Streamlining the Evaluation of Potential Marine Debris Targets for Disaster Response

Thomas Butkiewicz and Andrew H. Stevens
Center for Coastal and Ocean Mapping
University of New Hampshire

***Abstract*—Hurricanes generate massive amounts of marine debris in coastal waterways, causing environmental problems and hazards to safe navigation. Before debris can be removed, they must be located and identified, a tedious and time-consuming process. Automatic target recognition algorithms speed up this process by searching vast swaths of survey data to locate anomalies that could be debris. However, these targets must still be manually evaluated by analysts. We present a tool that increases analysts' efficiency by automating most of the interaction steps involved in marine debris evaluation. It automatically generates and displays multiple optimal views, which can remove the need for analysts to manually reposition their viewpoint. We also applied these optimal views to a web-based crowdsourcing application, which tested whether the general public could assist in marine debris evaluation. Our results showed that, even though the participants were untrained, their collective decisions were generally in agreement and reliable enough to greatly reduce the number of marine debris targets that must be evaluated by the limited pool of trained analysts. This has the potential to increase analytical capacity in time-critical disaster response situations.**

Keywords—marine debris; visualization; visual analysis; crowdsourcing

I. Introduction

In 2012, "Superstorm" Sandy became the 2nd costliest hurricane in U.S. history. Damage occurred primarily in the coastal zone, as flooding, high winds, and powerful waves destroyed buildings, roads, vehicles, etc. along hundreds of miles of coastline. Unlike the damage on land, the storm's impact on areas below sea level (i.e., debris and seafloor changes that can create navigational hazards) is more difficult to assess.

As part of a larger effort to improve the detection and identification of marine debris resulting from Superstorm Sandy, this project addresses the analytical process of determining whether potential marine debris targets are either natural features or actual debris that may need to be removed.

After a storm, surveys are run of the coastline, using technologies such as multi-beam sonar and airborne lidar. The data from these surveys is then processed through automatic target recognition algorithms, which identify anomalies that could be marine debris. Large numbers of these potential debris targets then need to be individually evaluated by analysts, who filter the list of targets into cleanup lists for salvage vessels. This task is currently a tedious and time consuming process.

We have created a pair of tools that speed up and distribute this analytical process: a rapid decision tool and a crowdsourcing website. Together, they can increase analysts' efficiency and decrease disaster response times.

Current methods for evaluating potential marine debris generally follow a common workflow: load a survey dataset, navigate to a target, iteratively reposition the camera to view the target from enough angles to identify it, record the decision, and then navigate to the next target and repeat.

To enhance the efficiency of trained analysts, our Marine Debris Rapid Decision Tool streamlines their workflow by removing as many navigation and interaction actions as possible. It accomplishes this by automatically generating multiple optimal views of each target, which are displayed simultaneously, hopefully revealing all of the necessary shape information required to register a decision without any manual navigation between targets or repositioning of the camera.

There are a limited number of trained analysts, and their workload capacity can be insufficient in time-critical disaster response scenarios. We explored the possibility of crowdsourcing marine debris analysis by creating a website which invites members of the public to assist in evaluating potential debris targets. The website utilizes the same strategy of presenting participants with multiple optimal views, using images generated by the rapid decision tool.

II. Related Work

A. Finding Optimal Views

Our Marine Debris Rapid Decision Tool is based on the concept of presenting users with multiple views of each target object. To ensure the views presented are enough to differentiate between marine debris or a natural features, we must determine the most optimal views of the target. Many techniques [1] have been developed in the field of computer vision to extract and reconstruct 3D shape information from multiple 2D images. However, there has been less study of techniques for doing the reverse, i.e. choosing 2D views of a 3D object that effectively convey its shape.

Of these techniques, many focus on the generation of *representative views*, which are then used later for object recognition and searching for objects by image.

978-0-933957-43-5 ©2015 MTS

For example, Mokhtarian and Abbasi [2] use a combination of curvature scale space (CSS) images, Fourier descriptors, and moment invariants to obtain optimal views of a 3D object for use in shape representation and recognition applications. This method automatically adjusts the number of optimal views based on the complexity of the object being represented and was shown to have a high rate of success in object-matching tasks.

Unfortunately, these techniques are not ideal for our purposes. They generally assume a single object as input, and generate their views based on the objects entire silhouette as viewed from various angles. However, with our snippets of bathymetry data, if there is an object, it is embedded within a patch of seafloor, and thus without somehow segmenting the target object itself from the seafloor, these techniques would be considering mostly the seafloor or snippet boundaries. To avoid confusion, a distinction must be made between the external silhouette of an object's mesh, as used in many of these techniques, and the often internal silhouette edges within the view of a mesh on which our technique focuses on. In fact, in our snippets of bathymetry, often most of the external silhouette of the mesh is comprised of boundary edges, which we ignore entirely.

An interesting approach to finding optimal views that might be adaptable to support our specific marine debris needs is the concept of *view stability*. Yamauchi et al. [3] combine measures of saliency and stability to generate optimal views of 3D models, mainly for working with shape repositories. Their method captures a series of 2D views by placing the object-origin-facing camera at discrete points around a spherical graph that bounds the 3D object. Each 2D view is compared to adjacent 2D views by feeding the image silhouettes into a rotation-, scale-, and translation-invariant analysis method to quantify their similarities. Views with high levels of similarity to all of their adjacent views are said to be stable; groups of stable views are then partitioned into stable view regions. The views contained in these stable regions are then evaluated using a saliency technique, after which a representative view of the stable region is chosen using the saliency-weighted centroid.

Similarly, Lee et al. [4] provide a method for finding an optimal set of views for 3D human face models using silhouettes. A viewing sphere bounding the face is discretely and uniformly sampled, after which the sample silhouettes are evaluated and merged into clusters of similar viewpoints. The weighted centroids of these view clusters form a set of ideal views which are reconstructed using a model-based technique that yields a subset of views representing the optimal and minimal multi-view configuration for representing the 3D shape of a human face.

