Surface Reconstruction of Ancient Water Storage Systems
An Approach for Sparse 3D Sonar Scans and Fused Stereo Images

Keywords: Geometric Reconstruction, Underwater Stereo Vision, Level Sets.

Abstract: This work presents a new approach to reconstructing surfaces of underwater structures from stereo images and sonar scans collected with a micro-ROV on the islands of Malta and Gozo. Using a limited sensor load, sonar and small GoPro Hero2 cameras, the micro-ROV is able to explore small hard to reach water systems and gather data. Our algorithm is able to reconstruct geometric models of explored regions, even when the data is noisy and sparse. As a preprocess to the reconstruction pipeline, a 3D evidence grid is created by mosaicing horizontal and vertical sonar scans. A rough implicit surface representation can then be reconstructed using a level set method. Small stereo cameras mounted to the ROV capture fine- and medium-scale details from the scene and store them in stereo image pairs, which are transformed into point clouds and projected into the volume. A raycasting technique is used to trim the volume in accordance with the projected point clouds, thus reintroducing finer details to the previously rough model. The resulting volume is surfaced, yielding a final mesh, which can be viewed and interacted with for archaeological and educational purposes. Initial results from both steps of the reconstruction pipeline are presented and discussed.

1 INTRODUCTION

Many underwater sites such as cisterns, small sea caves, and other areas inaccessible to humans offer extraordinary opportunities for archaeological study. Remotely Operated Vehicles (ROVs) are commonly employed to explore such sites due to their small size, maneuverability, and sensor payload capacity. A common research goal between archaeologists and scientists exploring these sites is the ability to create accurate reconstructions of the geometry found within. These reconstructions can be used to visualize scale, structure, and water level, examine interesting features more closely, and potentially date the cisterns and surrounding sites.

In this work we focus on the creation of surface meshes of underwater cisterns and water galleries from a sparsely populated 3D evidence grid input. The evidence grid input is created from a unified map of several horizontal and vertical sonar scans of walls and other geometry, which are collected with a sonar sensor mounted to a micro submersible ROV (McVicker et al., 2012). Previous work has successfully reconstructed 2D and extruded 2.5D meshes of scanned surfaces using an iterative probabilistic hole filling approach and marching cubes, using 2D evidence grids from sonar scans as an input (Forrester et al., 2013). Unfortunately, this method does not generalize well to sparsely sampled 3D evidence grids, thus inhibiting the reconstruction of surfaces representative of true site geometry.

The work presented in this paper is motivated by an ongoing interdisciplinary project with the broad goal of exploring and mapping cisterns, water galleries, and shoreline caves on the islands of Malta, Gozo, and Sicily for archaeological study. Using an ROV, these water storage cisterns, which date back as far as 350 B.C.E., were explored with a micro-ROV while collecting sonar scans, depth measurements, compass measurements, video and stereo images (Fig. 1). Six different expeditions have resulted in the exploration of over 100 sites. For more information on evidence grid generation and the ROV cistern mapping project, see (McVicker et al., 2012), (Forney et al., 2011), and (Dobke et al., 2013). Due to the small entry-ways to these water systems, limited sensors were used, making data collection challenging and resulting in fairly sparse sonar data and poor stereo images (albeit densely sampled). Our reconstruction pipeline must handle both of these aspects of the data: varying density and overall sparsity. For surface reconstruction of sparse 3D sonar data, we use level sets (Fig. 2). Unlike other hole filling algorithms, level set methods are capable of producing closed surfaces independent of the sample rate in the original evidence grid. The level set method outputs a 3D implicit surface which is used to compute a volume. The reconstructed volume serves as a good rough approximation of the shape of the site's true geometry. To add finer geometric details, stereo images

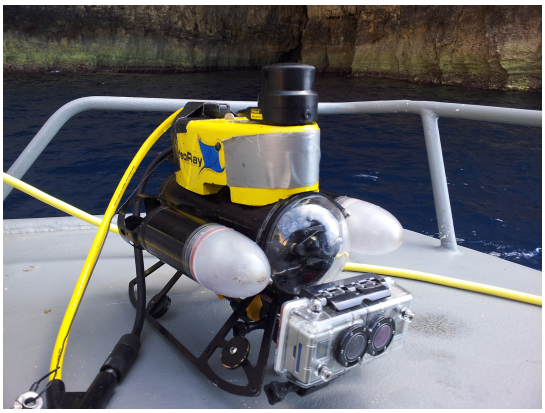
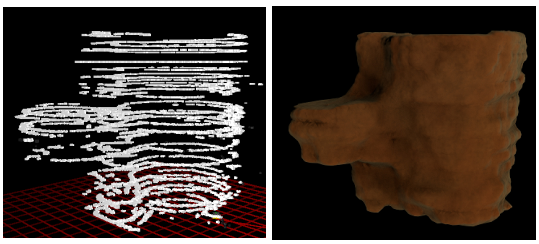


Figure 1: The VideoRay Pro III GTO is an underwater micro-ROV with dimensions 36.8 cm x 28.9 cm x 21.6 cm. The included sensors consist of a depth sensor, a compass, and a front and rear video camera. A removable Tritech Micron scanning sonar was mounted to the top of the ROV, and two vertically aligned GoPro Hero2 cameras were mounted to the front in a waterproof stereo casing.

of interesting features captured within the cisterns are turned into point clouds and projected into the volume. The point cloud data is considerably more dense in the local region it represents compared to the sparse 3D sonar data. To address this difference in density, the projected stereo point clouds are used as a basis for raycasting, where all voxels in the volume which lie beyond the projected point cloud have their occupancies set greater than zero. After trimming the volume in this manner, the newly introduced zero-crossings allow surfacing algorithms such as marching cubes (Lorensen and Cline, 1987) to reintroduce stereo features previously omitted from the model. An overview of the algorithm pipeline is shown in Fig. 3.

