3D visualization for improved manipulation and mobility in EOD and combat engineering applications

Joel Alberts, John Edwards, Josh Johnston, Jeff Ferrin
Autonomous Solutions, Inc.
990 North 8000 West,
Petersboro, Utah 84325

ABSTRACT

This paper presents a scalable modeling technique that displays 3D data from a priori and real-time sensors developed by Autonomous Solutions under contract with NAVEODTECHDIV and TARDEC. A novel algorithm provides structure and texture to 3D point clouds while an octree repository management technique scales level of detail for seamless zooming from kilometer to centimeter scales. This immersive 3D environment enables direct measurement of absolute size, automated manipulator placement, and indication of unique world coordinates for navigation. Since a priori data is updated by new information collected with stereovision and lidar sensors, high accuracy pose is not a requirement.

Keywords: EOD, 3D, Visualization, Manipulation, World building

1. INTRODUCTION

Improving the quality of the human-machine interface is imperative for increasing the effectiveness of military robots. Current interfaces operate in 2D, based on either video camera feeds or top-down map displays. Video interfaces do not present depth, range, or scale information and map displays require a priori data and accurate absolute positioning systems.

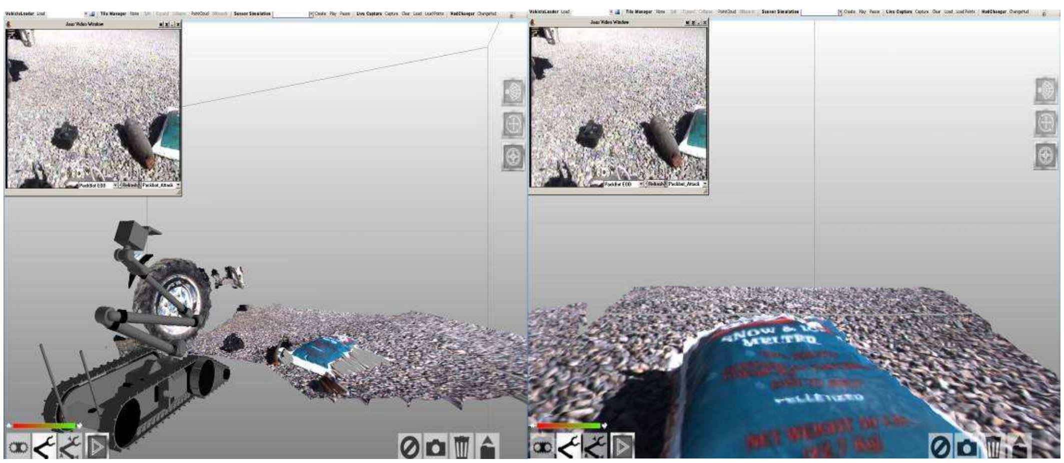


Figure 1: 3D rendering of a gravel surface and Improvised Explosive Device (IED).

This paper presents a scalable modeling technique that displays 3D data from a priori and real-time sensors. A novel algorithm provides structure and texture to 3D point clouds while an octree repository management technique scales level of detail for seamless zooming from kilometer to centimeter scales. This immersive 3D environment enables direct measurement of absolute size, automated manipulator placement, and indication of unique world coordinates for navigation. Since a priori data is updated by new information collected with stereovision and lidar sensors, high accuracy pose is not a requirement.

The interface consists of a 3D colorized model and a virtual camera viewpoint controlled by the operator. A simple point selection method lets the operator select unique world points without the problems of ambiguity, occlusion, and robustness inherent to visual servoing and 2D techniques. Because the selected points are exact world coordinates, they define waypoints for complex paths rather than simple bee-lines. Depth is known, so it is possible to place a manipulator at the correct location with desired position, pose, and joint posture.

2. ARCHITECTURE

Autonomous Solutions (ASI) uses a layered software architecture shown as Figure 2 which translates the raw data from an arbitrary set of 3D capable sensors into a 3D world model through a standard interface. This 3D world model is available to an arbitrary set of high level autonomy applications through another standard interface. In this way, perception sensors and autonomous software applications are modular which enables rapid development. This software package is referred to as TruSpective because of its goal of providing users with a real perspective on robot operation.

The lowest architecture layer contains the raw sensors available for robotic perception, like stereovision, planar lidar, and flash lidar. These sensors provide data with different coordinate frames, formats, densities, update rates, and other parameters. The next higher layer, the transformation layer, translates and rotates the data from each sensor into a common format, such as an *xyz* point cloud. The model layer contains the most recently updated full model available from the sensor data in a standard format. It is accessed by a variety of applications which implement autonomous behaviors, such as click and go driving.

The benefits of this approach are:

- Any set of 3D capable sensors can be used to collaboratively create a single 3D world model
- Common control and autonomy applications (such as navigation algorithms) can be made independent of the particular sensor set used
- Both sensors and autonomy applications, being independent of each other, can be replaced as technology is improved or as applications require.
- The model/render layers can exist off-robot or on an Operator Control Unit (OCU) which minimizes requirements for the vehicle processor.
- Sending compressed 3D points and still camera images across a radio link requires less bandwidth than a video channel.

