway Driving Simulation. *Proceedings of Driving Simulation Conference*, Sophia Antipolis, France, 1995, pp. 286-301.

8. *Tile Mosaic Tool User's Manual*, Evans & Sutherland Computer Corporation, NADS documentation library.

9. Allen, S., and C. Bruse. Development of Microscopic Terrain Surfaces and Features for Driving Simulation Environments, *Proceedings of Driving Simulation Conference*, 2001 (to appear).

10. Parkinson, B., et al. (ed.). Global Positioning System: Theory and Applications, Vol. II, Chapter 2.

11. W.I. Bertiger, Y.E. Bar-Sever, B.J. Haines, B.A. Iijima, S.M. Lichten, U.J. Lindqwister, A.J. Mannucci, R.J. Muellerschoen, T.N. Munson, A.W. Moore, L.J. Romans, B.D. Wilson, S.C. Wu, T.P. Yunck, G. Piesinger, and M. Whitehead. A Real-Time Wide Area Differential GPS System. Invited contribution to *Institute of Navigation "Red Books," Global Positioning System: Papers Published in Navigation,* Vol. IV, 1999, pp. 145-162.

12. Papelis, Y.E., S. Allen, B. Wehrle. *Automatic Correlated Terrain Database Generation and Management for Ground Vehicle Simulators*. Proceedings of the American Institute of Aeronautics and Astronautics Conference, Inc., Portland, Oregon, 1999.