Presented in this paper are the details behind applying the new reconstruction algorithm to 3D sonar and stereo image data. The proposed algorithm



(a) (b)

Figure 2: A two-chambered cistern in Mdina, Malta that was explored and mapped. (a) displays the evidence grid generated from 40 horizontal sonar scans taken while hovering the ROV up the cistern at 0.2 m intervals. (b) shows the water tight mesh produced by the first step of our pipeline.

can produce water tight geometric models, representing complex underwater storage systems, even given sparse input data. We present results of three general surface reconstructions from sparse 3D sonar and one surface reconstruction with detailed geometry added via stereo imagery.

2 RELATED WORKS

Surface Reconstruction: Surface reconstruction of unorganized points in three dimensions is a well studied problem with many valid methods. One of the most popular recent approaches is Poisson Surface Reconstruction (Kazhdan et al., 2006). This method takes a point cloud with oriented point normals as its input and creates an indicator function (an inside outside table) which it can then use to determine connectivity of input points and extract a 3D model.

In contrast, level set methods (Zhao et al., 2001) take surface patches, curves and points as input and generate a distance function to input data. Based on the distance function created, an initial surface surrounding the input data is generated. The initial surface is updated based on vector and scalar fields generated from the moving surface interface and the input data. A final surface representative of the initial data set can be extracted when the initial surface reaches the input data. Given the sparsity of our input data and the lack of oriented normals, level set methods are more appropriate for creating rough starting models in our setting.

Underwater Stereo Reconstruction: The creation of accurate reconstructions from stereo images is a field of ongoing study. Stereo matching is a difficult process, complicated further by the underwater setting where non-uniform illumination, visibility falloff, and optical aberrations cripple matching algorithms that work well in air. Entire research endeavors have been devoted to characterizing attenuation and light transmittance through water as a function of sediment levels and object distance for stereo imaging purposes (Nascimento et al., 2009). In (Swirski et al., 2010), researchers were able to produce accurate disparity maps of underwater scenes using light flicker. However, there are no such light flicker effects in the underground cisterns explored in this project with which to base a stereo correspondance algorithm.

The same budget stereo camera system (side by side GoPro Hero2 cameras) used in this project was utilized in (Schmidt and Rzhanov, 2012) to generate disparity maps of underwater scenes. While the cameras were able to resolve certain features down to 3

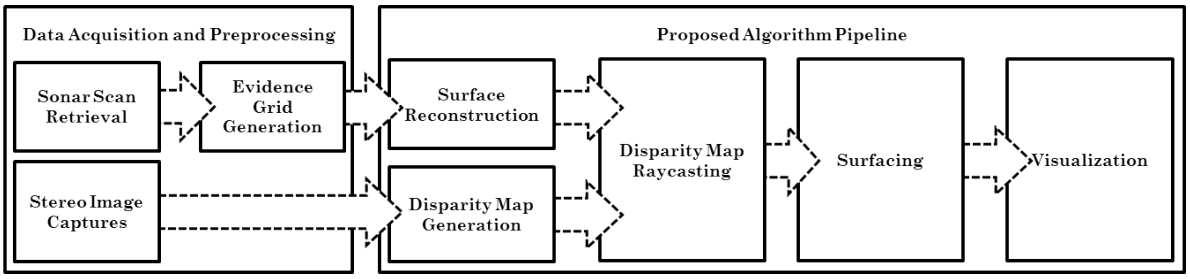


Figure 3: The proposed algorithm pipeline to create 3D reconstructions of cisterns.

mm, they were not found to be ideal due to their short 3.5 cm baseline and domed lenses.

Several research efforts have focused on reconstruction of underwater scenes from sensor information. For example, in (Beall et al., 2010) and (Drap et al., 2007), areas of the seafloor were reconstructed through image mosaicing. In (Hurtós et al., 2009), a sensor fusion approach is used to generate 3D mosaics of underwater settings using cameras, sonar, and other sensors on an AUV. Finally, in (Mahon et al., 2011), divers collected stereo images of a submerged town which were used to reconstruct a surface of the landscape. While many of these projects provide good means of reconstructing underwater surfaces, few attempts have been made to model closed 3D chambers without human aid. In addition, few of the available stereo matching algorithms account for poorly-lit underwater scenes with no sunlight penetration.

3 SURFACE RECONSTRUCTION

Due to the small entryways to the water systems we wish to map, only limited sensors can be used, (two GoPro HD Hero2 cameras and a Tritech Sea-Sprite sonar sensor). These limited sensors result in fairly sparse sonar data and poor stereo images. In order to construct the best representation of the underwater system, our reconstruction pipeline must handle both of these aspects of the data: varying density and overall sparsity.

3.1 Evidence Grid Input

For our geometric reconstruction we take a 3D evidence grid obtained from a Video Ray Pro III GTO ROV and Tritech Micron scanning sonar (McVicker et al., 2012). Unlike previous surface reconstruction attempts of cisterns which were made from 2D data (Forrester et al., 2013), this work uses new 3D sonar data sets. The input 3D evidence grid data structure is a uniform grid with each cell containing a probability that the cell represents a solid surface (i.e. walls, stones, etc.). In order to turn this into a true point cloud we only accept cells having a probability greater than a threshold, as points in the cloud. We wish to fit a surface to this input data that best represents the measured environment. Given the sparsity of the data, we use a level set method to fit a minimal surface the input points.

3.2 Level Set Method

Level set surface reconstruction works by starting with an initial surface that is a bounding volume of the input data and iteratively moving the surface towards the input data. This surface is represented as Γ, the zero level set of a function, ϕ, in 3 dimensions.