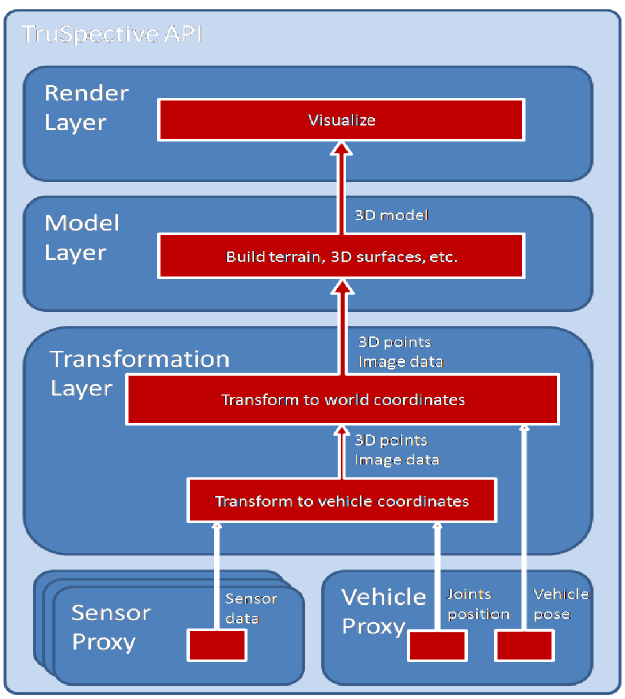


Figure 2. ASI's TruSpective 3D Architecture

The TruSpective software is built in a layered manner so as to simplify usage and extension. There are 3 primary layers: transformation, modeling and rendering. Sensor and vehicle plug-in proxies provide support for data acquisition and platform description.

- Vehicle Proxy: The vehicle proxy provides information necessary for rendering a model of the vehicle inside the 3D world and transforming sensor data into world coordinates.
    - Vehicle model: The model of the vehicle is defined with a DirectX .X file. This provides flexible support for virtually any vehicle configuration and rendering.
    - Joints position: Assuming the vehicle has a manipulator or other moveable joints; the vehicle proxy is responsible for providing updates of any changes to joint position. This enables the transformation software component to convert sensor data into the vehicle coordinate frame.
    - Vehicle pose: The vehicle proxy is responsible for providing updates of robot vehicle position. This enables the transformation software component to convert sensor data into the world coordinate frame.
- Sensor Proxy: The sensor proxy provides sensor data to the application. It queries the sensors for data, retrieves data and makes it available for consumption by the application.
- Transformation Layer: The transformation layer takes inputs from the vehicle and sensor proxies and transforms the sensor data first into vehicle coordinates and then into world coordinates.
- Model Layer: Once the data has been transformed into world coordinates it can be used by the model layer to generate a variety of 3D models to visualize the data in many different ways. The model layer currently has a suite of built-in model builders including simple point cloud, 3D triangulation, terrain triangulation and billboard. Decimation techniques include octrees, and other parameters are available to tune the classic speed vs. quality tradeoff.
- Render Layer: The render layer displays the 3D model to the display device. TruSpective currently uses DirectX for rendering.

3. 3D VISUALIZATION

Autonomous Solutions has developed a suite of sensors and software to enable an EOD technician to have a real-time three-dimensional (3D) view of the target environment and the robot's position and orientation in it (Figures 1,3,4) [1]. This view can be manipulated by the user at video frame-rates to observe the robot and its environment from any angle and distance.

After the 3D model of the world is generated, the rendered robot position can be updated using information from a pose source like GPS or odometry. This telemetry consists of much less data than streaming video but still enables teleoperation of the vehicle, including obstacle avoidance. Therefore, the 3D modeling is a way to continue operating the vehicle despite lapses in high bandwidth radio communication. When extra bandwidth is available, the 3D model can be extended and updated.

ASI's research has identified several methods of displaying texture on 3D models.

3.1 Point Cloud Visualization

The simplest and fastest 3D visualization method uses point sprite particle visualization. Each 3D point returned from the sensors is assigned a color. The color is either passed with the point if a stereo vision camera is used, or is assigned using mapping from a single camera image if the point contains no inherent color. Each sprite is rendered with its one color, regardless of the size at which it is rendered on the screen. Using sprites, a point cloud of up to and exceeding one million points can be visualized at video frame-rates.