Another metric that can be used to pick optimal views is the concept of *viewpoint entropy*. This information theory concept refers to the amount of geometric information a view provides. Vázquez et al. [5] used viewpoint entropy as their view scoring metric, measuring both the number of faces shown, as well as their projected area. This metric, along with a greedy view selection algorithm was used to minimize the number of views necessary for image-based rendering. This performed well for them, as their goal was a minimum number of optimal views covering all visible polygons of the object's surface mesh. However, as our meshes were "draped sheets" generated from gridded bathymetry data, any sufficiently high angle view would reveal all faces of the mesh, thus necessitating a more complex metric.

B. Crowdsourcing Science

Crowdsourcing data on the Internet has become an increasingly useful tool to researchers in many domains over recent years. Wikipedia is the classic success story of a crowdsourcing system created during the nascent years of web-enabled publicly-collaborative systems, with approximately 35 million articles and 75 thousand active editors across 250 languages at this time [6].

One evaluation of current crowdsourcing systems enumerates the key challenges faced by these approaches, the most critical of which is recruiting and retaining users [7]. This is not surprising, as the resources available to incentivize users to participate are typically very limited. Thus, crowdsourced scientific research must often rely upon word of mouth and the public perception of the work's relevance.

With respect to "crowd wisdom", the Galaxy Zoo project is a model system for collecting image categorization data as well as engaging the public in citizen science [8]. Their custom-built system displayed randomized images of galaxy formations taken from the Sloan Digital Sky Survey to participants, who then selected one of a handful of morphological categories for the galaxy. An important point noted by the researchers is to be aware of potential biases introduced by the images you display; an evaluation of the study uncovered a learning effect that biased users towards a particular morphological category if the galaxy image was a certain hue.

Some researchers harness the power of existing frameworks, such as Amazon's Mechanical Turk, to crowdsource empirical data for human perception experiments [9]. While these types of experiments require additional caution in their design, the data garnered from such experiments has been shown to be valid, supporting their use as a viable means of conducting research. Mason and Suri [10] provide an excellent analysis of Mechanical Turk's suitability for behavioral studies in comparison with more traditional subject pools, as well as discussion of data quality issues.

Crowdsourcing has been used to respond to time-critical disasters and crisis. Liu [11] developed a Crisis Crowdsourcing Framework (CCF) for designing interfaces for coordinating crowdsourcing to disaster response. In terms of the CCF, our marine debris application fits best with its *tagging* task category, as it addresses the problems of having too much data that is tedious to process and insufficient processing capacity during a crisis. A disaster response project that successfully leveraged the crowds for a similar task was Mission 4636 [12] in the aftermath of the 2010 Haiti earthquake, which allowed people to tag areas with humanitarian needs using SMS text messaging.

The CCF categorizes crowdsourcing projects by different types of interaction flows and engagement strategies. Our application is an example of *parallel sourcing* in which many users perform the same tasks, and their decisions are aggregated to extract collective decisions. This method allows for consistent measures of data quality, and has been successful in projects such as the USGS's "Did You Feel It?" application [13], which crowdsources Earthquake reporting.

The CCF framework is applied in the development of the "iCoast – Did the Coast Change?" [14] application, which allows the public to view aerial photographs of storm-affected coastline and tag any geomorphological changes they see. The change data collected is then used to validate their predictive coastal erosion models.

III. Marine Debris Rapid Decision Tool

The Marine Debris Rapid Decision Tool (MDRDT), shown in Figure 1, streamlines the marine debris evaluation analytical workflow by eliminating or automating as many analyst interactions as possible.

A. Interface Design and Input

Traditionally, marine debris target evaluation begins with the analyst loading a survey dataset, then iteratively navigating to and examining each target's location within the dataset. Instead of loading whole survey datasets, our tool loads individual targets. The tool automatically displays targets one after another, removing any need for navigation interactions between multiple targets in a dataset.

The targets themselves are generated upstream by whichever automatic target recognition algorithm is processing the survey data. Each time it detects a potential debris target, a file is created that contains a snippet of the survey data around that target. The basic version of this (which will be used throughout this paper) is bathymetry data in the form of a rasterized regular grids, stored in .BAG files (Open Navigation Surface's Bathymetry Attributed Grid).[15] However, supplementary data could be saved as well, including backscatter, raw point clouds, uncertainty values, etc. Exporting targets for the tool to ingest is simple, as the minimum data required is only raw bathymetry or a point cloud, and not any meta-data or other descriptors.

We did not define a set extent for how much surrounding data should be saved along with each target. The ideal amount would be "the minimum necessary to compare the target to its immediate surroundings." In practice, a value of roughly 15 meters away from the bounding box of the target works sufficiently in most cases.

When the user selects a directory, the tool scans it to find all possible target files contained within. The bathymetry data in each file is loaded and converted into a triangular mesh. For data in regular grids, this is simply done using triangle strips. Non-gridded data, such as point clouds, can be converted to a mesh using Delaunay triangulation, for example with the Triangle library.[16] These meshes are then used both for display purposes, as well as the calculation of optimal views. Once targets have been loaded and transformed into triangular meshes, the next step is to determine the optimal views.