In our implementation, we use a gradient flow model to move the surface, as described in (Zhao et al., 2001). Our adopted movement equation is:

$$\Delta\phi = \Delta t|\vec{n}|d^{(P-1)}\vec{g}\cdot\vec{n} + \frac{1}{P}d\kappa \qquad (1)$$

where Δt is a fixed time step, $\vec{n}$ is the gradient of ϕ, d is the distance to the nearest original data point, $\vec{g}$ is the gradient of the distance function, and κ is the curvature of the current surface. $\vec{n}$, ϕ, d, $\vec{g}$, and κ are all functions of 3 dimensions represented by voxels in a volume.

Eq. 1 is used to iteratively move the function ϕ so that its zero level set ϕ moves towards the final reconstructed surface.

During each iteration we update the values of ϕ, $\vec{n}$, and κ for the voxels in the volume. To increase performance, we only calculate these values for points in the narrow band. The narrow band is a set of voxels which are near to the iterating surface, as described in (Adalsteinsson and Sethian, 1994).

We calculate ϕ to be the distance from each voxel to Γ. To calculate κ, we use the following equations, adopted from (Osher and Fedkiw, 2003):

$$\kappa = \phi_x^2\phi_{yy} - 2\phi_x\phi_y\phi_{xy} + \phi_y^2\phi_{xx} + \phi_x^2\phi_{zz} - 2\phi_x\phi_z\phi_{xz} + \phi_z^2\phi_{xx} + \phi_y^2\phi_{zz} - 2\phi_y\phi_z\phi_{yz} + \phi_z^2\phi_{yy} \quad (2)$$

Here, ϕ_x is the first partial derivative of ϕ in the x direction. We use this second-order accurate central difference formula

$$\frac{\delta\phi}{\delta x} = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} \quad (3)$$

where ϕ_y and ϕ_z are similarly calculated. ϕ_{xx} is the second partial derivative of ϕ in the x direction. We use this second-order accurate finite difference formula

$$\frac{\delta^2\phi}{\delta x^2} = \frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{\Delta x^2} \quad (4)$$

where ϕ_{yy} and ϕ_{zz} are similarly calculated. ϕ_{xy} is the second partial derivative of ϕ in the x and y directions. We use this second-order accurate finite difference formula

$$\frac{\delta\phi}{\delta x}\frac{\delta\phi}{\delta y} = \left(\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x}\right)\left(\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x}\right) \quad (5)$$

where ϕ_{xz} and ϕ_{yz} are similarly calculated.

d must be calculated for all voxels in the volume, which can be a prohibitively large number of calculations. To efficiently calculate the distance function we use the fast marching method described in (Sethian, 2001).

The fast marching method works by considering three sets of voxels:

1. Voxels with accepted distance values. Initially this set containts all voxels on the surface, since distance for these voxels is known to be zero.
2. Voxels adjacent to accepted voxels. Since these voxels have neighbor voxels for which distance is known, their distance can be easily calculated.
3. Voxels that are far away. These voxels will be considered once they become adjacent voxels.

The fast marching algorithm works by:

1. Selecting the adjacent voxel with lowest calculated distance.
2. Finding all neighbors of the selected voxel that are in the far set and adding them to the adjacent set.
3. Calculating distance for all neighbors of the selected voxel, using this new distance only if it is less than any previously calcuted distance for that voxel.
4. Moving the selected voxel to the accepted set.
5. Continuing iteration until the adjacent set is empty.

The fast marching method is also applied to updating ϕ during the surface iteration process.

Under ideal circumstances the surface reconstruction process can be exited once the surface no longer changes significantly between iterations - at this point a minimum surface of the input data has been reached. However, in sparse data sets such as those we collected from cisterns, the surface may be pulled through gaps in the data where a surface actually existed. We therefore allow for human input to end the surface reconstruction process early when an acceptable surface has been reached but before that surface has been pulled too far through holes in the data.

4 STEREO RECONSTRUCTION

While the 3D volumetric reconstructions from Sec. 3 give a broad idea of the true shape of the cistern geometry, the Tritech Micron scanning sonar fails to capture small features such as crevices, rocky walls, and archways due to a large 35° vertical beam angle. In addition, both hardware and software resolution constraints are introduced in the scan retrieval and evidence grid generation preprocessing stages. These limitations cause the volumetric 3D reconstructions to omit many important small features. In the disparity merging step of our algorithm we account for the limited resolution in our volumetric reconstructions by reintroducing finer details captured in stereo images to the model. Note that stereo image pairs will be of a significantly higher resolution than the sparse 3D sonar grid in a local region. This difference in resolution of data is handled by our algorithm using a projective raycasting technique.

4.1 Disparity Map Generation

Fine- and medium-scale features from deployments into cisterns and caves are captured and stored in stereo image pairs using two vertically aligned GoPro Hero2 cameras. The captured stereo images have minor barrel distortions due to the domed camera lenses and non-uniform illumination due to the ROV's poor ability to fully light the scene, so all stereo images are retouched by applying a constant lens and lighting correction. Stereo image pairs are then matched to create disparity maps using MATLAB's Computer Vision System toolbox. The resulting collection of disparity maps of interesting features are converted to

points clouds using pixel intensity for each point's Z-coordinate, and projected into the volumetric reconstruction made in Sec. 3 by applying a 3D affine transform.

Stereo vision is inherently challenged by the underwater setting, and even after correction many stereo images were plagued with non-uniform illumination, visibility falloff, and optical aberrations. These complications led to difficulties in feature recognition and matching, and ultimately limited the quality and number of disparity maps we were able to produce.