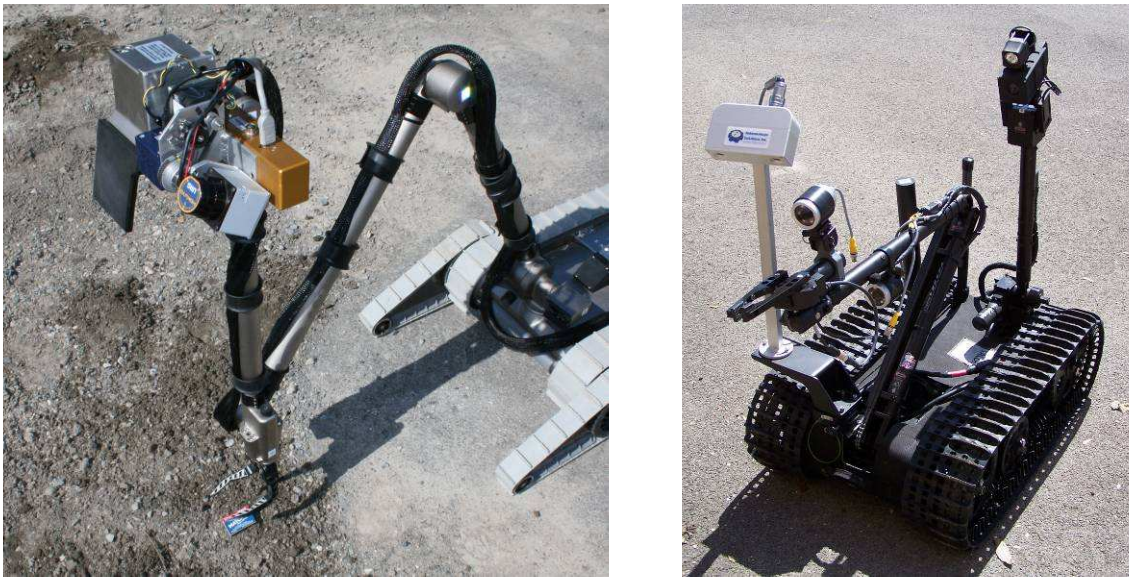


Figure 3: ASI's PackBot modified for 3D visualization. ASI's stereovision system mounted to the TALON robot using a quick-release bracket (silver camera on left of image).

The point cloud visualization is effective when the 3D data density is about the same as the texture resolution and the operator doesn't need to zoom too close to the surface. Additionally, if the sensor data is updated frequently, such as is the case with lidar sensors capturing data at rates of 100,000 points per second or more, then the point cloud visualization method gives the best capture-to-display rate.

When visualizing the data from a distance, the point sprite method effectively approximates a surface visualization. When viewing at large scales, however, the individual points begin to visibly resolve.

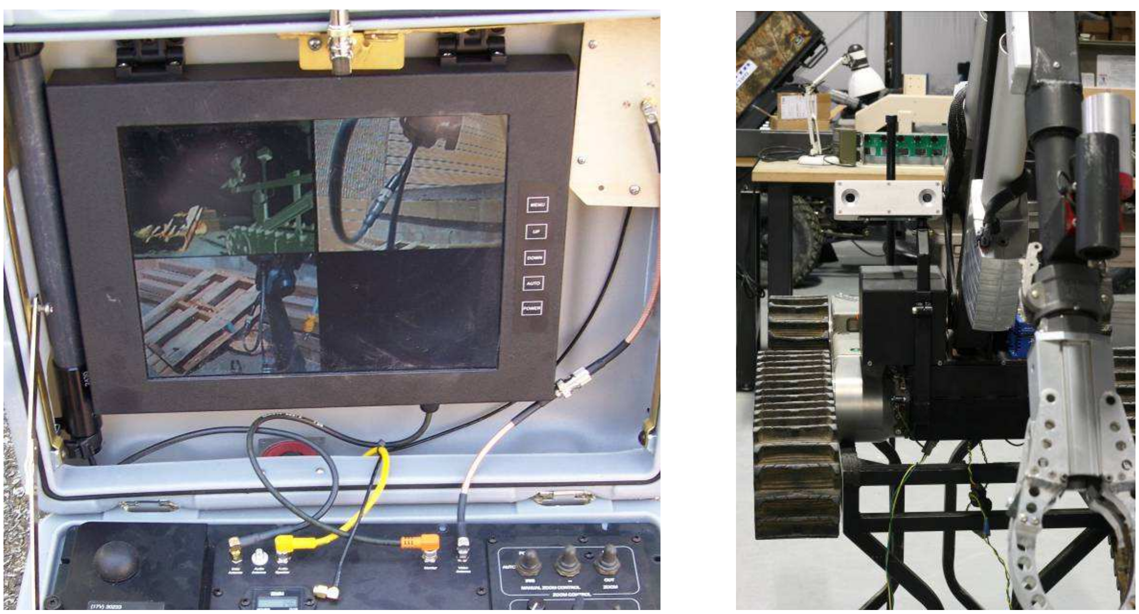


Figure 4: ASI's 3D visualization on a TALON OCU (upper left quad view). Right: 3D system on ASI Chaos platform.

3.2 Triangulation

ASI has developed a novel triangle surface generation method that more closely approximates the physical surfaces being sensed without sacrificing frame rate. When the 3D points are captured by the sensors, triangles are generated using an optimized raster-walking algorithm. Both color and spatial coordinates are used as arguments to the decision function which determines whether sequences of three points lie on a physical surface or not. The triangles are displayed with a two-dimensional (2D) color raster image projected onto them. Once the triangles are generated and initially displayed, the visualization system can easily display them at interactive frame-rates. The triangulation method displays triangles at near frame-rate speed after the initial calculations. It is optimal for cases in which a higher quality display image is desired or when sensor data is not being collected at high rates.

3.3 Quad Visualization

The third visualization option is similar to point cloud display, but uses quads (rectangular surfaces) with image projection. Using this technique, each quad displays a piece of the image, rather than a single color as with the point cloud technique.

