# Incorporating Different Reflection Models into Surface Reconstruction

**Andrew E. Johnson**
**aej@ri.cmu.edu**

**The Robotics Institute**
**Carnegie Mellon University**
**Pittsburgh, PA 15213**

## Abstract

*Generation of elevation maps of the seafloor is an important component of any autonomous system designed to navigate underwater. Because of its long range, sonar is the sensing medium of choice for underwater navigation. However, sonar systems designed to directly generate 3D images are generally complex or have low resolution. For these reasons, techniques that generate 3D elevation maps from 2D sonar intensity images are necessary for terrain modeling and navigation. Furthermore, recovering intrinsic terrain properties, such as albedo and surface roughness, is important for undersea mapping and exploration.*

*This paper presents two techniques for the generation of elevation maps of the seafloor from side scan sonar intensity images and sparse bathymetric data. These techniques employ a reflection model of the seafloor to establish a correspondence between the intensity at a point and the surface normal at that point. The first technique uses a constraint between the surface normal and the position of the sensor to generate a partial differential equation that describes the surface. This PDE is repeatedly solved using finite differences until the intensity from the surface is close to the actual intensity data. The second technique uses an iterative relaxation method to generate the surface by minimizing the difference between the intensity data and the calculated surface intensity. This technique is similar to shape from shading methods used in computer vision. In both techniques the sparse bathymetric data is used to generate an initial guess for the shape of the seafloor and an initial guess for the reflection model parameters.*

*These techniques are designed to support different reflection models, so they can be applied to different underwater environments. This is in contrast with other approaches that are generally less flexible with respect to the reflection model used. In addition to the elevation map of the seafloor, the parameters of the reflection model at every point in the image are generated. These parameters describe material properties of the seafloor, so the maps of the reflection model parameters can be used to segment the seafloor by material.*

*Both techniques were tested on real and noisy synthetic data by comparing the calculated elevation map to the real elevation map. Using a lambertian reflection model the average error in the elevation maps was found to be less than 0.5m at 150m range while the maximum error was less than 4.5m for an image with bathymetric data spaced evenly every 16m. Furthermore, for synthetic images, the parameter (albedo) of the reflection model was calculated to within 5% of its actual value. On real data the actual albedo is not known, but qualitatively correct segmentation of albedo was obtained. In general the two methods performed equally well. Other reflection models were used with similar results.*

# 1 Introduction

The undersea world is an unfriendly environment for navigation sensors mainly because the sensors must see through the water medium. Water severely attenuates and distorts light, so optical sensors produce poor results at far range. Sonar, on the other hand, is more suitable for sensing underwater.Water does not attenuate sound energy nearly as much as light energy, so sonar has a greater range. Sonar is also less susceptible to refraction. These benefits come at the price of lower resolution and increased noisiness of the sonar data. However, given the task of autonomous navigation underwater, sonar is the best alternative.

Generally, sonar data is in the form of an intensity image that does not directly convey the shape of the seafloor. Given knowledge of the physics of sonar reflection and transmission underwater it is possible to convert this intensity image into an elevation map of the ocean bottom. The reflection model with respect to sound of the seafloor along with its correct parameters are inverted to obtain the incidence angle of the sonar pulse. These incidence angles are then converted to elevation derivatives which are integrated together to form the surface of the seafloor.

Most existing methods of seafloor reconstruction have used a specific reflection model and somewhat arbitrarily selected reflection model parameters for the seafloor to determine the incidence angles [3]. If the model and parameters are not guessed correctly then the integration will produce a surface that is not correct. Furthermore, if the integration does not have fixed starting and ending points accumulation of errors will force the surface away from its real value.

In many cases it is possible to obtain sparse elevation information about the seafloor from other sensors (i.e a global map with matched features, acoustic beacons, additional phased sonars, etc.) This elevation information can be combined with the sonar intensity image to alleviate many of the problems plaguing previous seafloor reconstruction methods. The known elevation values provide an initial guess for the shape of the surface which can then be used with the reflection model of the seafloor to estimate the reflection model parameters at every point in the image. The estimates of the elevation map and the reflection parameters are then refined through iteration by minimizing the difference between the intensity data and that predicted by the reflection model with its parameters. Because the reflection model parameters are estimated based on shape and then refined iteratively, they are much more likely to have the correct values for the surface. If the parameters are closer to their true value then the calculated elevation map will be closer to its true value as well. Furthermore, the known elevation values provide constraints on the incidence angles that limit accumulation of errors through integration. This again forces the elevation map to be closer to its true value.

In addition to the elevation map of the seafloor the reflection model parameters at every point in the image are determined. This gives a segmentation of the seafloor which can be used to classify seafloor roughness and materials. This algorithm works independently of the reflection model chosen for the seafloor so it can be used to map parts of the ocean with different material properties and in some cases with different sonar sensors.

Figure 1 shows the block diagram for the algorithm. First, sparse bathymetric control points and a given reflection model are used to estimate the reflection model parameters and elevation map every where in the image. Then, dense intensity data is used to iteratively refine the estimates of the reflection parameters and the elevation map. Iteration continues until the elevation values and reflection parameters have converged every where in the image.

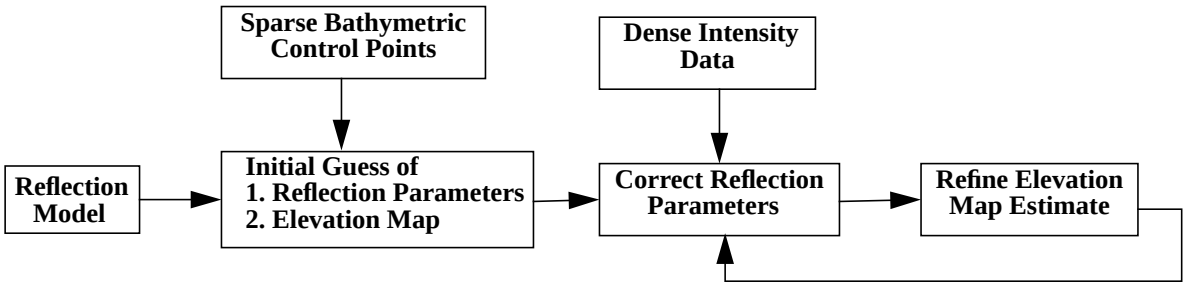


**FIGURE 1. Block Diagram for Map Generation**

The particular sensor from which the data was acquired was an AMS 120 side scan sonar with swath bathymetry. The geometry associated with this sensor is given in Figure 2. The incidence angle is the angle α between the vector from the sensor to the surface **R** and the surface normal **n**. The grazing angle φ is the angle between **R** and the horizontal.This sensor generates an intensity map in decibels and a bathymetry map in meters of the seafloor[4] [5]. The bathymetric map provides the elevation control points as well as a ground truth for verifying the reconstructed elevation map. For the purposes of testing the algorithm on a sparse set of elevation values, only discrete elevation control points were used from the bathymetric image. For simplicity of implementation the sparse elevation control points formed a grid. The grid cell size can be varied to test the algorithms dependence on the density of the elevation control points. Figure 2 also shows the orientation of the images taken with respect to the sonar fish. The fish travels in the y direction and creates a row in the intensity and bathymetric images for each sonar ping. The range to the seafloor is measured along the columns of the images. The size of the pixels in the image are 1.0 meters along the rows and 0.5 meters along the columns.

