

Consistency and Confidence: A Dual Metric for Verifying 3D Object Detections in Multiple LiDAR Scans

David L. Doria and Richard J. Radke*

Department of Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute, Troy, New York 12180

doriad@rpi.edu, rjradke@ecse.rpi.edu

Abstract

We introduce a dual, physically meaningful metric for verifying whether a 3D model occupies a hypothesized location in LiDAR scans of a real world scene. We propose two complementary measures: consistency and confidence. The consistency measure uses a free space model along each scanner ray to determine whether the observations are consistent with the hypothesized model location. The confidence measure collects information from the model vertices to determine how much of the model was visible. The metrics do not require training data and are more easily interpretable to a user than typical registration objective function values. We demonstrate the behavior of the dual measures in both synthetic and real world examples.

1. Introduction

Light Detection and Ranging (LiDAR) can be used to gather three-dimensional information about a scene that is impossible to obtain with standard optical imaging, such as the position of an object occluded by foliage. While devices for range scanning have recently become more widely available, substantial challenges remain for automatically extracting and understanding the information such scans contain.

Many problems including object detection, object recognition, and surface registration, share a common final step of aligning a 3D model with a LiDAR scan. The model is typically in the form of a triangulated mesh. In this paper, we propose a verification procedure using two complementary metrics that can be used on the outputs of any such alignments. We take as input a triangulated mesh representation of a 3D model, one or more LiDAR scans of a scene, and an estimated transformation of the model into the

scene. We wish to evaluate the hypothesis that the object is present at the given position. The verification procedure is independent of the method that produced the location hypothesis, so it is objective and unbiased in deciding if the position is indeed reasonable and correct. The advantage of the dual metric is that we can answer two independent questions simultaneously. We use a measure of *consistency* to determine if the object is in a position that makes sense physically. We use a measure of *confidence* to determine, if indeed the object is at a reasonable position, how much of it we have observed. The values produced by our consistency and confidence measures are both between 0 and 1, so they are easy to interpret for any data set. Together, the metrics enable a user to make a well-informed decision about the likelihood of an object's presence or the need for more scans of the scene to answer the question more conclusively.

The last step of many object detection methods is Iterative Closest Points (ICP) refinement [17]. For this reason, a variation of the ICP cost function value is generally used as the final "quality of match" value. To enable a model to be matched to a partial scan, the ICP cost function is typically modified to include only points whose nearest neighbor is within some threshold [13]. This drives the cost function to a very low value for a correct, partially overlapping match, but for an incorrect match, the value is still, by definition, fairly low. Furthermore, the ICP cost function value generally depends on the scale, sampling density, and parameterization of the problem, and is impossible to interpret as an absolute measure of match quality. We show that our metrics are much more discerning of the actual quality of a match.

The rest of the paper is organized as follows. In Section 2 we review related work on object detection and verification in range imagery. Sections 3 and 4 define the consistency and confidence measures, respectively. Section 5 describes the results of many experiments on both simulated and real data to demonstrate several aspects of the measures. Section 6 concludes the paper with discussion and ideas for future work.

*This work was supported in part by the DARPA Computer Science Study Group under the award HR0011-07-1-0016. Thanks to Barbara Cutler for the physical cat sculpture and its 3D model.

2. Related Work

Registering a model to a LiDAR scan is a common problem in object detection, object recognition, automatic target recognition, and 3D registration. Therefore, there is a significant body of work in which it is required to verify an object position as the last step of the algorithm.

Marino and Vasile [16] described a method to find military vehicles in LiDAR scans of outdoor scenes. Their final verification procedure, a “goodness of fit” test, used a weighted spin image [6] correlation coefficient. Huber et al. used a parts based approach [4] to classify objects with heavy self-occlusion into one of several predetermined classes. Chevalier et al. [2] located ground targets in large, outdoor scenes, first removing many scene points using a priori information (e.g., that the scene contains a large ground plane and many tall, thin trees). In each of these cases, our verification procedure could be used to provide an analyst with a method-independent, easily-interpretable physical check of the final detected object position.