As computation of each quad's texture is relatively expensive, a level of detail (LOD) control using octrees limits the number of quads shown at a given time. The quad visualization method approximates a surface display when viewed at a distance. With low resolution 3D points, this surface can look better than point clouds since the quad size can be increased without significantly increasing the "blocky" look of the data. However, the cost of display, even using decimated data, is higher than that of point clouds. The capture-to-display rate is better using quads than triangles, but the subsequent interactive frame-rate of quad display is significantly lower than triangles after the overhead triangulation step is complete.

3.4 GPU Utilization for Faster Processing

In order to render at frame rate speeds, much of the custom graphics processing is delegated to the graphics card using High Level Shader Language (HLSL). Shader support is implemented primarily for projection support, that is, projection of images onto arbitrary surfaces, including point clouds, triangles and quads. Shader projection supports both standard camera images and also spherical camera images, such as those taken by the Point Grey Ladybug2 camera. Adding this projection code into the shader, to be executed in the graphics pipeline on off-the-shelf graphics hardware, improves performance by an order of magnitude, enabling real-time rendering of captured sensor data. The shader source code is implemented so as to be compatible with a variety of standard graphics cards.

3.5 Repository (2.5D, 3D, Octrees, LOD)

Quad trees are an effective way of spatially indexing data [2]. Points are recursively subdivided into a tree of quadrants such that a tree node refers to a spatially-coherent set of points. The quad tree generation algorithm subdivides only areas of high point density as can be seen in Figure 5(a). Quad trees naturally lend themselves to level-of-detail and decimation. Choosing a level *n* in the quad tree yields a particular resolution. All nodes at level *n* and all leaf nodes at levels less than *n* are returned to give a spatially representative subset of the points.

Quad trees are useful in indexing 2D and 2.5D data. The 3D equivalent of quad trees are octrees, where each node has 8 children instead of 4. ASI has implemented both data structures and uses whichever approach best fits the data, i.e. quad trees for digital elevation maps (DEM) and octrees for point clouds.

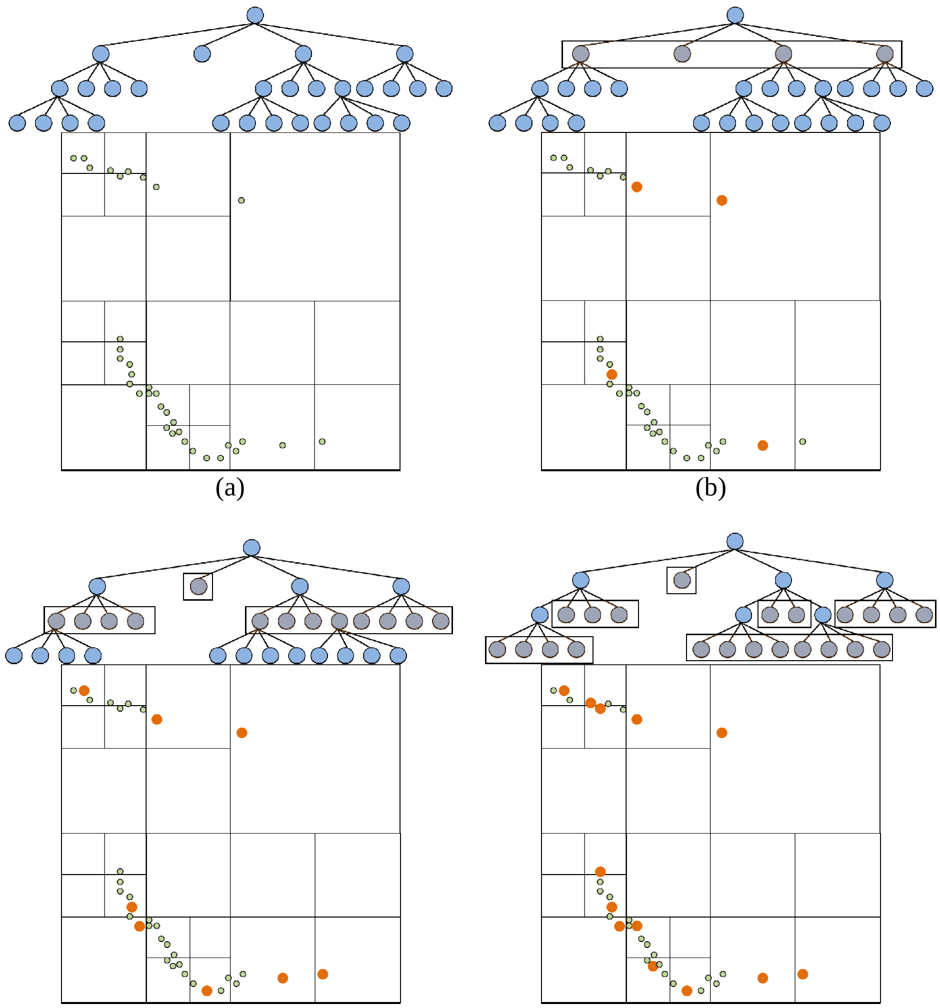
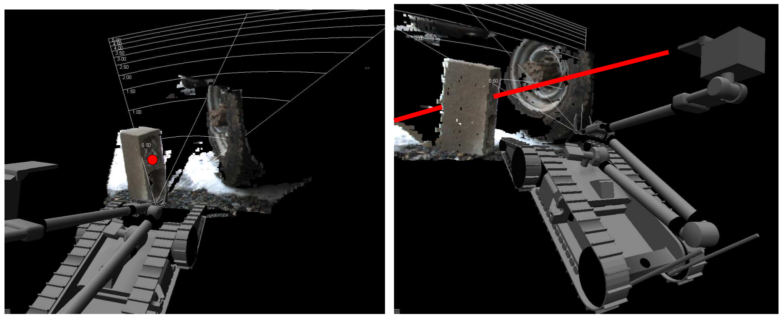


Figure 5: Quad trees for intelligent data decimation. As the Level of Detail (LOD) is increased (figures b through c), more points are expressed (shown in red).