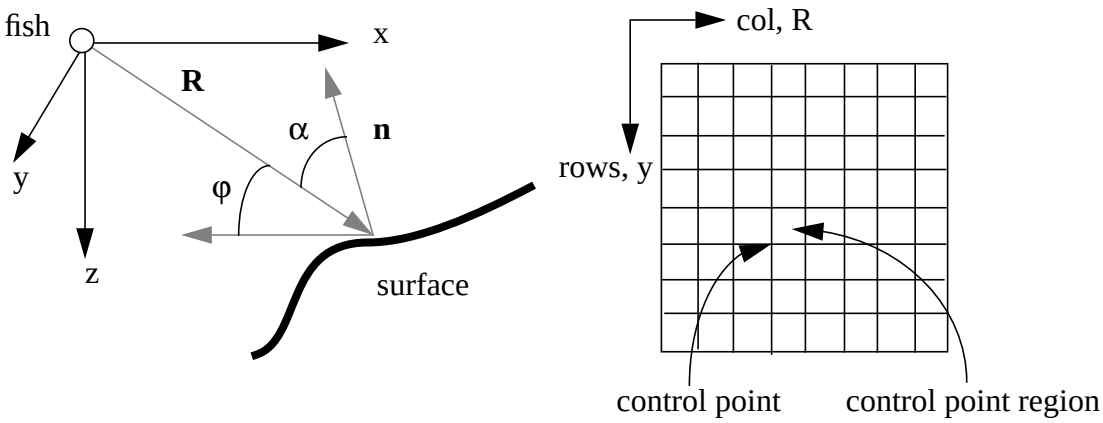


**FIGURE 2. Geometry of a side scan sonar and image orientation with elevation control points.**

In addition to the real data, sets of synthetic data (intensity and bathymetry maps) for a flat ocean bottom were generated for three different reflection models. To generate the images the reflection parameters of the surface had to be specified, so these synthetic images were used to test the ability of the algorithm to correctly estimate the parameters of the reflection models. Noise was added to the images to make them more typical of the images generated by side scan sonars. The noise was gaussian and had a magnitude that grew as the range squared.

The results of the algorithm on a set of real data is given in Figure 3. The map of the albedo shows a high albedo region and a low albedo region.This agrees with the bathymetric and intensity maps and shows that the algorithm is able to segment the seafloor based on reflection parameters.

# 2 Determining the Reflection Model Parameters

The reflection model of a surface is a function of the incidence angle of the signal striking the surface and some set of parameters that depend on the material properties of the surface and the sensor $R_s = R_s(\alpha, p_1, p_2, p_3, \ldots)$. Even if the incidence angle of the surface is know at every point in the image, the expected intensity values predicted by the reflection model cannot be determined because the reflection parameters of the surface are not known. However, given the intensity image of the seafloor, the values of the reflection parameters can be predicted using least squares fitting of the reflection model to the intensity data.

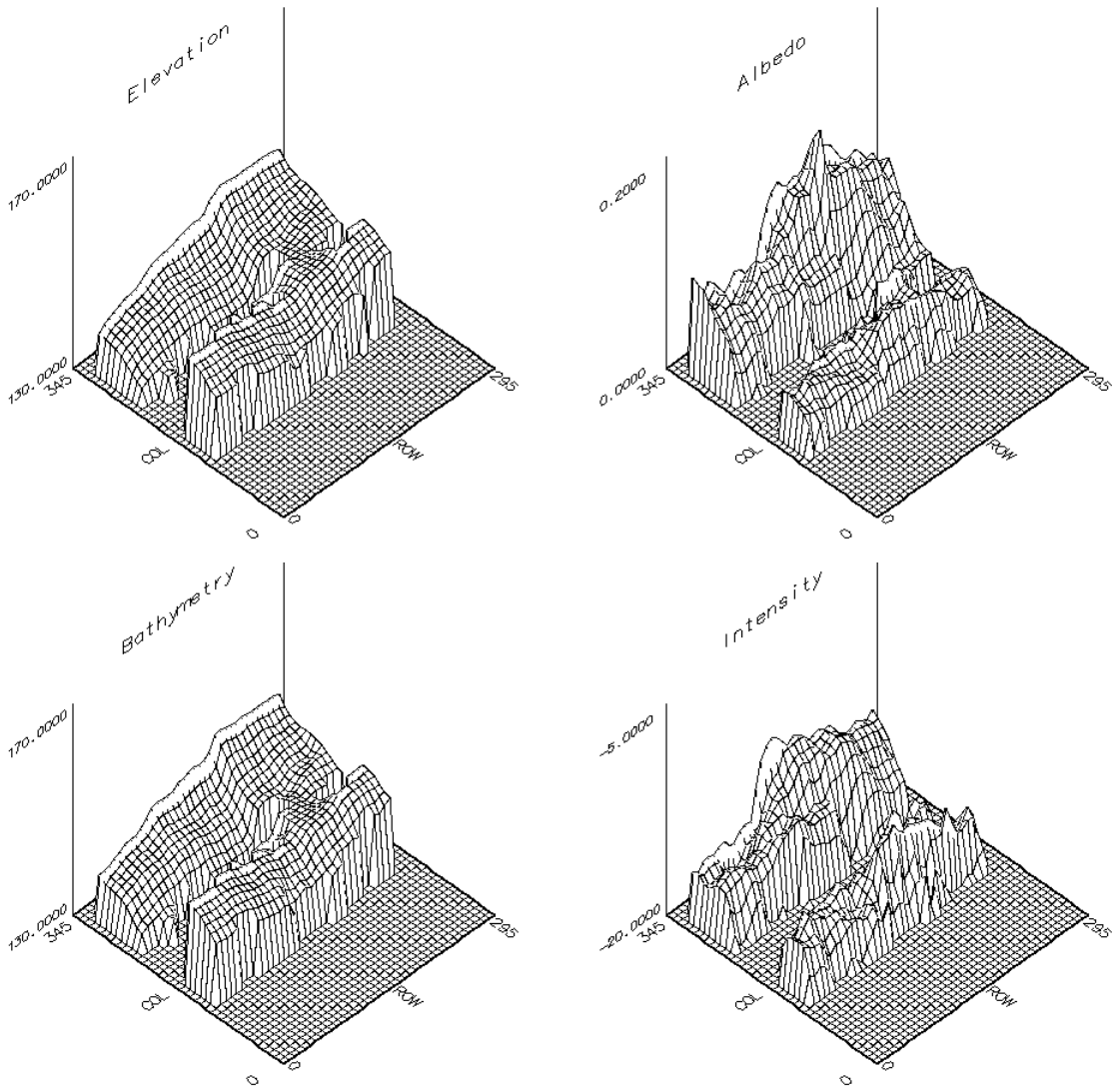


**FIGURE 3. Real bathymetric and intensity images along with reconstructed elevation and albedo maps using the relaxation method and control points spaced every 8 pixels.**