Huber’s visibility consistency [5] is a method of determining the quality of alignment between two surfaces derived from range scans. A free space violation occurs if, after alignment, points in one of the scans occur in the free space of another scan’s perspective. This technique requires preprocessing to extract surfaces from the range images, and hand-labeled training data to estimate the probability distributions of the distances between two surfaces along each ray in the case of correct and incorrect alignments. Mian et al. [9] introduced the related concept of “active sensor space violation” as a means of determining the accuracy of a model-to-scene registration. This technique requires the scene and the model to have approximately equal sampling densities and is based on the number of model points that have a scene point within a specified distance threshold. They also used the difference between the volume occupied by the registered sets of points and the volume occupied by the model itself to determine a “bounding dimension” constraint that provides a coarse idea of whether the point sets are approximately correctly aligned.

Patterson et al. [11] proposed a two-step method for finding multiple similar objects in large data sets. First, possible positions are identified using spin images. These positions are then verified using Extended Gaussian Images [3]. The verification procedure requires hand-labeling parts of the input to provide exemplars of the objects of interest. Smith et al. [15] proposed a verification function based on a learned linear combination of several measures of registration accuracy, including variation in the normals of corresponding points, the stability of the covariance matrix of the estimated transformation, and a novel boundary alignment check. We emphasize that our method requires no training data.

3. The Consistency Measure

The first measure we propose is *consistency*, which is based on the violation of free space. That is, for a LiDAR ray to have reflected off of a scene point s , there must have been no objects along the line segment from the scanner origin to s .

We place the model in the scene at a hypothesized location. For each detected point in the scene, s , if the ray from the scanner through the point intersects the model, we have a “comparable pair” with which we can reason about free space. We compute this intersection efficiently by storing the model triangles in an octree and using standard ray-triangle intersection techniques [14]. The number of comparable pairs is denoted N_c . We know the direction of each scene point, s , from the scanner, and denote its distance from the scanner as d_s . The distance from the scanner to the model intersection, m , is denoted d_m . By considering the difference $d_m - d_s$, we can decide the consistency of the pair. If $d_m - d_s \geq 0$, the scene point is in front of the model point. This point could have been produced by either an occluding object or the object in the correct position, so we label it consistent. If $d_m - d_s < 0$, the scene point is behind the model point, which indicates that the LiDAR ray has passed through the object. This is a contradiction to the model being located at the hypothesized position, so we label the point inconsistent. To allow for noise in the acquisition process as well as slight error in the alignment, we introduce a mismatch allowance, a . We modify the conditions accordingly as given in (1) and Figure 1.

$$C_i = \begin{cases} 1 & (d_m + a) - d_s \geq 0 \\ 0 & (d_m + a) - d_s < 0 \end{cases} \quad (1)$$

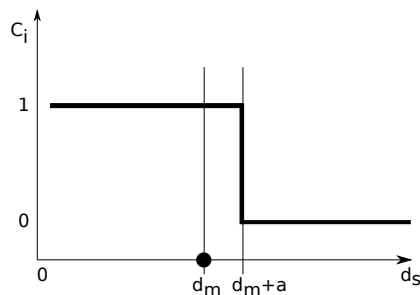


Figure 1: Diagram of consistency function

It is important to note the fundamental asymmetry in the consistency function. Model surfaces at equal distances in front of and behind a scene point would have very different consistency values, since the former is physically contradictory but the latter could have been produced by occlusion.

Figure 2 illustrates the idea with three examples of comparable pairs. In ray A, the scene point is significantly behind the model surface, so this point is inconsistent. In ray B, the scene point is only slightly behind the model surface, so this point is consistent. In ray C, the scene point is in front of the model, so this point is also consistent.