4. AUTOMATED MANIPULATOR PLACEMENT / MEASUREMENT

Interacting with the 3D world using a 2D interface has traditionally meant that depth or distance cannot be indicated. Methods such as visual servoing can approach a target, but the operator cannot determine the distance to that target or discriminate between conflated objects, such as when a further object is partially occluded by a closer one.

ASI has developed a technique to select locations and objects in 3D space using a 2D interface. The user views a camera image or an arbitrary perspective of the sensed 3D data. This flat representation is a mathematical projection of the 3D world into the 2D image plane. Therefore, when a user clicks on a location in this flat view, it expands into a line perpendicular to this view in 3D space. The different points along the line represent the depth ambiguity of this selection (see Figure 6).



(A) (B)

Figure 6: The point selected in (A), shown by a red dot, is not a unique position in 3D space. The set of possible points corresponding to the selection is represented by the red line in (B).

With a 2D camera-based approach like visual servoing, this range ambiguity cannot be solved. Since ASI has a full 3D model of the world, however, there are two ways to fix the actual point of selection.

The intersection of a line and a surface is one or more points. Where the ambiguity line first intersects the world model surface is the best interpretation of the user's desired selection point, because the other potential intersections are occluded. ASI has found that this is the best way to select a single point on the world model surface. ASI has added a visual tool to the TruSpective application that allows a user to select and visualize the absolute distance between two selected points, which could include a location on the robot itself. Often, it is desired to select a volume instead of a single point. By selecting the same object in two or more views of the world, the user can indicate the precise 3D volume of interest using a 2D viewer.

Click and go behavior is implemented by combining this 3D selection with a manipulator resolved motion (fly the gripper) technique (Figure 7). The user drives as normal to an object of interest and captures a 3D model of the relevant area. Clicking a point in the model indicates the user's desired manipulator location, to which the gripper autonomously travels once the user presses the 'go to' button.

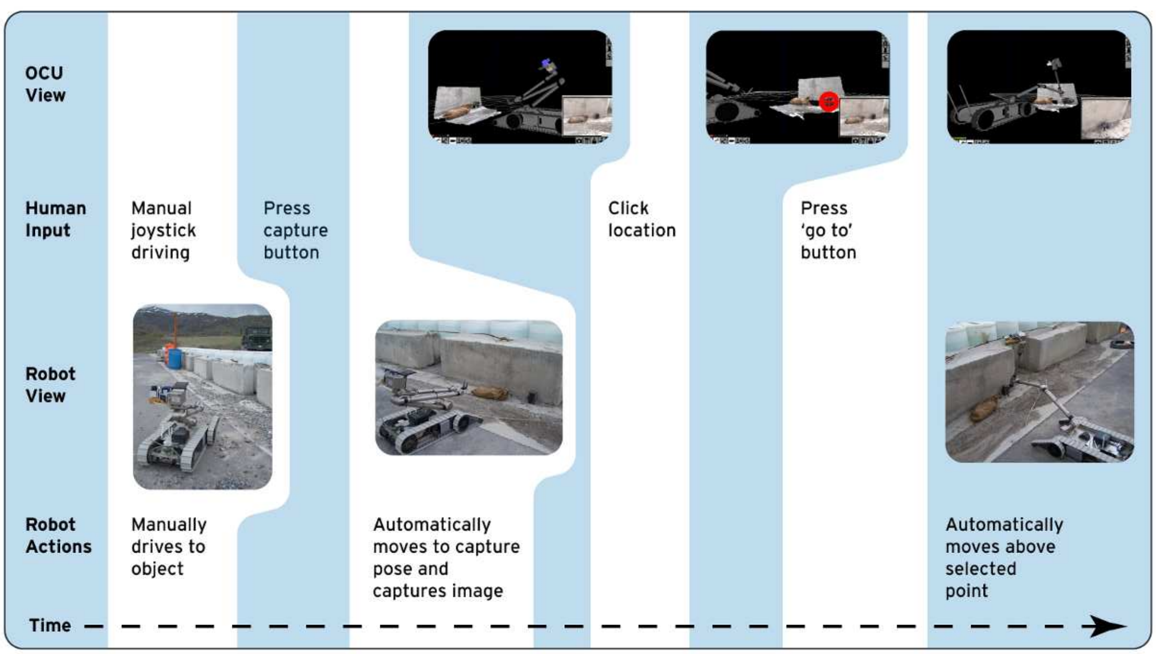


Figure 7. "Click and Go" manipulator implementation from the user's perspective.