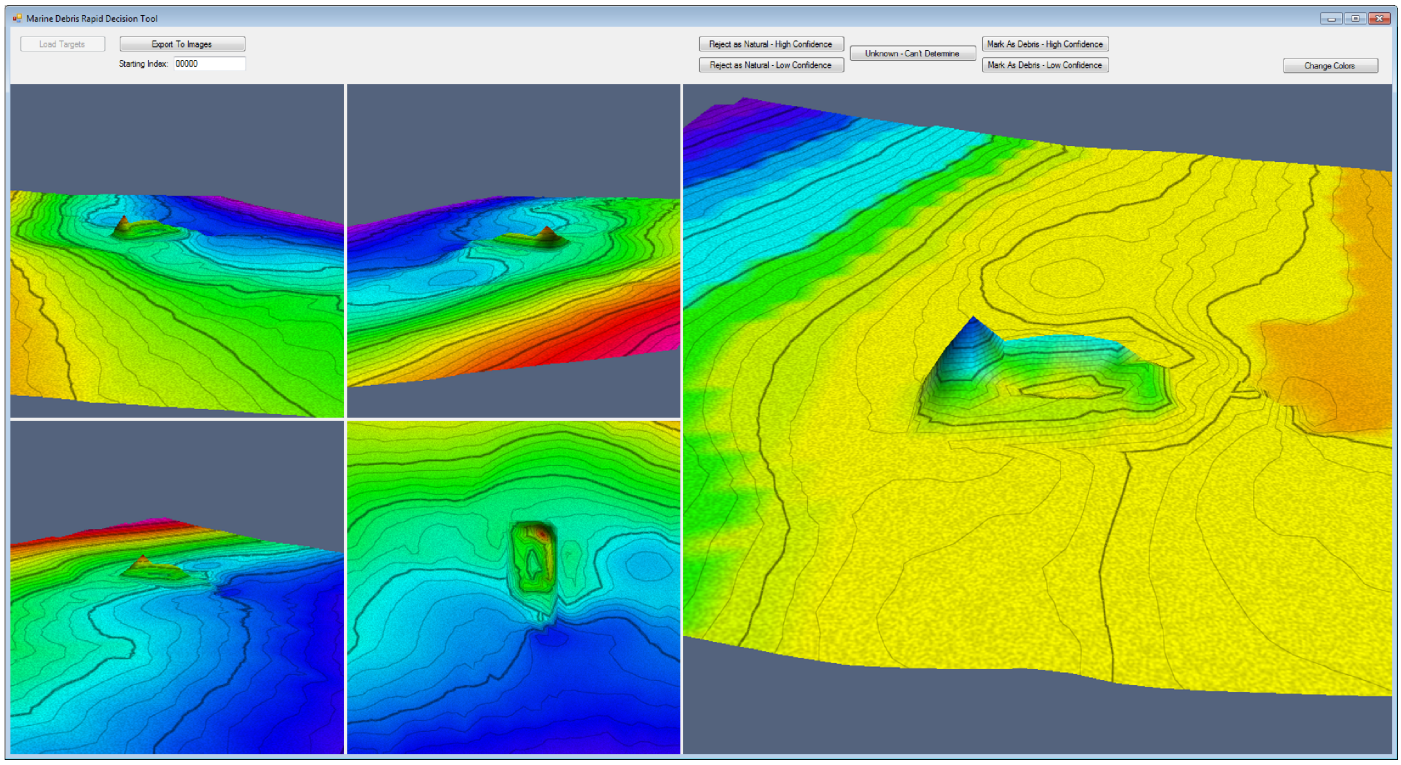

Fig. 1. Screenshot of the Marine Debris Rapid Decision Tool.

The use of multiple optimal views can save an enormous amount of analyst time. Traditional, single-perspective interfaces require users to repeatedly move and rotate the camera to view a target from multiple angles. Our tool instead provides multiple viewing windows, each of which automatically show different, optimized views of the debris object. Ideally, for the majority of cases, looking over the automatically chosen views will provide enough information to make a decision, and thus the analyst will not have to waste time repositioning the camera at all.

If the automatically chosen views are not sufficient, the analyst is able to manipulate each of them as necessary to get a better view. Each are fully-functional instances of our Virtual Test Tank 4D software, which is a 3D/4D interactive geospatial visualization and analysis package based on our previous research projects regarding visual analysis of datasets such as dynamic ocean flow simulations and sediment transport models.

B. Finding Optimal Views

The multiple optimal views are chosen based on a scoring system. The tool calculates how the triangle mesh will appear as viewed from a wide range of possible camera locations and viewing angles. It considers viewing from a full 360˚ around the target, and a range of inclinations from 15˚ to 75˚ from the horizontal. Less than 15˚ inclination is generally too oblique to be useful, and greater than 75˚ is too close to the 90˚ top-down view which is described later. Candidate views can be considered at various granularities across these angle ranges, depending on how quickly a result is desired. For example, splitting the 0˚-360˚ viewing direction into 30˚ sections will be much faster than 5˚ sections, as it cuts the number of calculations by a multiple of 6. In practice, it's best to use a viewing direction increment no higher than 30˚ and an inclination increment no higher than 15˚ to ensure that the candidate views tested are sufficiently representative of all possible views.

For each of these candidate views, a score is computed based on how many features it reveals (using metrics discussed in the following paragraphs). The candidate view with the highest score is selected, and its viewing angles are saved as the first optimal view. Next, all of the remaining candidate views are re-scored. This is done to remove any credit for features already revealed in the previously picked views. The highest scoring view is again saved, and this process continues until the desired number of optimal views is reached.

The primary metric we use to score views is based on the concept of silhouette edges. In 3D computer graphics, a silhouette edge is an edge within a triangle mesh that, for a particular camera view-point, is shared between a front-facing (visible) triangle and a back-facing triangle. They are often used in non-photorealistic rendering to illustratively emphasize shape features. See Figure XXX for an example of how they can be used to emphasize ridgelines when highlighted on a terrain model.

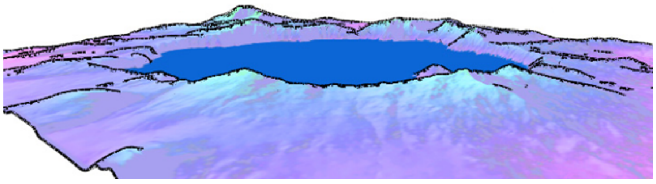

Fig. 2. Example of silhouette edges being used to emphasize ridgelines in a non-photorealistic terrain rendering.

In geospatial and terrain type models, these silhouette edge are most commonly encountered along the edges of prominent features or objects, where they generally form the boundary between the foreground and the background. This can be seen in in Figure XXX, which shows faces with silhouette edges along a shipwreck from a single camera viewpoint.