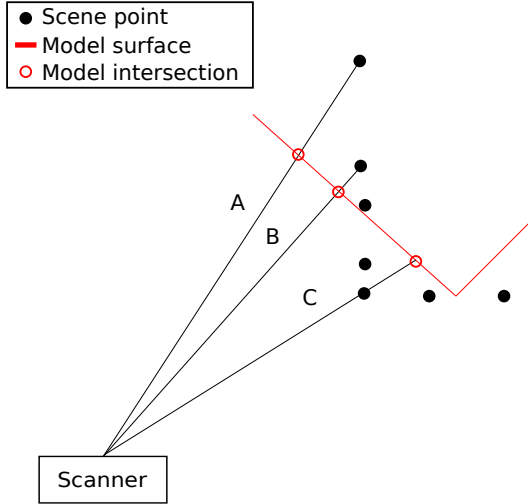


Figure 2: Consistency example for 3 rays.

We assign each comparable pair a binary value of 1 (consistent) or 0 (inconsistent) according to (1), and define the consistency of the model at the hypothesized location as the average consistency over all comparable pairs:

$$\text{Consistency} = \frac{1}{N_c} \sum_{i=1}^{N_c} C_i. \quad (2)$$

We note that this reduces the problem of verifying a 3D hypothesis to a combination of many 1D problems. We normalize by the number of comparable pairs to prevent the consistency value from being a function of the sampling density or the size of the object. A user can reasonably interpret this value between 0 and 1 without any other information.

If multiple registered scans of the scene are available, the consistencies of each scan should be combined into a total consistency score. It is assumed that the scene does not change between scans. Since the consistency of each scan is independent, the total consistency after observing K scans is

$$\text{Total Consistency} = \frac{\sum_{k=1}^K \sum_{i=1}^{N_c^k} C_i^k}{\sum_{k=1}^K N_c^k}. \quad (3)$$

4. The Confidence Measure

If a model position is completely consistent, we can only declare the model *could be* at the hypothesized location, not that it *is* at that location. For example, any object model is consistent with being entirely behind a scanned wall. Our second measure, *confidence*, indicates the reliability of an estimate based on what proportion of the model has been captured by the scan(s).

The confidence measure is based on the idea that a certain amount of *information*, I_i , is associated with every model point. This information should be related to how locally distinctive the point on the model is. For example, a point on the side panel of a car should have low information, since it looks similar to any planar surface, while the uniquely-shaped front bumper should carry more information. We require that the information from all the points in the model sums to 1.

Generally, a 3D model is constructed by an artist who uses a higher density of vertices to model more complex regions; this is the case for all the models in this paper. Thus, we can simply assign each point an equal amount of information, $I_i = \frac{1}{N_m}$, where N_m is the number of points in the model. If the model vertices are distributed uniformly (e.g., using an algorithm like [10]), the information content at each point could be related to the quality of a planar fit, with more locally complex regions containing more information. Linsen [7] proposed a more complicated method to determine the information content of a point based on point density, planarity, change in normals, and the uniformity of the change in normals.

Before any scans are acquired, we set the *observed information* O_i for each model point to 0. As scans are added, this value will increase to a potential maximum of I_i , the information content of the point. Each scanned scene point affects model points surrounding it at the hypothesis location. If a scene point is nearly coincident with a model point, it “uses up” that model point’s information- i.e., the model point has been completely “seen”. We define the incremental update rule for the influence of the j th scene point on the i th model point using a Gaussian function:

$$O_i \leftarrow \min \left(I_i, O_i + I_i e^{-\frac{d_{ij}^2}{2\sigma^2}} \right) \quad (4)$$

Here, d_{ij} is the distance between the two points. σ determines the radius of the sphere inside which model points are affected. One could reasonably choose σ to be a function of either the model bounding box volume m_v or the median model vertex spacing. For the experiments in this paper, we set $\sigma = 0.01m_v$.

Since the Gaussian function is negligibly small for $|d_{ij}| > 3\sigma$, we find all points within 3σ of the scene point using a KD-tree [1] and compute the update for only those

points. Figure 3 illustrates an example of the information observation process, showing the influence of one scene point on three model points.