We will assume that the reflection parameters $\mathbf{p} = (p_1,p_2,p_3,...)$ are constant throughout each control point region in order to estimate the reflection parameters. Under this assumption the parameters $\mathbf{p}$ that best fit the data $I_i$ with error $\sigma_i$ in the control point region can be found by minimizing the chi square merit function for fitting the reflection model to the data.

$$\chi^2(\boldsymbol{p}) = \sum_{i=1}^{N} \left[ \frac{I_i - R_s(\alpha_i, \boldsymbol{p})}{\sigma_i} \right]^2 \qquad \text{(EQ 1)}$$

This function was minimized with the Levenberg-Marquadt Method for non-linear least squares minimization[7]. The resulting parameters will be the best fit parameters to the data in the control point region given the reflection model and the intensity data.

A virtue of this fitting method is that different reflection models can be used simply by changing the function supplied to the minimization method. In this way different reflection models can be tested on the data to determine the one that is the most appropriate. In particular three reflection models were tested: The lambertian, Torrance and Sparrow [8] and Diffuse models[6].

Lambertian: $R_s = R_s(\alpha, \rho) = \rho \cos\alpha$

Torrance and Sparrow: $R_s = R_s(\alpha, \rho, \sigma, C) = \rho \cos\alpha \left( (1 - C) + 2Cexp\left( \frac{-\alpha^2}{2\sigma^2} \right) \right)$

Diffuse: $R_s = R_s(\alpha, \rho, \varepsilon) = \rho \left( \cos\alpha + \varepsilon (\sin\alpha)^2 \right)$

In these reflection models, each parameter has a physical meaning: the albedo ρ determines the ability of the seafloor to absorb sound, σ and ε are measurements of the roughness of the seafloor surface and *C* is the constant that determines the relative strengths of the diffuse and specular components of the seafloor.

The first stage in estimating the reflection parameters is to estimate the incidence angles at each point in the image. First, values for φ and elevation are estimated by linearly interpolating between control points. Second, the derivatives of φ with respect to *R* and *y* are determined at each point with local finite differences. Finally, these derivatives define the local surface normal **n** whose inner product can be taken with **R** to get the cosine of the incidence angle at each point. The inverse of the cosine is then taken to get an estimate for alpha at each point.

The reflection model parameters are re-estimated each time a new estimate of the incidence angles is determined from the new elevation map. In the algorithm the incidence angles are converging to their true values, so as a result the estimates of the reflection parameters should converge to their correct value. This is demonstrated by testing done on synthetic data where the true albedo is known. Figure 4 shows the convergence of the albedo for the synthetic data created for a plane that follows the Torrance and Sparrow model of reflectance. The true value of albedo is 0.5 which is the value that the albedo converges to. The final histogram of albedo error as well as the iterative convergence of a single point are given in Figure 5. The convergence of the parameter *C*, where $C = 0.6$, for the same set of data is shown in Figure 6.

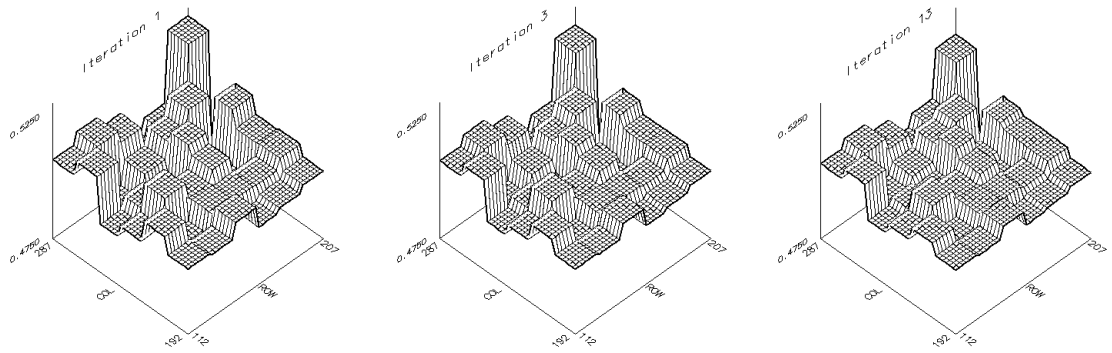


**FIGURE 4. Convergence of albedo for the Torrance and Sparrow reflection model.**

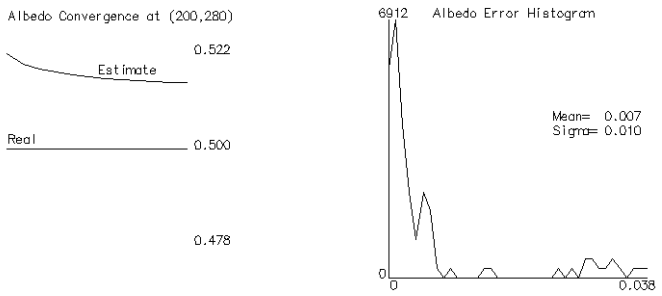


**FIGURE 5. Histogram of albedo error and iterative albedo convergence for one point.**

A shortcoming of this method for parameter estimation is that it cannot always estimate all of the reflection parameters in the reflection model. Fortunately this generally occurs when the parameter contributes little to the form of the reflection model. A specific example of this difficulty occurs when trying to estimate σ in the Torrance and Sparrow

model at large values of α. This is precisely because at large values of α the specular component of reflection is negligible, so, numerically a correct estimate of σ is not easy to obtain.

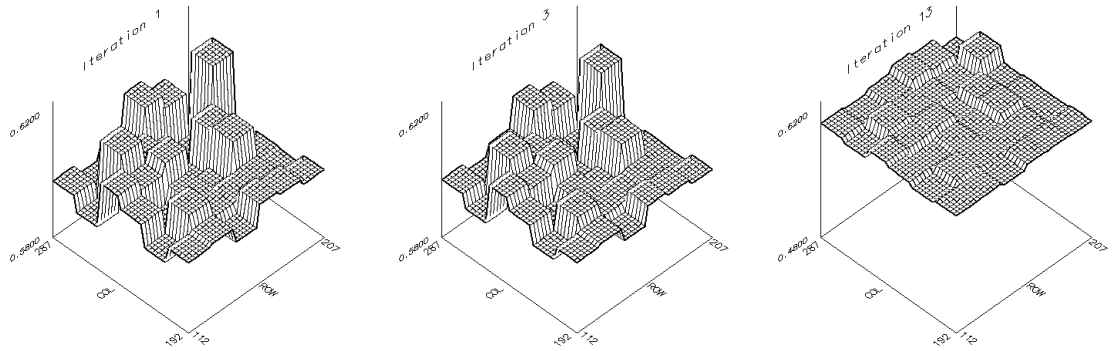


**FIGURE 6. Convergence of the parameter *C* for the Torrance and Sparrow reflection model.**

It is also difficult to estimate parameters correctly when there is a strong coupling between parameters. This coupling means there exists multiple sets of parameters that generate the same reflection model form. The data will fit the reflection model equally well for multiple sets of parameters, so the minimization method has no way to select the correct reflection parameters. However, the form of the reflection model will generally be estimated correctly. An example of this coupling occurs in the diffuse model where ρ and ε are multiplicatively coupled in the roughness term of the model.