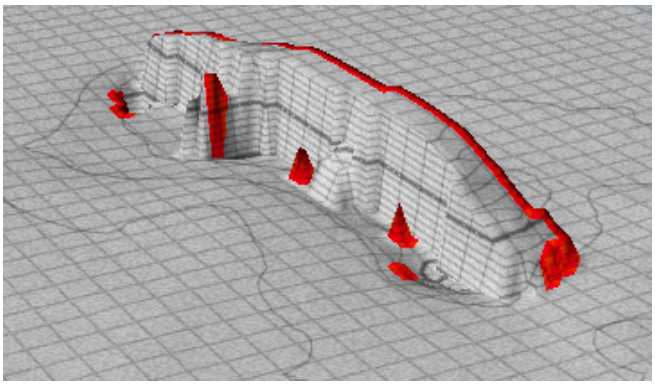

Fig. 3. In this view of a debris target mesh, the faces highlighted in red have silhouette edges for the current camera viewpoint.

Views that produce many prominent silhouette edges are likely to reveal the shape of target objects. Iteratively picking views that reveal different groups of silhouette edges helps to ensure all the shape information is being revealed across the multiple views. This effect can be seen in Figure XXX, which shows the faces with silhouette edges that have been revealed by picking the top four optimal views. Notice that the entire perimeter of the target object has been marked as revealed by the four views, indicating that the views provide a comprehensive depiction of the target's shape.

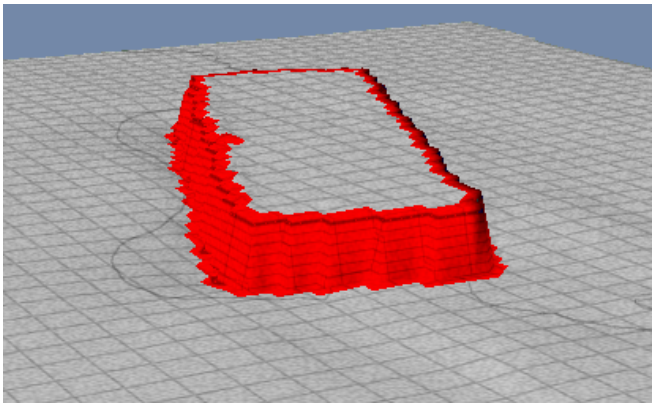

Fig. 4. In this view of a debris target mesh, the faces highlighted in red have silhouette edges that have been cumulatively revealed by the top four viewpoints.

Because silhouette edges are view-dependent, they must be calculated separately for each of the candidate views. There are multiple ways to determine which edges are silhouette edges:

It can be mathematically determined very quickly from the camera's position and the positions and normals of an edge's two vertices. First, calculate the pair of vectors from the camera to the two vertices. Then calculate the dot product of the first vector and the first vertex, and the dot product of the second vector the second vertex. If the signs of the two dot products are different, the edge is a silhouette edge for that camera position.

While this technique is very fast, it does suffer from a potential drawback, in that it does not consider occlusion from other faces. While uncommon with the small snippets of bathymetry, there are extreme cases, involving very steep terrain, in which some silhouette edges on debris will be occluded by the terrain from some viewpoints.

Occlusion issues can be overcome by calculating which triangles are actually visible. These types of visibility calculations can be done by rending the mesh to the framebuffer and then figuring out which triangles are not-occluded. There are multiple ways to do this, including checking z-buffer values or rendering each triangle with a unique color. Hughes et al. [17] provide an excellent overview of methods for determining visibility.

Once we find the silhouette edges for a candidate view, we calculate a silhouette score value for that view by weighting the silhouette edges by their total length. This avoids issues with varying resolutions etc.

All candidate views after the first to be chosen also receive negative penalty scores based on their similarity in viewing direction and inclination from any previously selected views. This helps produce a wider range of angles and directions in cases where complex objects produce clusters of silhouette edges.

For some situations, it may be advantageous to calculate a score to account for the faces that are revealed by a particular view. This requires calculating visibility/occlusion as mentioned previously. As views are selected, faces can be marked as having been shown. Candidate views that reveal faces not yet shown would receive a high reveal score, pushing them to the top of the list.

We did not use such a revealed face score in our work. Our data was solely gridded bathymetry, that when meshed, produces a "draped sheet" effect. When viewed from above (at or near 90˚) all faces are visible. Thus, this angle would always receive an overwhelming score for this test. Furthermore we had already manually set the fifth and final view to be a top-down 90˚ view if such an angle view had not already been selected.

The revealed face test could be modified to take into account the relative angle between the viewing direction and the face normal. In this case, even if a face could be seen in a candidate view, it would be marked as revealed unless it was seen "head-on". This would help ensure good coverage of vertical faces, such as those found on the sides of shipwreck-like objects.

Examples sets of optimal views for different marine debris targets can be seen in Figures 1 and 5.

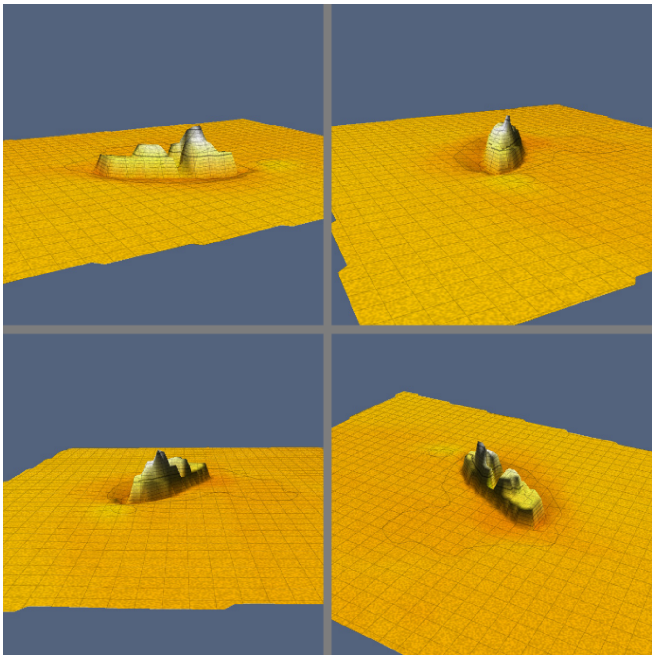

Fig. 5. The top four views chosen for a debris object mesh of a shipwreck using our optimal view algorithm.