4.2 Disparity Map Raycasting

Disparity maps are converted to 3D point clouds, a more approachable data structure for raycasting, following

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_x \\ \mathbf{p}_y \\ \mathbf{p}_z \end{bmatrix} = \begin{bmatrix} I_x \\ I_y \\ \delta(I_{xy}) \end{bmatrix} \tag{6}$$

where $\mathbf{p} \in \mathbf{P}$ is a point in the point cloud $\mathbf{P}$, I is a disparity map image with pixel space coordinates (I_x, I_y), and $\delta(I_{xy})$ is the depth value stored in the intensity of pixel I_{xy} (Fig. 4). The δ function is a calibration function which maps a disparity value between left and right stereo images to a real distance based on the properties of the camera. δ is formulated such that one unit in world space correlates to one meter in the real world. Point clouds are then individually assigned to projectors, which are implemented as user-controlled objects that may be manually rotated and translated within our program based on mouse and keyboard input to allow the user to align the features captured in the stereo images with features in the model. Rather than using the pixel space point cloud, $\mathbf{P}$, for raycasting, we project points outwards from each projector, $\hat{\mathbf{j}}$, in such a way that points are constrained within the projector frustum. This projection produces a new point cloud, $\mathbf{P}'$.

$$\mathbf{P}' = \sum_{\theta \in \Theta} \sum_{\phi \in \Phi} \mathbf{p}'(\theta, \phi) \tag{7}$$

$$\mathbf{p}'(\theta, \phi) = \hat{\mathbf{j}} + \delta(I_{xy})(\hat{\mathbf{w}} + \sin(\phi)\hat{\mathbf{v}} + \alpha \sin(\theta)\hat{\mathbf{u}}) \tag{8}$$

where θ and ϕ are a horizontal and vertical angle along the projected image plane such that $\Theta = \{-\theta_{max}, \ldots, \theta_{max}\}$, $\Phi = \{-\phi_{max}, \ldots, \phi_{max}\}$, and θ_{max} and ϕ_{max} equal half of the horizontal and vertical field of view of the GoPro Hero2 GTO cameras, α is the disparity map's aspect ratio, and $\hat{\mathbf{w}}$, $\hat{\mathbf{u}}$, and $\hat{\mathbf{v}}$ are the basis vectors of the projector. Projectors are manually aligned in the volume to coincide with the observed locations of the real geometry captured in each

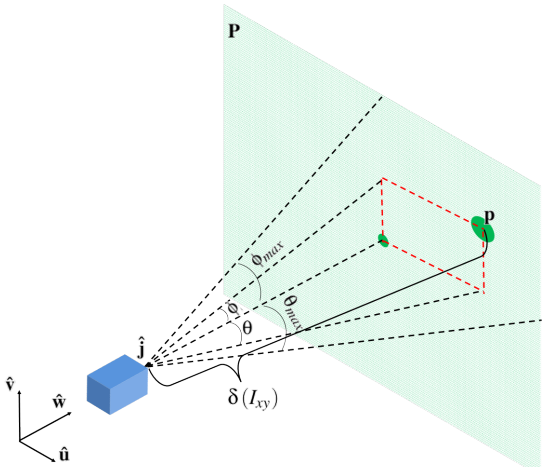


Figure 4: A diagram of the projector and initial point cloud of stereo data, $\mathbf{P}$.

point cloud. Since projections are hand-aligned, it is difficult to orient projections correctly within the volume. To aid the user in projector alignment, marching cubes is run on the volume, producing a mesh of the original surface. The mesh is visualized on top of the volume so that the user may align projections with respect to the mesh itself.

Once projectors are situated, rays are cast through the viewports of the projectors. Rays originate at the projector and are cast through each point in $\mathbf{P}'$. Bresenham's line algorithm (Bresenham, 1965) is continuously executed along each cast ray to find the next voxel in the ray's path. When the ray arrives at a voxel containing a point, a boolean switch is triggered, setting all following voxel occupancies greater than zero (denoting that the voxel is outside of the surface). By setting new occupancies in the volume, each ray cast alters the position of the eventual surface by redefining several zero-crossings along voxel edges (Fig. 5). Once a ray passes through a voxel whose occupancy is already greater than zero, the ray is terminated to make certain that no surfaces are trimmed unintentionally.

Although ray casting cannot guarantee that all voxels beyond the projected point cloud will be modified, the alterations to the occupancies in the volume generally occur near walls, so cast rays do not diverge far enough to miss any voxels. Additionally, projected point clouds may be sampled with sub-pixel accuracy allowing the ability to cast rays at a finer resolution, effectively minimizing the possibility that a voxel will be missed.

The original resolution of a volume is decided based on the properties of the sonar sensor and the capabilities of the occupancy grid generation algorithm

utilized for the input to this reconstruction pipeline. Since the original resolution of the volume is only good enough to retain the details of the general surface, the volume is subdivided prior to raycasting to increase the amount of detail achieved in the areas which will be be modified by stereo data. In most cases the volume can be subdivided one to three times, yielding 8x to 512x as fine of a resolution. In order to facilitate smooth surface generation using marching cubes, voxel occupancies are interpolated trilinearly between subdivisions.

The algorithm is currently limited by memory consumption. Even with efficient data storage structures, the entire volume must be subdivided, so medium volumes subdivided in excess of two times will crash. Due to the volume subdivision limitations, the algorithm cannot truly add the same level of detail to the mesh as what is stored in the stereo images. Future work includes multi-resolution approaches to address this issue.

5 SURFACING AND VISUALIZATION

Marching cubes is run on the trimmed volume to produce a closed surface mesh, which can be visualized and interacted with by researchers. In addition to being able to manipulate the mesh in our visualization software, the interpolated marching cubes mesh is rendered in Cinema4D with a bump map and fresnel shader to produce visually appealing static images and flyby videos.