As Figure 4 and Figure 6 show, the estimates of the parameters in the reflection model are close to there true values from the very first estimation. This is true for all reflection models as long as the initial estimate of the elevation map is close to its true value. If this is not the case then the reflection models parameters take longer to converge properly. The reflection parameters are considered to be close to there true values when the value of $X^2$ is close to 1.0. When this is true for all control point regions the reflection parameters are considered to be converged for the surface.

# 3 Determining the Elevation Map

During each iteration of the algorithm the elevation at each point needs to be adjusted to bring the shape of the surface closer to that which is predicted by the intensity data. Mathematically it is more convenient to adjust φ at each point in the elevation map than the elevation *z*. This is feasible because φ can be easily converted to *z* with the equation $z = R\sin\varphi$. The first step in determining the elevation map is to create an initial estimate of φ at every point in the elevation map that is bounded by control points. This is done with linear interpolation between three control points. The result is a mesh of triangular planar facets. Once φ is estimated it is iteratively adjusted using one of two methods The first method is based on iteratively adjusting φ so that it will satisfy a partial differential equation derived from a constraint on the surface. The second method uses a relaxation technique to iteratively move φ in a direction that creates a surface geometry that better satisfies the intensity data.

## 3.1 METHOD 1: Finite Differences

Partial Differential Equation

The initial estimate of the surface can be refined by repeatedly solving a PDE that characterizes the surface in each control point region. This PDE is based on the constraint that at every point the normalized dot product of the vector from the sonar fish to the surface **R** and the normal of the surface **n** is equal to the cosine of the angle of incidence.

$$\frac{\boldsymbol{R}\cdot\boldsymbol{n}}{\|\boldsymbol{R}\|\,\|\boldsymbol{n}\|} = \cos\alpha \qquad \text{(EQ 2)}$$

However, **R** and **n** are functions of φ and its derivatives with respect to *R* and *y*, so the constraint can be re written in the form of the PDE

$$(\tan\alpha)^2 \varphi_R^2 - \varphi_y^2 = \frac{1}{R^2}. \quad \text{(EQ 3)}$$

The discrete forward differences version of this equation is

$$\varphi_{r,c+1} = \varphi_{r,c} - \frac{\Delta R}{|\tan\alpha|}\left(\frac{1}{R}^2 - \left(\frac{\varphi_{r+1,c} - \varphi_{r,c}}{\Delta y}\right)^2\right)^{\frac{1}{2}} \quad \text{(EQ 4)}$$

where $\Delta R$ and $\Delta y$ are the pixel sizes in meters in the $R$ and $y$ directions respectively. More details on this derivation are in the Appendix.

For each iteration the new values of $\varphi$ are computed in each control point region by first calculating a value $\varphi_f$ from the discrete forward difference equation as shown in Equation 4. Then another value $\varphi_b$ is calculated by a similar backward difference equation. These two values are then combined to form the new estimate of $\varphi$ with

$$\varphi_{new} = \frac{d_b \varphi_f + d_f \varphi_b}{d_f + d_b} \quad \text{(EQ 5)}$$

where $d_f$ is the distance from the point being adjusted to the nearest control point with a lesser column value and $d_b$ is the distance from the point to the nearest control point with a greater column value. This combination function weights $\varphi$ values that are closer to control points higher to keep accumulation of errors down. The final step in the iteration is to adjust the incidence angle $\alpha$ at every point based on the new values of $\varphi$.

The complete algorithm for surface reconstruction using finite difference is as follows:

1. Linearly interpolate the values of $\varphi$ between control points
2. From the current estimate of $\varphi$ and its derivatives determine a new estimate for $\alpha$.
3. Estimate the reflection parameters as in Section 2 given the $\alpha$ values in each control point region.
4. Determine a new value for $\varphi$ with finite differences.
5. Go to step 2 until the intensity predicted by the reconstructed surface is close to the intensity given by the data.
6. Convert the map of $\varphi$ into an elevation map with $z = R\sin\varphi$.

Figure 7 shows the convergence of elevation for a synthetic image that follows the lambertian form of reflectance with control points spaced every 16 pixels. The image is that of a flat plane with an expected height of 107.5 meters. The final elevation map for the entire surface is given in Figure 8 along with the convergence history of a point in the image and the global elevation error histogram. The figures show that the elevation converges form a rough initial estimate to a the expected value of 107.5 meters in 5 iterations. In most cases the finite differences method went through less than ten iterations to find the correct maps of the seafloor. The periodic bulging that occurs in the elevation map in Figure 8 is caused by accumulation of errors between control points.