C. Making Decisions

After targets have been loaded, and their optimal views selected, analysis can begin. The tool automatically displays the first target and waits for user input. Once the user has evaluated the target, they can input their decision using keyboard shortcuts or buttons at the top of the screen.

Analysts can mark targets to indicate whether they are natural features or marine debris, as well as the confidence level of the decision, including a no-confidence 'unknown' option. By having analysts include uncertainty measures in their decisions, we can re-display the sub-set of ambiguous targets to other analysts for a second round of evaluations.

When a decision is entered, the tool records the decision, confidence level, and target information to a report file covering that analysis session. The tool then switches to the next target, and the process repeats until all targets have been evaluated.

The tool also provides a method for automatically rendering and exporting all optimal views for all targets to form datasets for use outside the tool. In the next section we will explore using these exported images for web-based analysis.

D. Future Expansion

Gridded bathymetry is just one data product from the surveys of debris fields that producing targets that need evaluation. Other popular sources include the backscatter data from multibeam and side-scan sonar.

In addition to determining depth from bottom returns, multibeam systems can now record the backscatter, which is a measure of the returned acoustic energy after interaction with the seafloor. This can reveal details about bottom type, with hard rock producing stronger returns than softer sediments. Marine debris are likely to have different backscatter signatures than their surroundings, so this is an attractive method for locating them. Backscatter data could be incorporated into our tool by texturing the bathymetry mesh with the backscatter intensity values. This could either replace the height-based coloring, or modulate it.

Side-scan sonar, with or without backscatter data, is also popular for finding debris, especially in shallow waters. However since it produces 2D output, there is no advantage to multiple views from different angles. However, it is possible that multiple views with different rendering, color, or layer conditions might be helpful.

Our silhouette metrics only apply to meshed data. For point clouds or other non-mesh data types, finding optimal views will require different techniques. For example, Takahashi et al. [18] successfully adapted techniques for finding optimal views of 3D surface meshes for use in volume rendering. This is done by breaking down the volume into feature-based subvolumes and then evaluating sampled viewpoints based on the saliency of these feature subvolumes to find optimal views of the volume.

Finally, our silhouette scoring system could potentially be modified by incorporating view stability (as previously described in Section II). Instead of the current method of picking a single view with the most significant silhouette edges, it could instead consider clusters of nearby views which together produce similar (overlapping) sets of silhouette edges, then select a single view that best represents this cluster, and mark all the silhouette views from the entire cluster as having been revealed.

IV. Crowdsourcing Marine Debris Evaluation

After a disaster, minimizing response times is critical. While it is best to have trained analysts perform debris identification, their numbers and time are limited. We sought to experiment with and evaluate the concept of using crowdsourcing to conduct marine debris identification as method of increasing capacity during times of high demand.

We invited members of the general public to conduct these analyses via an interface on our website. While the users have no formal training, if large majorities make the same decision, they are likely to be correct. Even if their decisions are not 100% reliable, they can still be used to filter massive collections of targets into more manageable numbers for re-examination by trained analysts.

While the Marine Debris Rapid Decision Tool provides interactive viewports that can be adjusted, the website provides only static imagery. However, the images it provides are the same optimal views generated by the tool, as described in the previous section. Thus, it is likely that the multiple static views provided will be sufficient to make an informed decision. Figure 6 shows an example view of the web-based debris identification interface.

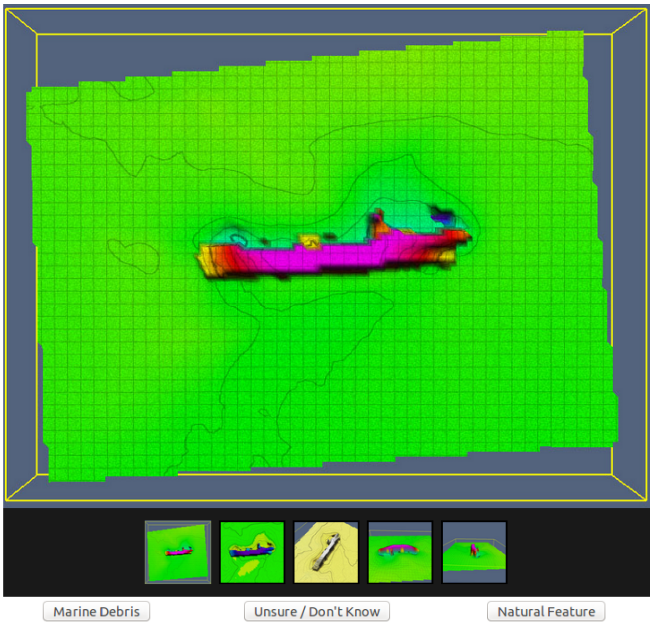

Fig. 6. A screenshot of the web-based crowdsourcing interface. Users see a single zoomed-in image, and can scroll through or click on the thumbnail images at the bottom to view each of them as many times as needed.

We advertised the project with links to our website through our research center's social media accounts and our university's main webpage. Users were given a brief overview of the project's goals and motivations. They were provided with only the bare minimum of basic directions and examples, enough to ensure they understood the task, but not detailed or in-depth enough to be considered training.

In addition to collecting users' decisions (i.e. debris, natural feature, or unsure), the system also records dwell times for each target. This information can be used for data-quality checks and helps identify difficult-to-interpret targets.

A total of 82 targets were available for evaluation via the online interface. To assist in data quality assessment, we included image sets for 12 targets whose categorizations were already known, and visually were obviously debris or featureless seafloor. These quality assessment targets were distributed amongst the unknown targets, with the distribution skewed towards the beginning of the target set to ensure they were evaluated even by people who chose to quit after only a few targets.