Our visualization software also grants the ability to view errors in the level set reconstruction step by using a signed distance function to color map error onto vertices in the mesh. This visualization mode allows archaeologists to understand which areas of the reconstruction are likely to be most accurate.

6 RESULTS

The level set reconstruction method was applied to three data sets, including a mushroom shaped chamber (labeled "*The Mush-room*") from a large water gallery in Valletta, Malta, a complete water system connected by two ROV deployment entrances (labeled "*Site 3+4*") in Tal Gruwa, Gozo, and a complete two-chambered cistern (labeled "*The Archives*") in Mdina, Malta. The stereo reconstruction method was applied to The Mush-room.

Both horizontal and vertical sonar scans were collected in The Mush-room. During ROV deployment, the ROV was flown to a resting position (noting its heading and depth), and two 360° sonar scans orthogonal to one another (one horizontal and the other vertical) were collected. For more information on the double sonar scanning configuration see (Dobke et al., 2013). The collection of paired horizontal and vertical scans from each rest position were unified into an evidence grid. Data for The Archives and Site 3+4 was collected by horizontally scanning the cistern walls at 0.2 and 0.5 meter vertical intervals, respectively (with no vertical scans, due to hardware constraints). In some cases, scans would be duplicated and translated to a different depth in the evidence grid due to vertical symmetry in the walls of the cistern. This is demonstrated in the three planes of horizontal scans making up the lower channel in The Mush-room's evidence grid, and in some sonar scans present in Site 3+4. Note however, that these are all true 3D data sets, not just an extruded single horizontal layer as in (Forrester et al., 2013). The evidence grids for these three sites were processed into meshes and visualized (Fig. 6).

The level set technique described in Sec. 3 worked well in most situations, but caused ceilings to cave in in regions where gaps existed in the data. For example, the roof caved in on the Site 3+4 mesh due to a lack of data in a circular central chamber. While these dimples were a source of error for data sets consisting of horizontal sonar scans, they were not present in models where vertical sonar scans were included in the evidence, such as The Mush-room. The vertical scanning capability was added mid-way through our data acquisition process and will be used on all future missions.

Stereo images captured in The Mush-room were processed into disparity maps and used to trim the volume before resurfacing. Due to the poor quality of the GoPro Hero2 lenses and ROV lights, as well as the cloudiness of the water and plainness of features, it was extremely difficult to produce high-quality disparity maps. Several methods, including prepackaged frameworks such as OpenCV and MATLAB's Computer Vision Toolbox, as well as three custom algorithms from stereo literature (Zitnick and Kanade, 2000) (Scharstein and Szeliski, 1998) (Nalpantidis and Gasteratos, 2010) were utilized in attempts to make good disparity maps. However, the distortions from the hardware as well as the plainness of the walls limited successful identification and matching of salient features. The disparity maps produced from MATLAB's Computer Vision Toolbox were the cleanest and had the most matched features,

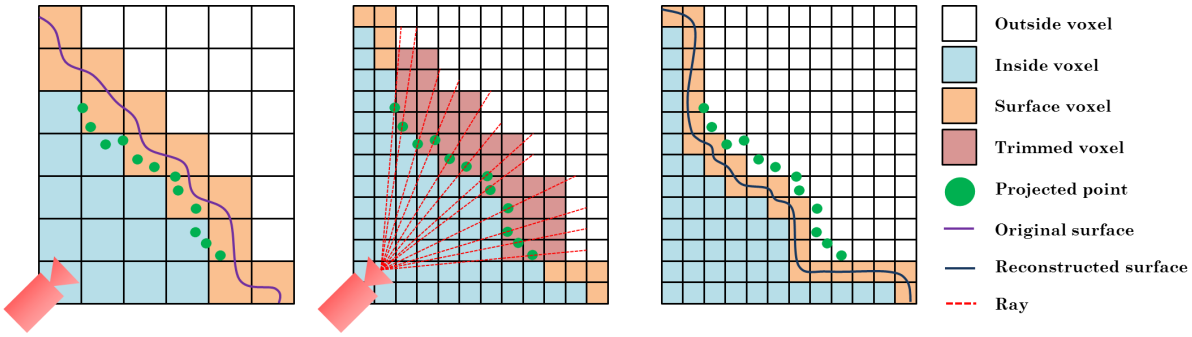


Figure 5: To trim the volume in accordance with stereo image data, point clouds are manually aligned near existing zero-crossings (the orange to white boundary), which are viewed as the surface of a mesh from the user's perspective. When the user is ready for raycasting, the volume is subdivided to provide finer resolution. Rays are then cast from the projector through points in the point cloud, marking voxels that lie beyond the intersected points and setting their occupancies greater than zero. Rays are terminated upon contact with a voxel whose occupancy is already greater than zero. Finally, the newly introduced zero-crossings allow surfacing algorithms to more accurately reconstruct the surface.

so they were used. To demonstrate the results of our algorithm, disparity maps were hand-modified, cleaned, and mirrored in some situations. The initial stereo reconstruction results were not given to archaeologists for study due to their variation from true geometry data, but are presented here as a proof of concept.

Two distinct features in The Mush-room, an archway and a large bump, were selected to add to the volume. The archway disparity map was mirrored to capture the entire feature, and both features were trimmed from the volume. Even in the presence of poor disparity maps, medium sized features were reintroduced successfully (Fig. 7). In addition, for water systems with fine rock structure, stereo disparity maps can also be used to add fine details (Fig. 8).