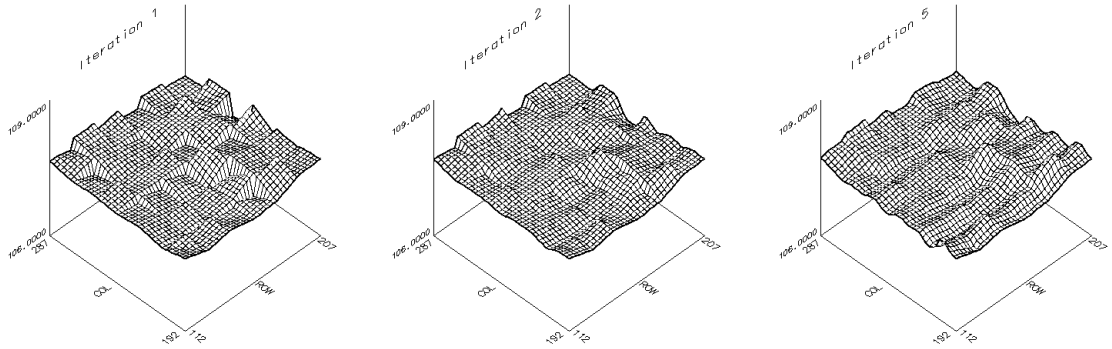


**FIGURE 7. Convergence of elevation using finite differences.**

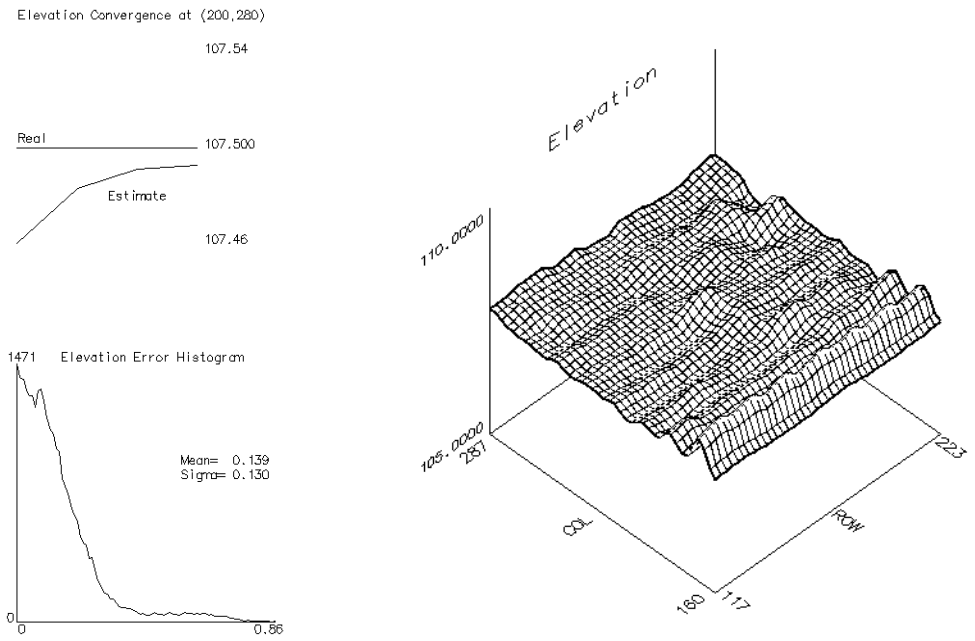


**FIGURE 8. Final elevation error histogram, iterative elevation convergence for one point and final reconstructed elevation map using finite differences on a flat plane.**

In some cases data sets will contain a control point region where the elevation map created by finite differences rapidly diverges from the true surface. This occurs because the initial estimate of the surface is too far from the correct shape. As a result, accumulation of errors will force the surface farther and farther from the correct value. The reflection parameters predicted in these control point regions will be incorrect as well. To deal with this problem the regions where the blow up was occurring were detected by looking for large changes in φ. Once detected, these regions were constrained to stay close to the initial estimate of the surface as given by linear interpolation of the control points. This prevented extremely incorrect elevation values and reflection parameters from being created which gave the method a better global convergence. In all data sets this rapid divergence occurred in at most two control point regions.

## 3.2 METHOD 2: Relaxation

The second method for updating the elevation map uses a relaxation technique similar to that used in shape from shading in computer vision [1]. In this method the iterations occur in two stages. The first stage updates the derivatives of φ and the parameters of the reflection model. The second stage updates φ itself.

Given the reflectance model the best estimates of $\varphi_R$ and $\varphi_y$ satisfying smoothness constraints can be found by minimizing the integral

$$\iint_{yR} I(R,y) - R_s(\varphi_R, \varphi_y)\ )^2 dRdy + \lambda \iint_{yR} \left( \varphi_{RR}^2 + \varphi_{Ry}^2 + \varphi_{yR}^2 + \varphi_{YY}^2 \right) dRdy \qquad \text{(EQ 6)}$$

where the first integral insures that the data matches the reflectance model in the least squares sense and the second integral ensures that the reconstructed elevation map varies smoothly. Some of the variation from pixel to pixel in the intensity and bathymetric data is caused by high frequency noise and is physically infeasible. Therefore, the smoothness term is added to the integral to keep the noise out of the reconstructed map by forcing adjacent values of φ to values that are close to each other. The amount of noise in the data is not known so, the parameter λ is used to weight smoothness against goodness of fit.

In the discrete case the update equations for the derivatives that minimize Equation 6 can be found using Euler's equation from the calculus of variations and the discrete derivatives.

$$\varphi_R^{n+1}(r,c) = \bar{\varphi}_R^n(r,c) + \lambda\left(I(r,c) - R_s\left(\varphi_R^n(r,c), \varphi_y^n(r,c)\right)\right)\frac{dR_s}{d\varphi_R}$$
$$\varphi_y^{n+1}(r,c) = \bar{\varphi}_y^n(r,c) + \lambda\left(I(r,c) - R_s\left(\varphi_R^n(r,c), \varphi_y^n(r,c)\right)\right)\frac{dR_s}{d\varphi_v} \quad \text{(EQ 7)}$$

The bar represents an average in a 3x3 neighborhood of the point. The incidence angle $\alpha$ is a function of $\varphi_R$ and $\varphi_y$ because as is shown in the Appendix

$$\alpha = \operatorname{acos}\left(\frac{R\varphi_R}{\sqrt{R^2\left(\varphi_R^2 + \varphi_Y^2\right) + 1}}\right). \quad \text{(EQ 8)}$$

This expression can be substituted into any reflection model $R_s$ to determine the derivatives of $R_s$ with respect to $\varphi_R$ and $\varphi_y$. The derivatives of $\varphi$ determine $\alpha$ which in turn determines the value predicted by the reflection model. Hence, only the derivatives of $\varphi$ need to be adjusted along with the reflection model parameters to bring the intensity predicted by the reflection model close to that predicted by the data. Once $\varphi_R$ and $\varphi_y$ are done being adjusted the reflection model parameters are determined because $\alpha$ has become fixed.

The convergence of $\varphi_R$ for a synthetic image of a plane with the lambertian reflection model and control points spaced every sixteen pixels is shown in Figure 9. $\varphi_R$ is expected to be constant along the rows while increasing along the columns of the image. This is what the final image shows.

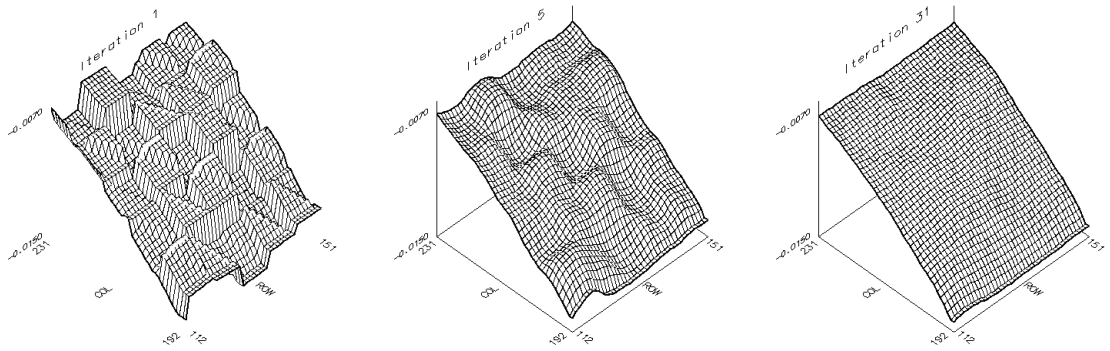


**FIGURE 9. Convergence of $\phi_R$ using the relaxation method.**

Once $\varphi_R$ and $\varphi_y$ are fixed $\varphi$ can be determined from them by minimizing

$$\int\int_{\phi R}\left(\left(\frac{d\varphi}{dR} - \varphi_R(R,y)\right)^2 + \left(\frac{d\varphi}{dy} - \varphi_y(R,y)\right)^2\right)dRd\varphi. \quad \text{(EQ 9)}$$

In the discrete case this integral can be minimized using Euler's equation and discrete approximations to the derivatives to give the following update rule:

$$\varphi^{n+1}(r,c) = \varphi^n(r,c) + h\left(\varphi_R^n(r,c) - \varphi_R^n(r,c-1) + \varphi_R^n(r-1,c-1) - \varphi_R^n(r-1,c)\right) + h\left(\varphi_y^n(r,c) - \varphi_y^n(r,c-1) + \varphi_y^n(r-1,c-1) - \varphi_y^n(r-1,c)\right) \quad \text{(EQ 10)}$$

where $h = \Delta R\Delta y$ is the size of a pixel in meters squared.