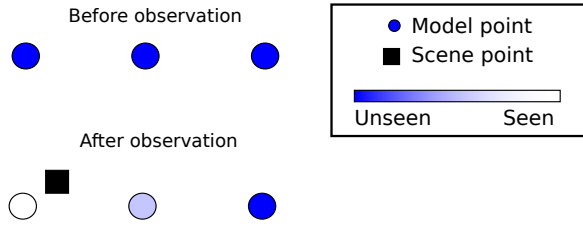


Figure 3: Confidence Example

The confidence that a model exists at a given location after all of the information has been collected is

$$\text{Confidence} = \sum_{i=1}^{N_m} O_i \quad (5)$$

where N_m is the number of model points.

We note that unlike the consistency measure, the confidence measures are not independent from scan to scan, because any overlap in scans will “see” some of the same model points. Therefore, the computation of the confidence over K multiple scans is computed as if all scene points came from a single scan. The confidence equation does not change; the only difference is that the observed information is iteratively accrued from all the points in all K scans.

5. Experimental Results

In this section, we report the results of several experiments that demonstrate the properties of our dual metric. All real LiDAR scans were acquired with a Leica HDS 3000 scanner with sample spacing approximately 3 mm on the object surface.

5.1. Cat Sculpture – Varying Position

We obtained a high precision triangulated model of a real cat sculpture using a hand-held scanner. The dimensions of the bounding box of the model are 30x25x12cm. We then LiDAR-scanned the physical sculpture in an unoccluded scene. We used spin images followed by ICP to automatically estimate the position of the cat sculpture in the unoccluded scan (Figure 4a). We computed the baseline confidence and consistency values for this real world registration. The confidence value is 0.544 because we only acquired one scan covering about half the model. The consistency value is 0.792, due to slight misalignment in the registration process as well as scanner noise. Throughout the experiments

with the cat sculpture, we use a mismatch allowance of 2cm for the consistency calculations, and $\sigma = 0.5\text{cm}$ for the confidence calculations.

We then placed the model behind the correct position, i.e., in the “shadow” of the LiDAR scan (Figure 4b). Table 1 shows that the consistency value in this position is very high, since almost all of the scan points do not contradict the hypothesized location. We then placed the model in front of the correct position (Figure 4c). The consistency is extremely low in this position, since the model is in front of the observed scene points, clearly a contradiction to the hypothesis. In both the in front and behind positions, the confidence measure is extremely low because there are almost no scene points near the model.

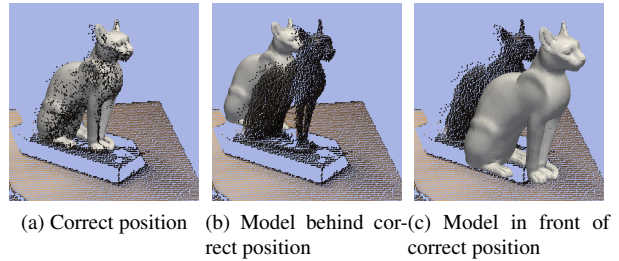


Figure 4: Cat sculpture in varying positions.

Position	Confidence	Consistency
Aligned correctly (a)	0.544	0.792
Model behind scene (b)	0.003	0.995
Model in front of scene (c)	0.000	0.151

Table 1: Consistency and confidence values for varying model positions in Figure 4.

5.2. Cat Sculpture – Varying Occlusion

To demonstrate the effect of occlusion on our metrics, we scanned the cat sculpture behind several different types of material. We used spin images and ICP to register the model of the cat sculpture in the scan with no occlusion, and used this position to compute the confidence and consistency metrics in five situations.