7 CONCLUSIONS

This work has presented a process pipeline that addresses the problem of reconstructing geometric models from 3D sonar scans of underwater settings with a micro-ROV. Surface reconstructions of underwater settings inaccessible to humans were produced for archaeological study using a small number of sonar scans taken with a low-payload micro-ROV. In addition, the work has demonstrated a method of affixing depth data captured in low cost stereo cameras to enhance rough sonar generated reconstructions. Previous work has successfully created 2D and extruded 2.5D models of closed underwater systems, while the work here presents an initial success for a unified solution for surface reconstruction from 3D sonar scans with stereo image enhancements. We are able to successfully build representational surface reconstructions given sparse 3D sonar data and able to integrate locally higher resolution stereo data to add geometric details when possible.

Three large sonar data sets were reconstructed to test the proposed pipeline's ability to handle sparse 3D sonar data. Two of the three data sets contained only horizontal scans taken incrementally along the depth of the site, and one data set was collected using both a vertical and horizontal sonar on the ROV. All three of the reconstructed models are visually appealing, representative of true data, scaled according to ground truth data, and useful for archaeological study. The process of producing accurate disparity maps from left and right images proved difficult underwater, hampering the virtues of the second phase of the pipeline, however, initial results have validated the potential to incorporate higher resolution stereo image data with the coarse resolution sonar data.

Several stages of the pipeline could be improved in future work. An octree could be substituted for the volume data structure in order to prevent memory limitations when subdividing voxels, replacing marching cubes with a dual contouring method (Ju et al., 2002) to prevent cracks. Projectors could be automatically aligned in the volume using visual SLAM, or could have their positions determined before runtime using a localization device on the ROV such as an IMU or a SmartTether. With regards to stereo hardware, the proposed stereo algorithm would greatly benefit from cameras with a wider baseline and lenses corrected for underwater photography. Additionally, a more uniform lighting system or structured lighting system would be useful in illuminating the scene properly or to provide salient features.

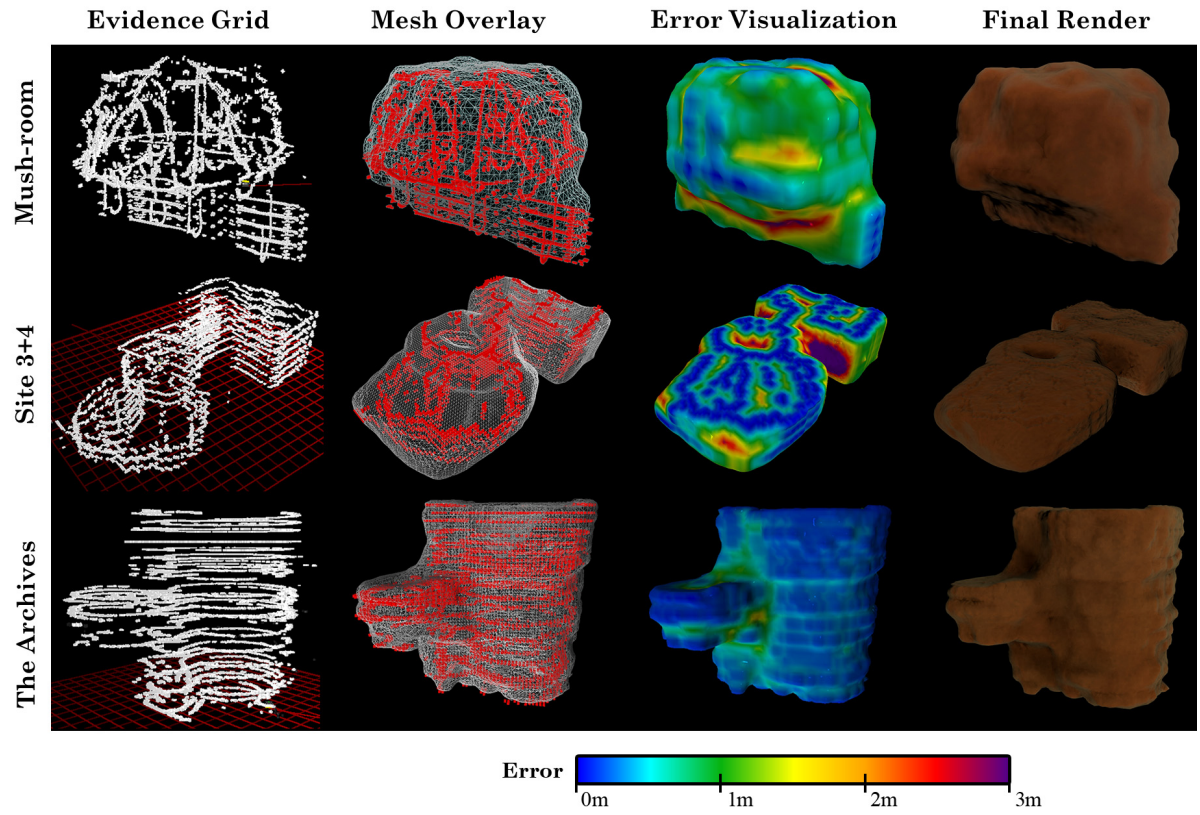


Figure 6: Results from applying the proposed 3D reconstruction technique to three large data sets - The Mush-room (top), Site 3+4 (center), and The Archives (bottom). The Mush-room's sonar data was captured in 3D by mounting a vertical sonar head onto the ROV in addition to the horizontal sonar and collecting two scans simultaneously at each capture position. Sonar data for The Archives and Site 3+4 were collected by scanning the cistern walls at 0.2 and 0.5 meter vertical intervals. The first column displays the sparsely populated evidence grid input. The second column shows the reconstructed mesh overlaid on top of the input. The third column shows the reconstructed mesh colorized according to error in distance from the input points. The final column displays Cinema4D renders of the meshes, which are given to archaeologists studying the sites.