Given a good initial estimate of the surface, completing these two stages of iteration will result in a solution for $\varphi$ which can be converted into an elevation everywhere in the image.

The complete surface reconstruction algorithm for the relaxation technique is as follows:

For each control point region.

1. Linearly interpolate the values of φ between control points.
2. Determine estimates of $\varphi_R$ and $\varphi_y$ between control points as finite differences of φ as estimated in step 1.
3. Given $\varphi_R$ and $\varphi_y$ determine a new estimate for α with Equation 8.
4. Estimate the reflection parameters as in Section 2 given the α values in each control point region.
5. Determine new values for $\varphi_R$ and $\varphi_y$ with Equation 7.
6. Go to step 3 until the intensity predicted by the reconstructed surface is close to the intensity given by the data.
7. Iteratively adjust φ with Equation 10 until φ changes very little between interactions.
8. Convert the map of φ into an elevation map with $z = R\sin\varphi$.

In Figure 10 the convergence of elevation is shown for the same set of synthetic data as was used in Figure 9. It also converges to the shape expected for a flat plane with an elevation of 107.5 meters. Figure 11 shows the histogram of elevation error and the iterative convergence of elevation at a single point in the image. It also shows the final reconstructed elevation map for the entire surface.

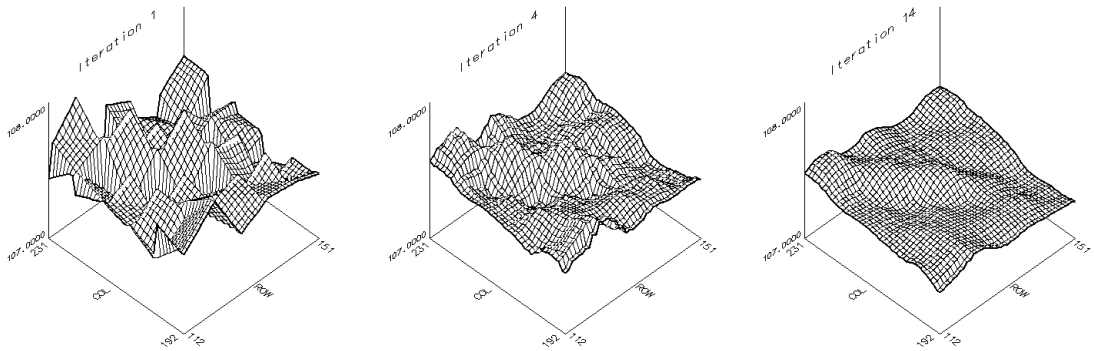


**FIGURE 10. Convergence of elevation using the relaxation method.**

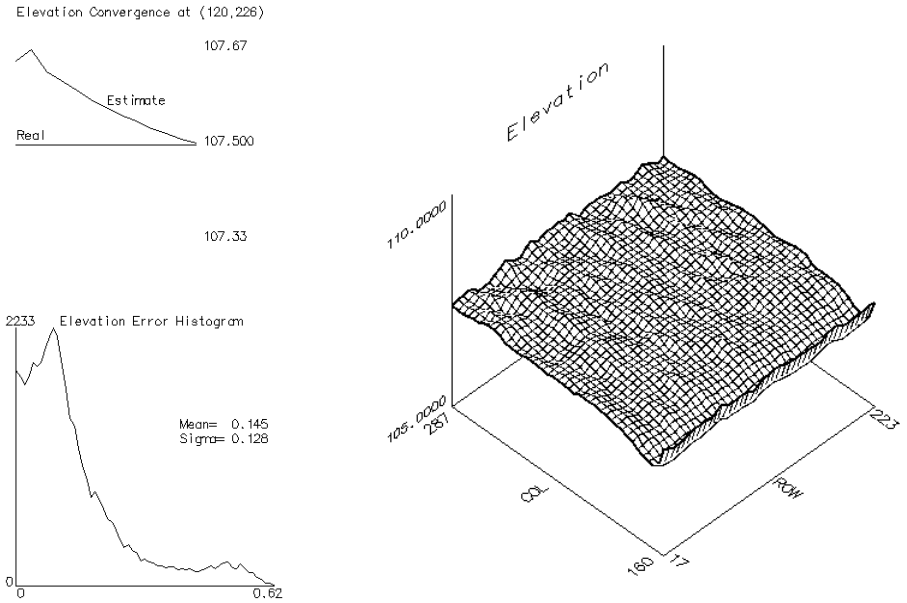


**FIGURE 11. The final elevation map, elevation error histogram and iterative convergence of elevation for a point using relaxation.**

# 4 Surface Reconstruction

Both of the methods for surface reconstruction iteratively adjust an initial elevation estimate of the seafloor. This adjustment is based on incidence angles predicted by a reflection model whose parameters are also iteratively adjusted. As long as the initial estimate of the shape of the surface is close to its actual form, the initial estimates of the reflection model parameters do not have to be close to their true values for them to converge correctly.

The convergence of the elevation maps and the reflection parameters also depends on the spacing of the control points. Table 1 shows the global mean, variance and maximum elevation error for a set of real data as a function of control point spacing for both of the methods investigated. As spacing increases the reconstructed surface becomes a worse match to the true bathymetric surface. However, the methods will produce a result as long as the spacing of the control points is smaller than the bounds of the image data.

**TABLE 1. Global elevation error at different control point spacings using real data.**

| Method | Spacing | Mean | Standard Deviation | Max |
|---|---|---|---|---|
| Relaxation | 8 by 8 | 0.284 m | 0.241 m | 3.40 m |
| Relaxation | 16 by 16 | 0.389 m | 0.486 m | 4.36 m |
| Relaxation | 24 by 24 | 0.566 m | 0.482 m | 2.86 m |
| Finite Differences | 8 by 8 | 0.128 m | 0.193 m | 3.66 m |
| Finite Differences | 16 by 16 | 0.399 m | 0.435 m | 3.31 m |
| Finite Differences | 24 by 24 | 0.475 m | 0.594 m | 6.68 m |

In Figure 12 the reconstructed surface and estimated albedo created with the finite differences method are shown for the real set of data where the lambertian reflectance model and a control point spacing of 16 pixels was used. In Figure 13 the reconstructed surface and albedo using the relaxation method, lambertian reflection model and control point spacing of 16 pixels are shown for the same set of real data. As Table 1 and Figure 14 show the methods have similar global and local convergence properties. For the most part this means that the methods can be used interchangeably as iterative elevation update methods.