The quality assessment targets were of two types: four obvious debris targets of obvious shipwrecks, and eight snippets of barren seafloor devoid of any notable features. The design of the evaluation system was fully-crossed, meaning in this context that each participant evaluates the same targets as any other participant in the same fixed order.

V. Results

Over the course of approximately one week, we were able to attract approximately 800 members of the public to our crowdsourcing site, 35 of whom registered as participants. Of those 35, 34 users participated in the project by performing at least one evaluation. Additionally, an expert familiar with interpreting hydrographic and bathymetric data was recruited

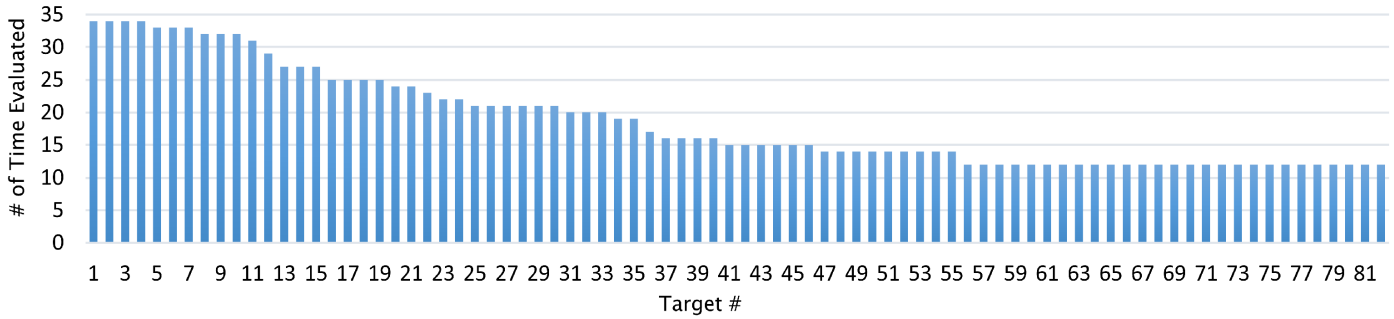

Fig. 7. A histogram displaying the frequency of target evaluation.

to evaluate the targets using the interface in order to provide a baseline by which to compare the evaluations of the participants.

Figure 7 shows a histogram of how many targets users chose to evaluate. A total of 12 users (35%) evaluated all 82 targets, and 15 users (44%) evaluated at least half of the targets. An average of 45 evaluations were made per user.

A. Quality Assessment Targets

The 12 quality assessment targets were compared with user evaluations to identify confused or possibly malicious users who may have provided lower-quality classifications through misunderstanding the task directions or intentionally attempting to confound the result.

While evaluating the quality assessment targets, a total of 25 users (74%) made no errors, while nine users (26%) made at least one error, as shown in Figure 8. As the maximum number of errors made across the 12 quality assessment targets was three, all users appeared to be making evaluations in good faith.

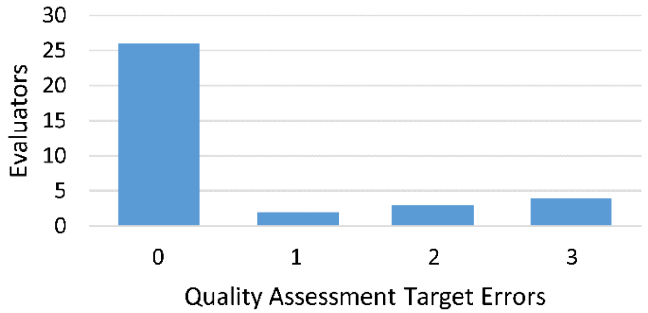

Fig. 8. Number of errors made by users judging quality assessment targets.

B. Decision Times

Page dwell times were used as a surrogate measure for decision time. The average time spent per user on each target was 11.4 seconds. Any dwell times above 120 seconds were considered to be from a distracted user and were discarded; as a result, 15 responses (less than 0.01% of the total responses) were not included in the calculation. There was a slight decrease in dwell time as participants progressed through the study, possibly indicating that they were learning what to look for or becoming more confident in their initial judgements. There was no correlation between page dwell times and users entering an "unsure / don't know" response. Ultimately, besides being useful to identify unreliable or malicious participants, dwell times did not reveal any particular insight.

C. Agreement and Inter-Rater Reliability

To assess the quality of the data collected from participants, we calculated raw percentage agreements between users as a rough metric of user quality as well as Krippendorff's alpha coefficient [19] to gauge the overall inter-rater reliability (IRR). Many options exist for analyzing IRR; we chose to use Krippendorff's alpha for its ability to handle nominal classifications made by any number of raters, while still being robust enough to accommodate missing data [20]. In these calculations, user responses of "unsure / don't know" were treated as missing data since that classification provides no extra information in the context of IRR; these responses can be regarded as a proxy for a "next" button to advance the targets without entering an evaluation.

As shown in Figure 9, of the 82 targets, 60 (73%) were agreed upon by a majority of the users who rated them; 48 (59%) were agreed upon by a supermajority; four (5%) were unanimously agreed upon; and 22 (27%) had no majority agreement amongst users. The inter-rater percentage agreements ranged from 53.1% to 93.3% with a weighted average agreement of 77.4%. Comparing users to our expert's evaluations yielded percentage agreements ranging from 57.1% to 100%, with a weighted average of 78.9%.

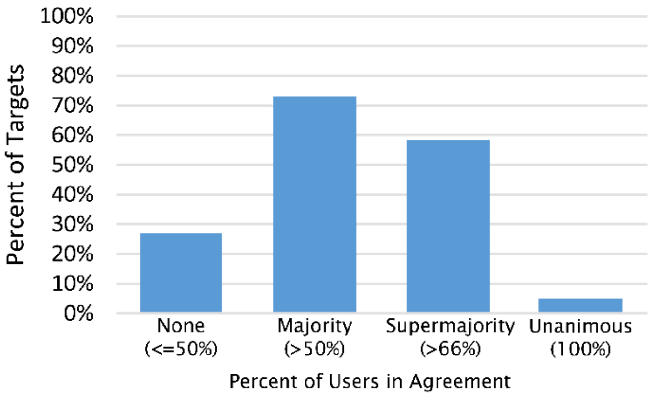

Fig. 9. Distribution of agreement between users across all targets.