REFERENCES

Adalsteinsson, D. and Sethian, J. A. (1994). A fast level set method for propagating interfaces. *Journal of Computational Physics*, pages 269–277.

Beall, C., Lawrence, B. J., Ila, V., and Dellaert, F. (2010). 3d reconstruction of underwater structures. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4418–4423. IEEE.

Bresenham, J. E. (1965). Algorithm for computer control of a digital plotter. *IBM Systems journal*, 4(1):25–30.

Dobke, A., Vasquez, J., Lieu, L., Chasnov, B., Clark, C., Dunn, I., Wood, Z., and Timothy, G. (2013). Towards three-dimensional underwater mapping without odometry. In *To appear in: Proceedings of the 18th International Symposium on Unmanned Untethered Submersible Technology (UUST)*.

Drap, P., Seinturier, J., Scaradozzi, D., Gambogi, P., Long, L., and Gauch, F. (2007). Photogrammetry for virtual exploration of underwater archeological sites. In *Proceedings of the 21st International Symposium, CIPA 2007: AntiCIPAting the Future of the Cultural Past: Athens (Greece), 01–06 October 2007*. Citeseer.

Forney, C., Forrester, J., Bagley, B., McVicker, W., White, J., Smith, T., Batryn, J., Gonzalez, A., Lehr, J., Gambin, T., et al. (2011). Surface reconstruction of maltese cisterns using rov sonar data for archeological study. In *Advances in Visual Computing*, pages 461–471. Springer.

Forrester, J., McVicker, W., Gambin, T., Clark, C., and Wood, Z. J. (2013). Uncertainty visualization and hole filling for geometric models of ancient water system. In *Proceedings of the 4th International Conference on Information Visualization Theory and Application (IVAPP)*.

Hurtós, M., i Soler, X. C., and Salvi, J. (2009). Integration of optical and acoustic sensors for d underwater scene reconstruction. *Instrumentation viewpoint*, (8):43.

Ju, T., Losasso, F., Schaefer, S., and Warren, J. (2002).

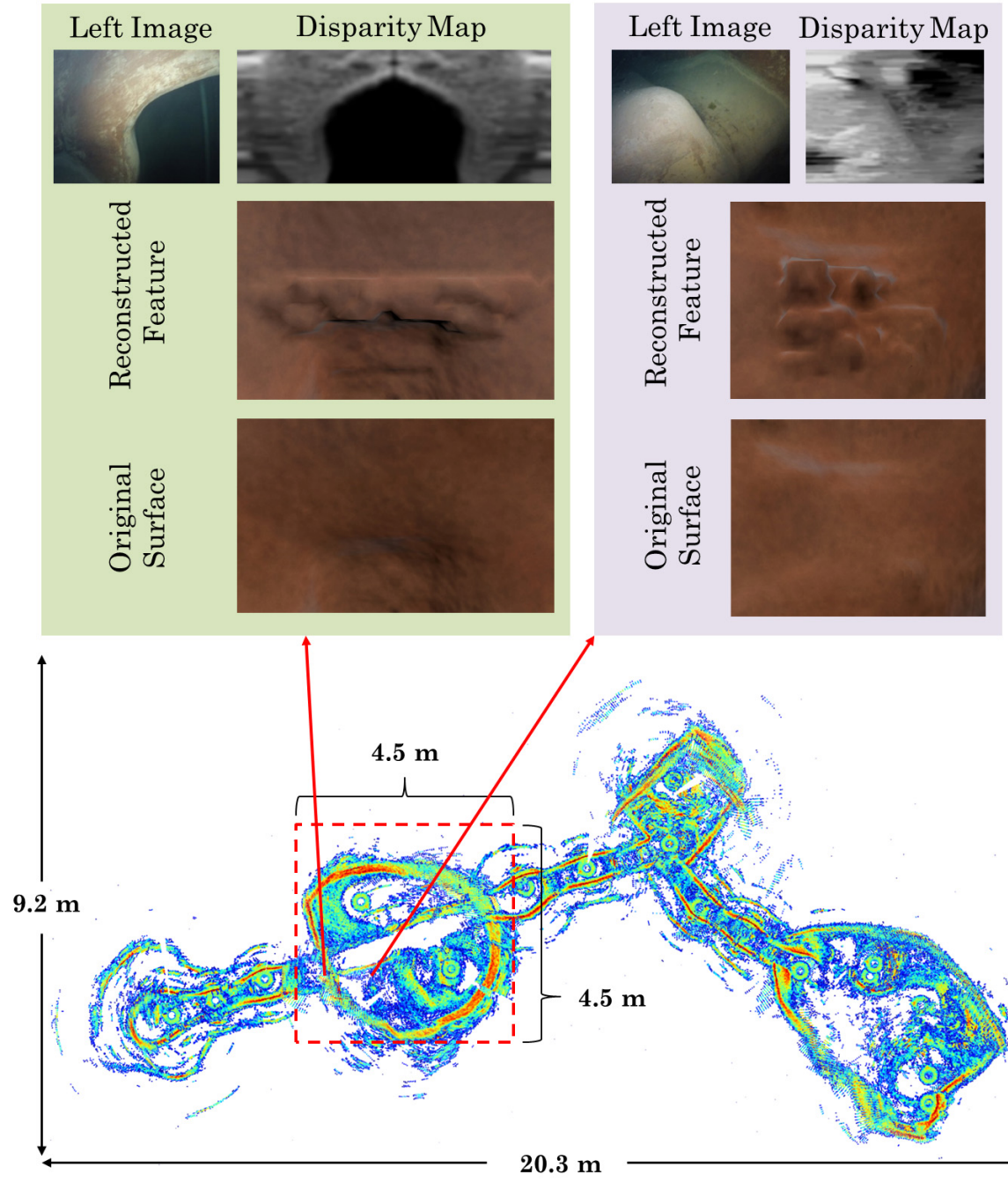


Figure 7: A sonar mosaic of the large water system containing The Mush-room (boxed in red). Interesting features omitted from the sonar model were captured with stereo cameras within the cistern and converted to disparity maps. Due to the poor image quality feature matching was extremely difficult, and disparity maps were blurred and mirrored before being projected into the volume. Reconstructed features, including an archway and a bump, were smooth and meaningful additions to the overall geometry. The 4.5x4.5x5.2 meter chamber produced a volume with dimensions 33x34x43 voxels (using an evidence grid cell size of 0.25 meters per voxel), which was subdivided once to produce a 86x68x66 voxel volume before raycasting.