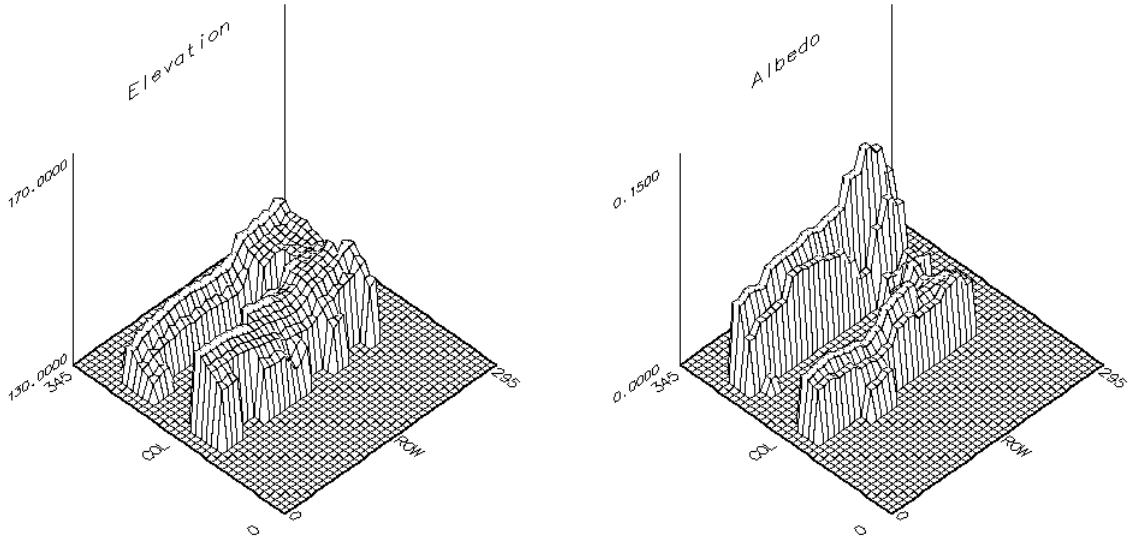


**FIGURE 12. The reconstructed elevation and albedo maps for a real set of data using the finite difference method.**

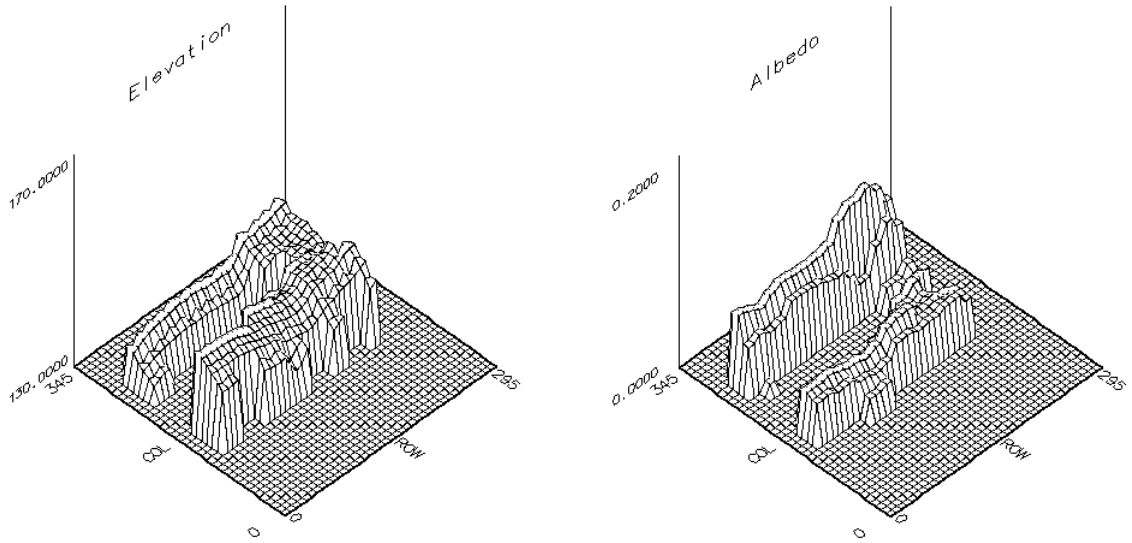


**FIGURE 13. The reconstructed elevation and albedo maps for a real set of data using the relaxation method.**

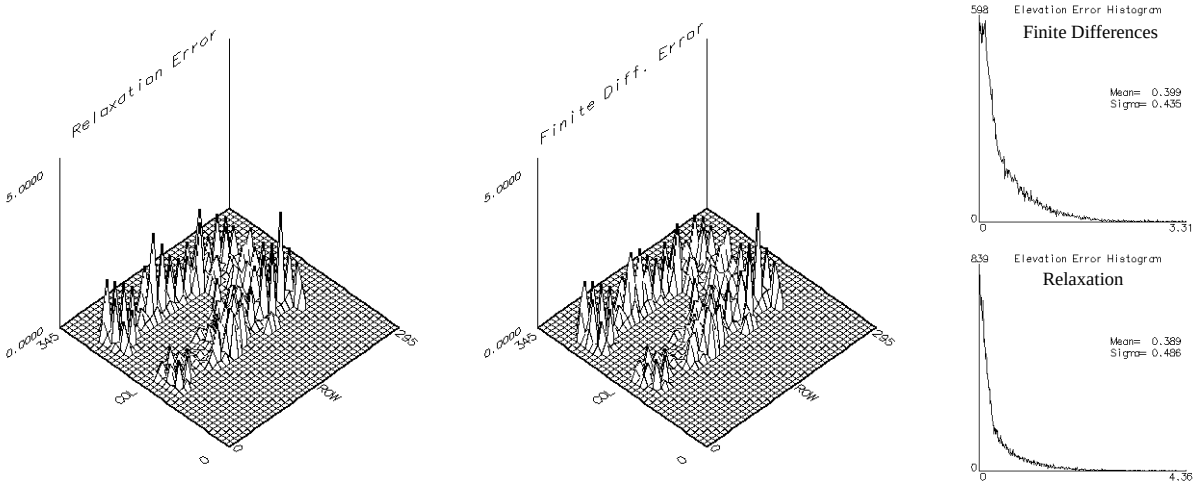


**FIGURE 14. The error maps on elevation for the reconstructed surfaces for the relaxation method and the finite differences method along with their elevation error histograms.**

Furthermore, both methods exhibit segmentation of albedo on real data. This is shown in the reconstructed albedo maps because the albedo on the right side of the map is higher than the albedo on the left side. This is expected from the real data because while the bathymetric data has a constant profile along the rows of the image, the intensity is not constant but decreases to the left of the image. This decrease in intensity could very likely be associated with a decreasing in albedo of the seafloor material. This decrease in albedo is predicted by the albedo map in both methods of surface reconstruction. Determining seafloor material properties, like albedo, is important because this information can be used to determine the material components of the ocean floor using sonar. For instance, sand and rock have different albedos, so determining the albedo everywhere in the image could show where the rock and sand seafloor types occur in the image.