The first row of Figure 5 shows digital images of the cat sculpture under the varying occlusion conditions. The second row shows the LiDAR scans of the occluding object as well as the cat sculpture to illustrate the scan points that fell on the sculpture. Table 2 summarizes the consistency and confidence measures for the five cases. Figure 5a shows the scene with no occlusion and is provided as a baseline reference. The consistency is very high, and the confidence is 0.476, a typical value after observing the object from only one viewpoint. In Figure 5b, we scanned the scene through

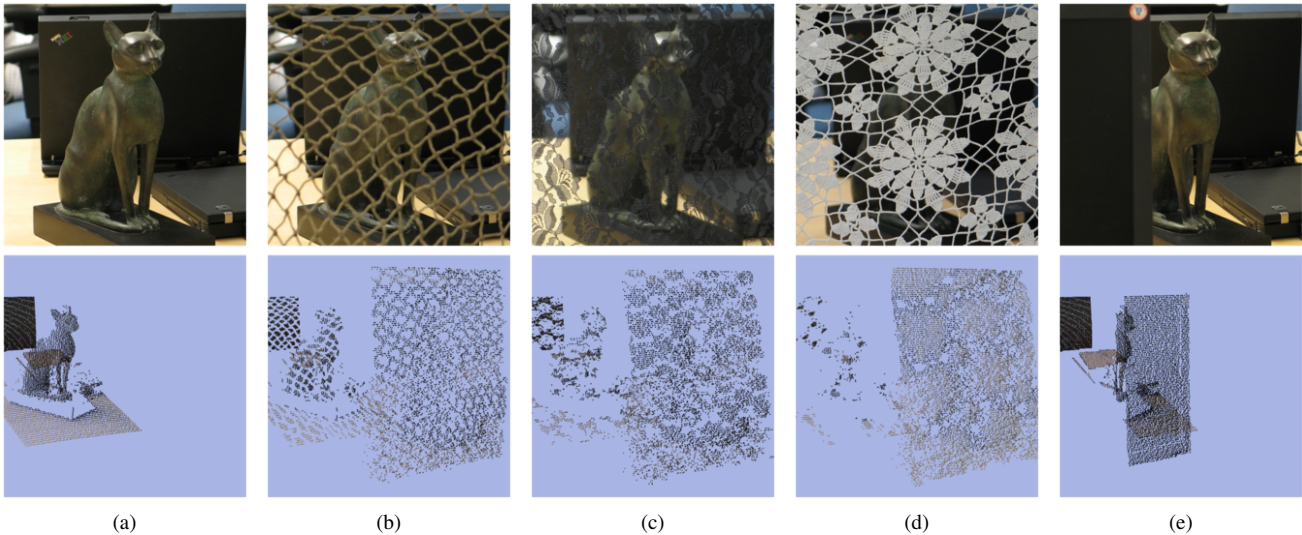


Figure 5: Cat sculpture scans with varying occlusions. (a) No occlusion, (b) Light, sporadic occlusion (net), (c) Heavy, sporadic occlusion (lace), (d) Heavy, sporadic occlusion (tablecloth), (e) Heavy, contiguous occlusion (monitor)

a net to imitate a scaled down camouflage net. The confidence of the model decreases by about half, which agrees with our intuition that we only see about half as many points on the sculpture as we did in the unoccluded scan. However, the consistency is still very high. In Figure 5c, we scanned the scene through a piece of lace fabric to imitate extremely dense foliage. Again, the consistency value is still very high, but the confidence has decreased even further, as even fewer points on the sculpture have now been seen. In Figure 5d, we occluded the cat sculpture with a tablecloth. The results are similar to the lace fabric. Finally, in Figure 5e, we occluded the cat with a monitor. The consistency value is still high, but the confidence value is similar to that of the “net” case of Figure 5b.

Occlusion	Confidence	Consistency
None	0.476	0.879
Light, sporadic (net)	0.257	0.952
Heavy, sporadic (lace)	0.195	0.958
Heavy, sporadic (tablecloth)	0.083	0.985
Contiguous (monitor)	0.256	0.963

Table 2: Experimental values of consistency/confidence for different types of occlusion, cat sculpture.

5.3. Synthetic Cars – Multiple LiDAR Scans

In the next experiment, we demonstrate how additional LiDAR scans of a scene help improve our knowledge, as

well as how the consistency and confidence metrics can be used to disambiguate similar objects. We considered a database of five synthetic automobile models, each with its center of mass at the origin. The models are all at life-size scale. We simulated sequentially LiDAR scanning each car from four different perspectives (front, driver side, rear, passenger side). The synthetic scans were created using custom software that we wrote to simulate the output from the Leica scanner that we use for real-world scans. The input is a scene consisting of triangulated meshes, a forward direction, spherical angle bounds, and spherical angle spacing. The output is a point cloud of the visible surfaces in the scene.