5. WORLD BUILDING FOR NAVIGATION

The use of 3D data can aid in robot operation for navigational tasks where the robot is being driven remotely. The sensed environment can give the user situational awareness and help planning paths to drive. By using the same application originally designed for small scale manipulation and extending the world to include large terrain models, ASI has enabled a virtual click and drive user interface.

5.1 Implementation

For large world models longer range sensors with wide field of view are needed, for this reason the Velodyne High Definition LIDAR (HDL) was integrated into the system (Figure 8). The Velodyne has a large full 360 degree field of view with range of over 50m and collects a massive amount of data per second. To accompany the ranging a Pt Grey Ladybug2 is used to provide spherical texture projection that colors the perceived 3D data. The system requires position estimation from any source, which in the case of the figures in this proposal are GPS based.

Figure 8. Large scale world building implementation integrated on a Toyota Highlander. Top red sensor is Pt Grey Ladybug2 and metallic cylinder object is a Velodyne HDL.

5.2 2.5D and 3D representation

The massive amount of data being registered from LIDAR is compressed into a 2.5D terrain model that can be updated in real time. The texture model can either come from calibrated cameras (Ladybug2) or from a priori satellite images. The terrain model uses scalable tiles such that levels of detail are changed based on current frame rate and viewer perspective on the virtual world.

ASI's work to provide 3D surface modeling of laser points has proven to be difficult due to the large amounts of data and structure of data from a spinning LIDAR source. Currently models are generated from single 360 degree captures to avoid registration problems. Because of this, the data structure is spherical with less density in far regions. This resulting model's mesh is difficult to use for object identification unless the terrain is very structured and rectangular (Figure 9). Current work involves collecting multiple scans into octrees and creating surfaces from subsets of the fused data.

Figure 9. A barrel standing outside of the ASI building. Left: Real image of the scene, Right: 3D modeled image using triangle surfaces. The gap behind the blue barrel is caused by occlusion and the current limitation of surfacing from one viewpoint.

GPU processing is used to enables high speed projection and rendering of input videos, whether they are orthogonal or spherical projections. The texture added to the terrain helps add context to the scene and fills in details where 3D resolution is low (Figure 10).

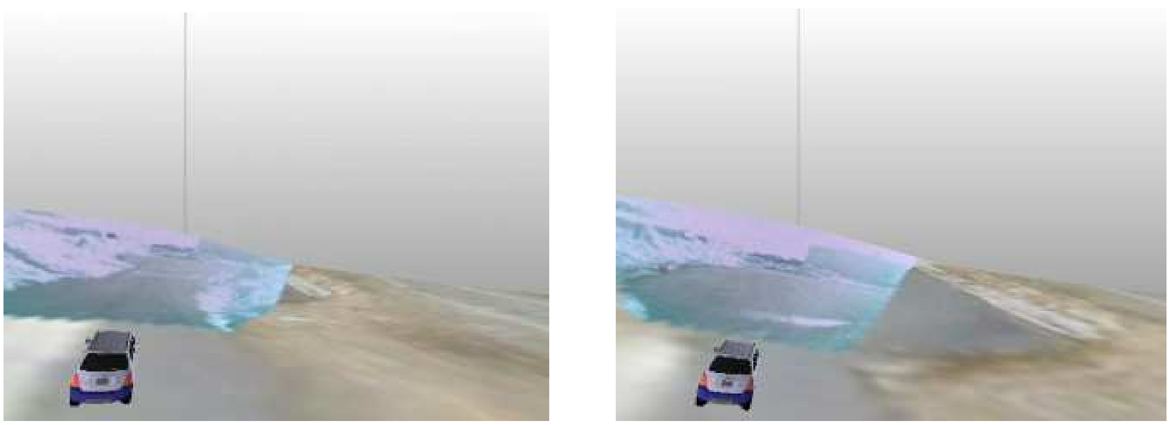


Figure 10. Live video projection onto terrain. The texture data in front of the vehicle is captured as the vehicle drives.

5.3 Click and Go Mobility

Since the vehicle position is relative, goal waypoints sent from the UI can be reached within the accuracy of the local positioning system (Figure 11). This method has advantages over visual servoing because a vehicle path can be generated and modified by obstacle avoidance algorithms as well as being more robust to image condition such as lighting or feature detection issues. The point selection method is used as described in Section 4. The path planning and obstacle detection and avoidance system relies on ASI's navigation software.