Dual contouring of hermite data. *ACM Trans. Graph.*, 21(3):339–346.

Kazhdan, M., Bolitho, M., and Hoppe, H. (2006). Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*.

Lorensen, W. E. and Cline, H. E. (1987). Marching cubes:

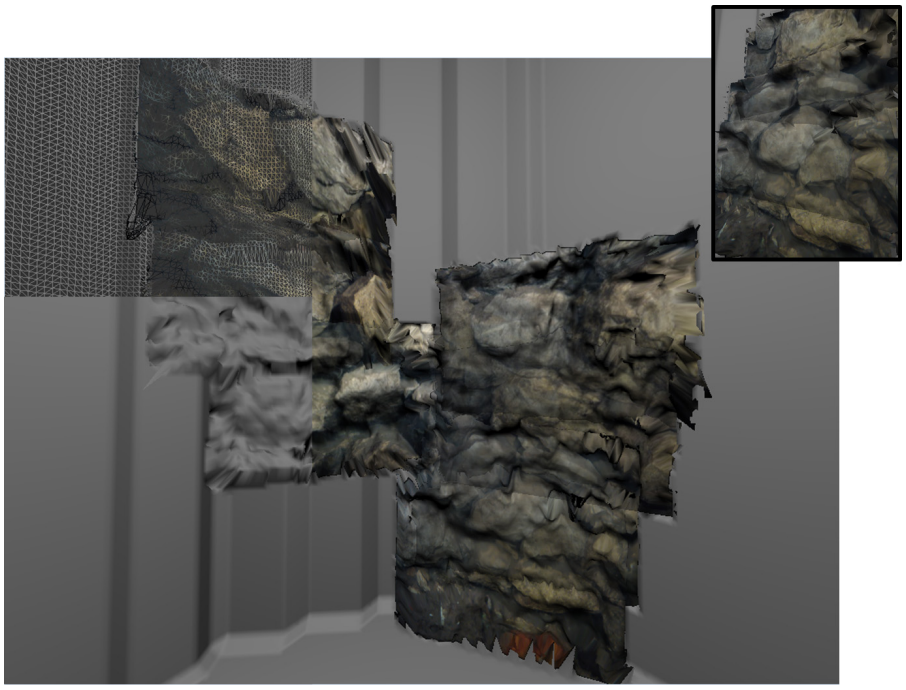


Figure 8: For water systems with fine rock structure, such as the one shown above, stereo disparity maps can be used to add fine geometric details. The result image shown here is from a well in Sicily with rock walls and was generated using our previous solution to integrating stereo data and meshes via warping the mesh directly. Similar results could be obtained with our current level sets and ray casting method.

A high resolution 3d surface construction algorithm. In *ACM Siggraph Computer Graphics*, volume 21, pages 163–169. ACM.

Mahon, I., Pizarro, O., Johnson-Roberson, M., Friedman, A., Williams, S. B., and Henderson, J. C. (2011). Reconstructing pavlopetri: Mapping the world's oldest submerged town using stereo-vision. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2315–2321. IEEE.

McVicker, W., Forrester, J., Gambin, T., Lehr, J., Wood, Z. J., and Clark, C. M. (2012). Mapping and visualizing ancient water storage systems with an rov an approach based on fusing stationary scans within a particle filter. In *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*, pages 538–544. IEEE.

Nalpantidis, L. and Gasteratos, A. (2010). Stereo vision for robotic applications in the presence of non-ideal lighting conditions. *Image and Vision Computing*, 28(6):940–951.

Nascimento, E., Campos, M., and Barros, W. (2009). Stereo based structure recovery of underwater scenes from automatically restored images. In *Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on*, pages 330–337. IEEE.

Osher, S. and Fedkiw, R. (2003). *Level set methods and dynamic implicit surfaces*, volume 153. Springer Verlag.

Scharstein, D. and Szeliski, R. (1998). Stereo matching with nonlinear diffusion. *International Journal of Computer Vision*, 28:155–174. 10.1023/A:1008015117424.

Schmidt, V. E. and Rzhanov, Y. (2012). Measurement of micro-bathymetry with a gopro underwater stereo camera pair. In *Oceans, 2012*, pages 1–6. IEEE.

Sethian, J. (2001). Evolution, implementation, and application of level set and fast marching methods for advancing fronts. *Journal of Computational Physics*, 169(2):503–555.

Swirski, Y., Schechner, Y. Y., Herzberg, B., and Negahdaripour, S. (2010). Underwater stereo using natural flickering illumination. In *OCEANS 2010*, pages 1–7. IEEE.

Zhao, H.-K., Osher, S., and Fedkiw, R. (2001). Fast surface reconstruction using the level set method. In *Variational and Level Set Methods in Computer Vision, 2001. Proceedings. IEEE Workshop on*, pages 194–201. IEEE.

Zitnick, C. L. and Kanade, T. (2000). A cooperative algorithm for stereo matching and occlusion detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(7):675–684.