One advantage of the relaxation technique is that it produces smooth elevation maps while the finite differences method produces elevation maps that are not as smooth. This is because the relaxation technique incorporates a

smoothness term into its iterations and the finite differences method does not. However, the finite differences methods does take fewer iterations than the relaxation technique because it only has one stage of iterations.

# 5 Conclusions

An algorithm that generates maps of the elevation and reflection model parameters from a side scan sonar intensity image and sparse bathymetric data of the seafloor has been presented. This algorithm works independently of the reflection model used. In regions bounded by known elevation values it determines the best set of reflection parameters given the reflection model using non-linear least squares fitting. It can also accommodate two different forms of iterative elevation update. One method is based on iteratively solving a partial differential equation using finite differences until the elevation surface converges to its true value. The other method uses a relaxation technique to iteratively solve for the derivatives of the elevation map based on matching the intensity data to the reflection model. It then iteratively adjusts the elevation map to correspond to the derivative values converged to in the previous stage of iterations.

The algorithm has been shown to correctly predict elevation and reflection parameter values for synthetic data using the lambertian, Torrance and Sparrow and Diffuse reflection models. It has also been shown to build accurate elevation maps and segment albedo where appropriate on real data. Furthermore, the algorithm relies on an initial estimate of the seafloor structure generated by linear interpolation between known elevation values. If the initial estimate is sufficient then the algorithm will iteratively adjust the reflection parameters to their best fit positions in the least squares sense regardless of the initial starting values for the parameters.

A future improvement to the algorithm will be to add a stage where the intensity and bathymetric data are corrected for the movement of the side scan sonar fish (assuming this motion is known). Another improvement will be to allow the control points to have any position in the intensity image. To implement this the Delaunay Triangulation of the control points will be determined and the triangulated regions will be used as control point regions just like the grid cell between control points are being used know. Along the same lines, a more intelligent way of determining the regions in which the reflection model parameters are held fixed will be investigated. This will be done by spliting regions apart that have a bad measure of fit to their reflection parameters into regions where the goodness of fit is improved for each individual region. To minimize numerical problems regions of similar reflection parameters will be grouped together to increase the number of data points available for minimization.

# References


[1] Berthold Klaus Paul Horn. Robot Vision. MIT Press, 1986.

[2] Erwin Kreyszig. Differential Geometry. Dover Publications, 1959.

[3] Dirk Langer & Martial Hebert. Building Qualitative Elevation Maps From Underwater Sonar Data for Autonomous Underwater Navigation. In Proc. IEEE Conference on Robotics and Automation, 1991.

[4] Sandipa Malik. Quantitative Seafloor Backscatter Characterization Using an Interferometric Sidescan Sonar. Master's Thesis, University of Virginia, 1991.

[5] Charles Mazel. Side Scan Sonar Record Interpretation. Klein Associates, 1985.

[6] Michael Oren and Shree K. Nayar. Diffuse Reflection Model for Rough Surfaces. Dept. of Computer Science Technical Report 057-92, Columbia University, 1992

[7] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling. Numerical Recipes in C. Cambridge University Press, 1988.

[8] K. E. Torrance and E. M. Sparrow. Theory for Off-Specular Reflection from Roughened Surfaces. Journal of the Optical Society of America, 57, 1967.

[9] R. Urick. Principles of Underwater Sound. McGraw-Hill, 1983.

# Acknowledgments

We would like to thank Ken Stewart of Woods Hole Oceanographic Institution for providing the sonar data and Hanumant Singh of Woods Hole and Dirk Langer of the Robotics Institute for answering many questions about accessing and using the data.


# Appendix: Derivation of the Finite Differences PDE

The surface **F** that is going to be reconstructed can be parameterized in *R* and *y* by converting to cylindrical coordinates.

$$\mathbf{F}(R, y) = \begin{bmatrix} x(R, y) \\ y(R, y) \\ z(R, y) \end{bmatrix} = \begin{bmatrix} R\cos(\varphi(R, y)) \\ y \\ R\sin(\varphi(R, y)) \end{bmatrix} \qquad \varphi = \varphi(R, y) \tag{EQ 11}$$

Assuming the surface is explicit, at any point *(R,y)* the normal to the surface is given by[2]

$$\hat{n} = \frac{d\mathbf{F}}{dR} \times \frac{d\mathbf{F}}{dy}. \tag{EQ 12}$$

Differentiating Equation 11 with respect to *R* and *y* gives

$$\frac{\partial \mathbf{F}}{\partial R} = \begin{bmatrix} \cos\varphi - R\sin\varphi(\varphi_R) \\ 0 \\ \sin\phi + R\cos\phi(\varphi_R) \end{bmatrix} \qquad \frac{d\mathbf{F}}{dy} = \begin{bmatrix} -R\sin\varphi(\varphi_y) \\ 1 \\ R\cos\varphi(\varphi_y) \end{bmatrix} \tag{EQ 13}$$

where $\varphi_R$ and $\varphi_y$ are the derivatives of φ with respect to *R* and *y*, so

$$\boldsymbol{n} = \begin{bmatrix} -\sin\varphi - R\cos\varphi(\varphi_R) \\ -R\varphi_y \\ \cos\varphi - R\sin\varphi(\varphi_R) \end{bmatrix}. \tag{EQ 14}$$

Figure 2 shows that the constraint

$$\boldsymbol{R} \cdot \boldsymbol{n} = \|\boldsymbol{R}\| \|\boldsymbol{n}\| \cos\alpha \tag{EQ 15}$$

must be true. Given

$$\mathbf{R} = \begin{bmatrix} R\cos\varphi \\ 0 \\ R\sin\varphi \end{bmatrix} \qquad \|\boldsymbol{n}\| = \sqrt{R^2\left(\varphi_R^2 + \varphi_Y^2\right) + 1} \tag{EQ 16}$$

Equation 15 becomes

$$-R\varphi_R = \left(\sqrt{R^2\left(\varphi_R^2 + \varphi_Y^2\right) + 1}\right)\cos\alpha \tag{EQ 17}$$

which can be rewritten as the partial differential equation

$$(\tan\alpha)^2\varphi_R^2 - \varphi_Y^2 = \frac{1}{R^2}. \tag{EQ 18}$$

The discrete forward differences version of Equation 18 is

$$\varphi_{r,c+1} = \varphi_{r,c} - \frac{\Delta R}{|\tan\alpha|}\left(\frac{1}{R}^2 - \left(\frac{\varphi_{r+1,c} - \varphi_{r,c}}{\Delta y}\right)^2\right)^{\frac{1}{2}}. \tag{EQ 19}$$

The discrete backward differences equation is similar.