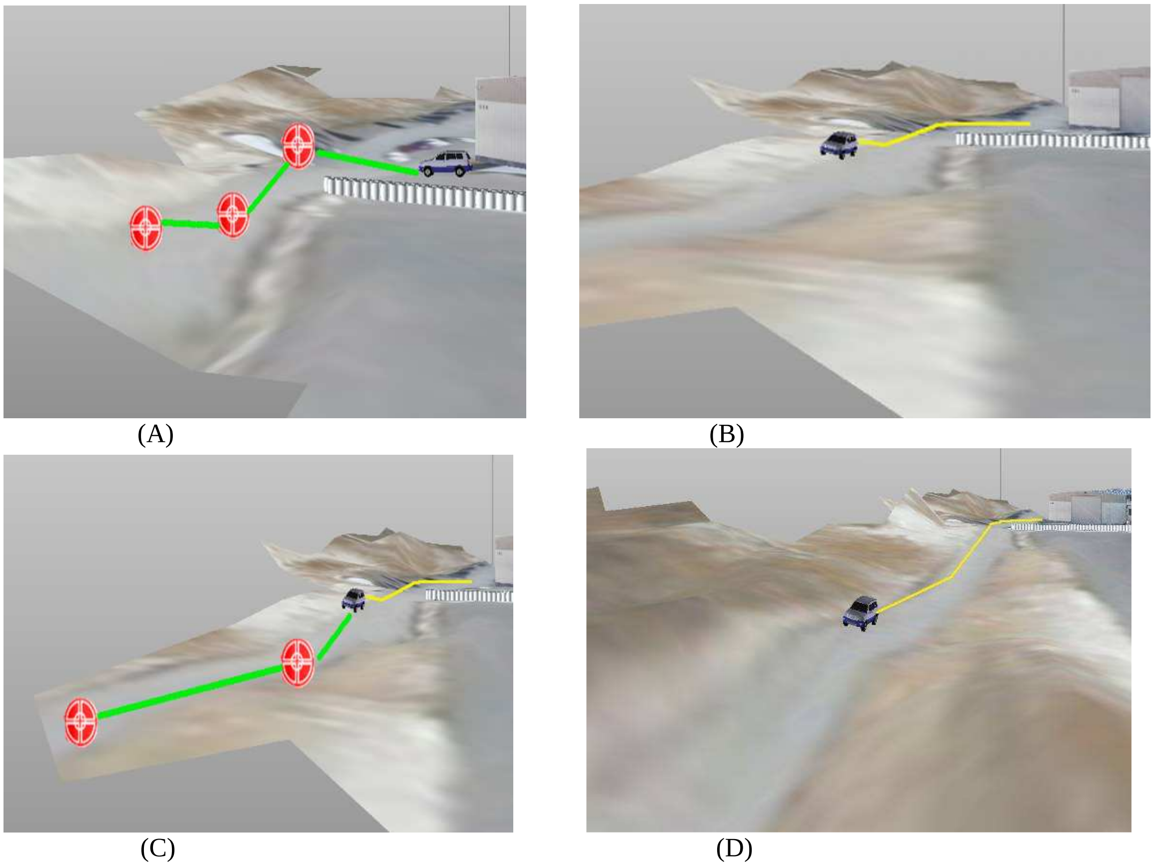


Figure 11. Point and click driving example. (A) User selects waypoints, which are used to create a planned path. (B) the vehicle navigates the chosen path while avoiding obstacles. (C) Using newly generated terrain further waypoints are selected by the user and a new path is generated. (D) Vehicle continues following path.

6. RESULTS

Preliminary testing on user performance gains for manipulation have been performed using civilians and EOD soldiers. The Human Robot Interaction (HRI) study was run in three different operational modes for comparison. These include line-of-sight operation, video feedback only and 3D visualization feedback with video. Three tests by were conducted in order to encapsulate tasks normally conducted on an EOD mission. Eight people with various experience levels performed the tests. The three tests used in this study were: uncover a bomb and pull out a blasting cap, object removal and inspection, pickup and move an object and place a charge (Figure 12).

Figure 12. HRI testing. Left User removes an IED from a rock pile. Right: user removes a blasting cap from and IED.

Each task was completed in each of the three modes of operation. The task time from start to completion was recorded as well as the number of times the arm or gripper made accidental contact with objects. The results of the study show the addition of the 3D system can help a user make fewer collisions with obstacles (Figure 13) and sometimes reduce the time to complete a mission for EOD technicians (Figure 14). Surveyed EOD technicians and ASI employees reported that they took longer on some missions because they had more information about the scene and used the virtual camera panning and zooming to better assess the task. On one of the tests the 3D captured world did not prove useful to the users because of quality/accuracy issues involved with manipulating a small object only a few millimeters in width.