In Figure 6, the i^{th} row represents that we are hypothesizing the i^{th} model exists. The j^{th} column represents that we are comparing a hypothesis to synthetic LiDAR scans of the j^{th} model. For example, in cell $i = 2, j = 4$, we are hypothesizing the existence of the sedan2 model and comparing it to LiDAR scans of the SUV.

Each square cell in Figure 6 contains an independent coordinate system with confidence on the horizontal axis and consistency on the vertical axis. The k^{th} point from the left in each square represents the value of the confidence/consistency after seeing the first k scans.

Throughout this paper, for experiments with automobiles we use a mismatch allowance of 10cm for the consistency calculations, and $\sigma = 0.3\text{cm}$ for the confidence calculations.

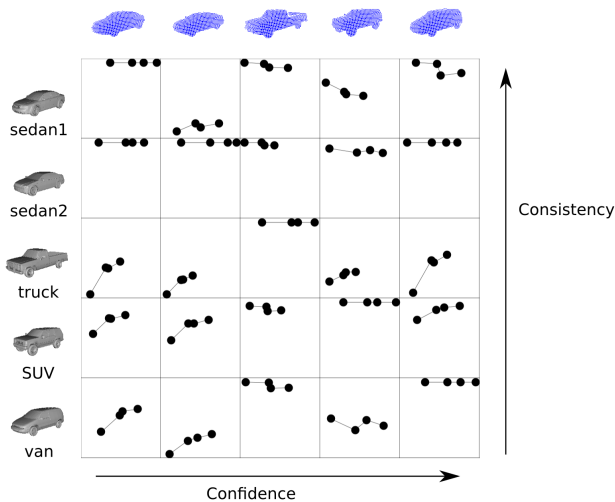


Figure 6: Confidence/Consistency evaluation between all combinations of five automobile models. Rows: models, columns: LiDAR scans. Each dot (left to right) represents an additional scan taken from the front, driver side, rear, and passenger side viewpoints, respectively.

Some noteworthy observations are:

- The consistency is always 1 for squares on the main diagonal. This indicates that each model’s consistency with itself is 1.
- The confidence increases or remains constant with each additional scan.
- Since it is smaller, the sedan2 model is consistent with the scan of sedan1 (cell (2,1)), but the sedan1 model is not consistent with the scan of sedan2 (cell (1,2)).
- Three of the models are smaller than the van. Therefore, they are each consistent with the scans of the van (cells (1,5), (2,5) and (4,5)). However, the truck is longer than the van, so the truck model is inconsistent with the scan of the van (cell (3,5)).
- In cell (3,1), we can see that the front of the truck is inconsistent with sedan1, but the sharp increase with the second scan indicates that their sides are similar.

5.4. Real Parking Lot Scans

Typical coarse registration algorithms produce several initializations that are refined by an ICP method. Some of these initializations produce high average point-to-point distances and can quickly be discarded. However, several positions often need to be manually discarded by the user. Such positions have a low average distance, but are physically very incorrect. Since a typical ICP cost function value

depends on the scale, sampling density, and parameterization of the problem, it is very difficult to compare the quality of matches across multiple search objects and scales. Our metrics, however, are independent of object size and therefore can easily be directly compared. In this example, we demonstrate how our metrics are much easier to interpret than the ICP cost function values.

Position	ICP Cost Function	Confidence	Consistency
Correct	0.057	0.579	0.589
Incorrect	0.094	0.252	0.077

Table 3: Measures for Audi positions in parking lot scan.