Using our quality assessment targets, we split users into two groups to examine the effects of filtering out lower-

quality evaluations. Those users who made at least one error on a quality assessment target were gathered into an "error" group, and those making no errors formed the "non-error" group. Table 1 shows weighted average group percentage agreements between users and compared to our expert. It can be seen from the data that using these quality assessment targets to filter out lower-quality evaluators results in a significant increase in agreement both between users and against our expert.

TABLE I. WEIGHTED AVERAGE GROUP AGREEMENTS

Group	Agreement Between	
	Users	***Expert***
Error	69.8%	69.0%
Non-Error	81.3%	84.1%
All Users	77.4%	78.9%

Krippendorff's alpha coefficient was calculated for the entire group of evaluators and two subgroups consisting of users who made at most one quality assessment target error and users who made no quality assessment target error. To obtain the 95% confidence intervals for the alpha coefficients, 10,000 bootstrapping operations were performed on the data. The Krippendorff alpha coefficient for the entire group of evaluators was $\alpha = 0.473$, 95% CI of [0.378, 0.562]; for evaluators with at most one error $\alpha = 0.557$, 95% CI of [0.450, 0.655]; and for evaluators with no errors $\alpha = 0.573$, 85% CI of [0.457, 0.677]. These alpha coefficients and their confidence intervals are plotted in Figure 10. Given the imperfect nature of the data collected and the purpose of the project as a method to distribute a large workload amongst untrained volunteers, we regard these alpha coefficients as reasonable levels of inter-rater reliability.

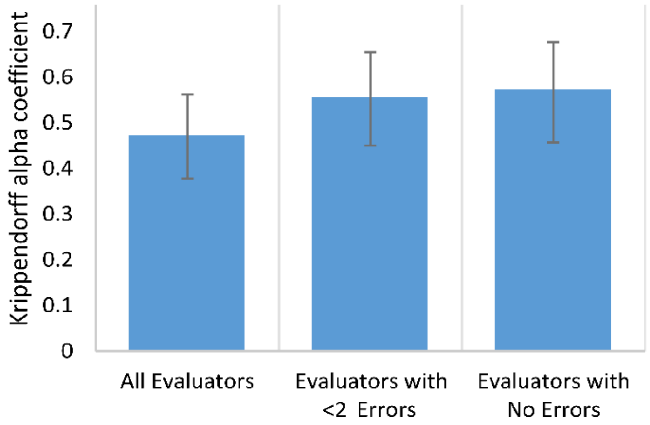

Fig. 10. Krippendorff alpha coefficient values with 95% confidence intervals obtained by 10,000 bootstrapping operations for three groups of evaluators.

VI. DISCUSSION

While our crowdsourcing system performed well in the context of this small dataset, there are a number of improvements that should be considered for larger-scale implementations in order to maintain a robust and attractive system for participants to use.

A. Barriers to Entry

One way to increase participation is to remove or simplify mechanisms that act as barriers to entry. One such barrier we noted was that we required users to sign up for an account by providing a username, password, and email address. Considering our low conversion rate (less than 5% of visitors created accounts), this was likely a significant deterrent for many who were either unwilling to take the time to fill out the registration form or did not want to provide that information.

To overcome this barrier, the easiest solution is to remove the need to register a new account with the system and bring a potential participant directly into the evaluation interface. One way that this can be achieved is by setting a persistent cookie or session in the user's web browser. This would allow a participant to leave and return to the site without losing the evaluations that have already been recorded, assuming the participant uses the same web browser and does not clear the cookie or session between visits.

Another option is to implement common login solutions, allowing visitors to use their existing accounts (e.g. Google, Facebook, or Windows Live) so that potential participants can quickly register with minimal effort. Recruiting participants through Amazon's Mechanical Turk would also provide users who are explicitly interested in participating in such tasks.

It is likely that the type of training a user receives affects their willingness to participate. In our system, we provided basic instructions and examples above the evaluation interface, enough that users could get a feel for the task and what is being looked for before registering and participating. If the potential participant does not feel as though the instructions are clear or that their evaluations will be effective, they are not likely to proceed. An engaging interactive training session that gives real-time feedback to the user is likely to boost the user's confidence in their ability to successfully complete the task, and thereby increase the likelihood of the user becoming a participant.

B. Incentivising and Motivating Participation

Once a user registers to participate, it is advantageous to keep the user engaged in evaluations for as long as possible. Feedback from some participants indicated that they would have liked to see a counter showing the number of evaluations made and the number of targets remaining to evaluate. This has the potential benefit of motivating users to finish evaluating the entire set of targets, but may have the side effect of lowering the evaluation quality of latter targets as the user reaches the "home stretch".

The order in which the targets are presented to the user could also influence the user's motivation. For example, if the user is shown too many low-resolution targets in a row, the user may become disinterested in the now-monotonous task and abandon it. To ensure that the user does not experience this, a measure of target complexity can be approximated by the total number and length of the silhouette edges revealed in the optimal views, as more and longer silhouette edges generally correspond to more complex (and therefore more interesting) targets. These targets can then be interspersed

amongst less-interesting targets in a manner that leads to increased user engagement.

There is also the possibility of "gamifying" the task by keeping score and offering virtual rewards to users. "Badges" may be awarded to users for successfully completing different tiers of evaluations, for sharing the study and recruiting other participants, or for any other beneficial metrics that keep the user entertained and involved. By allowing users to easily share their progress on social media platforms, this strategy has the secondary effect of increasing exposure and promoting the project to others in an organic way.

Obviously, one of the most effective ways to recruit and retain participants is monetary compensation. This strategy is readily available through systems like Mechanical Turk. When this type of compensation is chosen, it is important to implement safeguards against system abuse to keep users focused on sincerely completing the assigned task. One such method proposed by Welinder and Perona [21] dynamically assesses the quality of users' evaluations as they make them, excluding users who are unreliable from participating, leading to higher confidence in evaluations and reducing the cost of obtaining evaluations from Mechanical Turk workers.