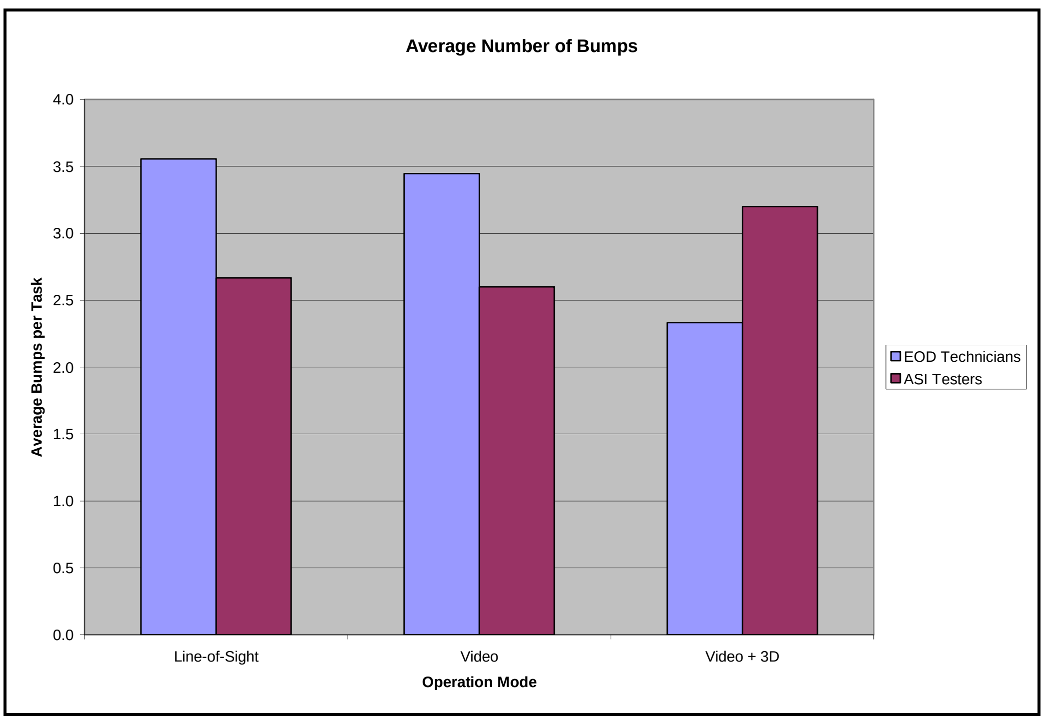


Figure 13, Average number of accidental collisions with objects

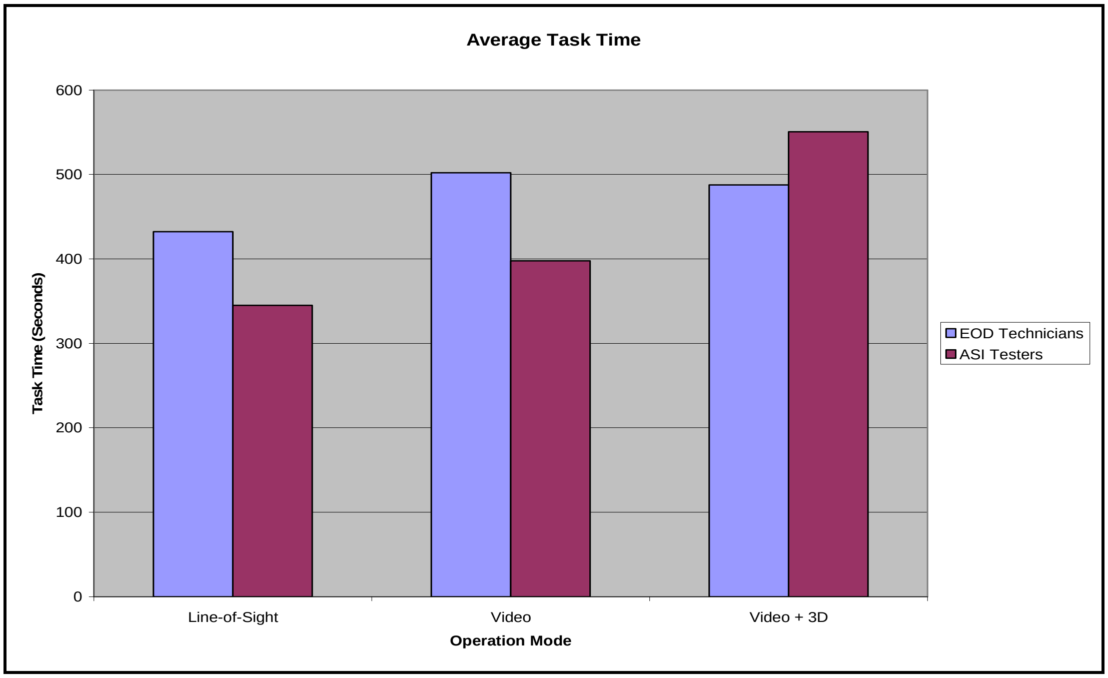


Figure 14, Average task time

7. CONCLUSIONS

This paper presented a modular 3D visualization architecture that enables a user interface for situational awareness and automation of manipulation and mobility tasks. The main conclusion from the feedback of the HRI study was that the 3D visualization was helpful for gaining situational awareness. The most requested area of improvement dealt with quality and accuracy of the 3D data. The HRI study also provided useful feedback on user interface improvements. More work needs to go into using 3D for mobility operations but initial testing has been positive in enabling a simple interface for autonomous driving behaviors. The use of real time world modeling enables systems that eliminate the need for high accuracy pose and a priori map data.

8. FUTURE WORK

The major focus of future research and development will be the integration of a fine point cloud registration algorithm such that larger worlds can be built that are more consistent. The addition of visual odometry with the stereo camera will be used to correct for local inaccuracies of the sensor frame caused by odometry and manipulator encoders. Other efforts will go toward enhancing the resolution and quality of the 3D perceived data through multiple view fusion techniques. Further HRI studies will be conducted using the TruSpective world building and click and go navigation technologies.

Acknowledgments

The authors would like to thank NAVEODTECHDIV and TARDEC for their funding support in this research.

REFERENCES

[1] Berkemeier, M., Poulson, E., Aston, E., Johnston, J., Smith, B., "Development and Enhancement of Mobile Robot Arms for EOD Applications," Proc. of SPIE, Vol. 6962, 69620P-69620P-12 (2008).

[2] Finkel, R.A., Bentley J.L., "Quad trees a data structure for retrieval on composite keys," Acta Informatica, Vol 4, No 1., 1-9 (1974).