VII. Conclusions

Finding and identifying marine debris is a tedious and time consuming task. Automatic target recognition algorithms can greatly speed up the task by searching vast survey datasets for anomalies that could be debris. However, they still leave analysts with thousands of potential marine debris targets that need to be individually examined.

We have presented a tool that automates many of the common and repetitive steps in the marine debris target evaluation workflow. Using metrics based on the computer graphics concept of silhouette edges, our Marine Debris Rapid Decision Tool automatically calculates multiple optimal views for each debris target. These optimal views are likely to reveal enough shape information for analysts to make decisions without the need for manually repositioning the camera themselves, increasing analysts' efficiency.

These optimal views were also utilized within an experimental crowdsourcing website that allowed the public to participate in marine debris target evaluation. We found that, even though participants were untrained and unexperienced in the task, most were able to understand and complete the task with reasonable accuracy. Their collective decisions can be used to greatly reduce the number of marine debris targets that must be evaluated by the limited pool of trained analysts. This can increase analytical capacity in time-critical disaster response situations.

References

[1] Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R., "A comparison and evaluation of multi-view stereo reconstruction algorithms", *Proc. IEEE Computer Vision and Pattern Recognition*, vol. 1, pp. 519–528, 2006.

[2] Mokhtarian, F., Abbasi, S., "Robust automatic selection of optimal views in multi-view free-form object recognition", *Pattern Recognition*, vol. 38, no. 7, pp. 1021–1031, 2005.

[3] Yamauchi, H., Saleem, W., Yoshizawa, S., Karni, Z., Belyaev, A., Seidel, H. P., "Towards stable and salient multi-view representation of 3D shapes", *Proc. IEEE Shape Modeling and Applications*, p. 40, 2006.

[4] Lee, J., Moghaddam, B., Pfister, H., Machiraju, R., "Finding optimal views for 3D face shape modeling", *Proc. IEEE Automatic Face and Gesture Recognition*, pp. 31-36, 2004.

[5] Vázquez, P.P., Feixas, M., Sbert, M., Heidrich, W., "Automatic view selection using viewpoint entropy and its application to image-based modelling", *Computer Graphics Forum*, vol. 22, no. 4, pp. 689–700, 2003.

[6] "Wikipedia Statistics", Wikimedia Foundation, Inc., http://stats.wikimedia.org/EN/TablesWikipediaZZ.htm, Aug. 2015

[7] Doan, A., Ramakrishnan, R., Halevy, A. Y., "Crowdsourcing systems on the world-wide web", *Communications of the ACM*, vol. 54, no. 4, pp. 86-96, 2011.

[8] Lintott, C. J., Schawinski, K., Slosar, A., Land, K., Bamford, S., Thomas, D., Raddick, M. J., Nichol, R. C., Szalay, A., Andreescu, D., Murray, P., Vandenberg, J., "Galaxy Zoo: morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey", *Monthly Notices of the Royal Astronomical Society*, vol. 389, no. 3, pp. 1179-1189, 2008.

[9] Heer, J., Bostock, M., "Crowdsourcing graphical perception: using Mechanical Turk to assess visualization design", *Proc. SIGCHI Conf. on Human Factors in Computing Systems*, pp. 203-212, 2010.

[10] Mason, W., Suri, S., "Conducting behavioral research on Amazon's Mechanical Turk.", *Behavior research methods*, vol. 44, no. 1, pp. 1-23, 2012.

[11] Liu, S. B., "Crisis crowdsourcing framework: designing strategic configurations of crowdsourcing for the emergency management domain", *Computer Supported Cooperative Work*, vol. 23, no. 4-6, pp. 389-443, 2014.

[12] Hester, V., Shaw, A., Biewald, L., "Scalable crisis relief: Crowdsourced SMS translation and categorization with Mission 4636", *Proc. ACM Computing for Development,* no. 15, pp. 1-7, 2010.

[13] Wald, D., Quitoriano, V., Worden, C. B., Hopper, M., Dewey, J. W., "USGS 'Did You Feel It?' Internet-based macroseismic intensity maps", *Annals of Geophysics*, vol. 54, no.6, 2011.

[14] Liu, S. B., Poore, B. S., Snell, R. J., Goodman, A., Plant, N. G., Stockdon, H. F., Morgan, K. L. M., Krohn, M. D., "USGS iCoast -- did the coast change?: designing a crisis crowdsourcing app to validate coastal change models", *Proc. ACM Conf. on Computer Supported Cooperative Work & Social Computing*, pp. 17-20, 2014.

[15] Calder, Brian, et al., "The open navigation surface project", *The International Hydrographic Review*, vol. 6, no. 2, 2005.

[16] Shewchuk, J. R., "Triangle: engineering a 2D quality mesh generator and Delaunay triangulator", *Applied Computational Geometry: Towards Geometric Engineering: Lecture Notes in Computer Science*, vol. 1148, pp. 203--222, 1996.

[17] Hughes, J. F., van Dam, A., Foley, J. D., Feiner, S. K., "Computer Graphics: Principles and Practice", Third Edition, Pearson Education, 2013.

[18] Takahashi, S., Fujishiro, I., Takeshima, Y., Nishita, T., "A feature-driven approach to locating optimal viewpoints for volume visualization", *Proc. IEEE Visualization*, pp. 495-502, 2005.

[19] Krippendorff, K, "Estimating the reliability, systematic error and random error of interval data", *Educational and Psychological Measurement*, vol. 30, no. 1, pp. 61–70, 1970.

[20] Hayes, A. F., Krippendorff, K., "Answering the call for a standard reliability measure for coding data", *Communication methods and measures*, vol. 1, no. 1, pp. 77–89, 2007.

[21] Welinder, P., Perona, P, "Online crowdsourcing: Rating annotators and obtaining cost-effective labels", *2010 IEEE Computer Vision and Pattern Recognition Workshops*, pp. 25–32, 2010.