We acquired a LiDAR scan of two cars in a parking lot. Two hypothetical outputs of a coarse registration algorithm between an Audi A4 model and the scene are shown in Figures 7a and 7b. One is correct, and the other is incorrect (it lies halfway between the two cars in the scan). Table 3 reports the ICP cost function value, consistency, and confidence for the two positions. We employed a standard ICP cost function, shown in (6).

$$\text{ICP Cost Function} = \frac{1}{N} \sum_{i=1}^N \|\mathbf{R}\vec{x}_i + \mathbf{t} - \vec{y}_i\| \quad (6)$$

Here, x_i is a scene point and y_i is the nearest model point to x_i . Scene points for which the nearest model point is more than 0.2 meters away were not included in the ICP cost function, a common technique described in [13]. It is important to note that regardless of which variant of the ICP cost function is used the value is always in meters, in contrast to our metrics which both take unitless values between 0 and 1. Also, as the complexity of the selected ICP function increases (e.g., by weighting each point’s contribution differently), the ability to intuitively interpret the value decreases.

Both positions have comparable average point-to-point distances (the ICP cost function value), which are both below 10 cm. For our new measures, the correct position has a high confidence value (given only one viewpoint) as expected, and the consistency is reasonable, though slightly lower than ideal. This is largely due to the transparent windshield in the real scene, which causes discrepancies in model fitting (see [8]). However, in the other position, the extremely low consistency value alone is grounds to declare this position incorrect. The confidence is non-zero because the each side of the model aligns with the adjacent side of one vehicle in the scene.

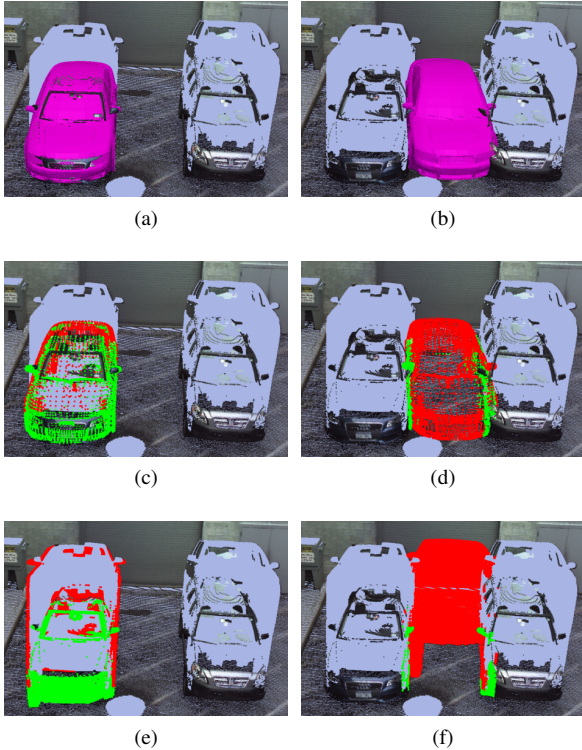


Figure 7: Parking lot demonstration. (a) Model registered to correct position in scene. (b) Model registered to incorrect position in scene. (c) Model points at correct position colored by confidence (unseen: red, seen: green). (d) Model points at incorrect position colored by confidence (unseen: red, seen: green). (e) Scene points at correct position colored by consistency (inconsistent: red, consistent: green) (e) Scene points at incorrect position colored by consistency (inconsistent: red, consistent: green). This figure is best viewed in color.

In Figures 7c and 7d, we show the observed information of the car model vertices in both positions. In the correct position, the front and driver side points are green (seen) and the rest are red (unseen). In the incorrect position (between the two cars), points on both sides of the model are seen, but the rest of the points are unseen.

In Figures 7e and 7f, we see that in the correct position most of the points are consistent. The inconsistencies stem from the model not being a perfect match (i.e., the model is a 2000 Audi A4 and the scene is a 2009 Audi A4) as well as slight misalignment. In the incorrect position, almost all of the points are inconsistent because the scanner “saw through” the model to the back wall. This is a typical example of how the consistency and confidence measures play a useful dual role for understanding if a hypothesized position makes physical sense.

Figure 8 illustrates a second LiDAR scan of three auto-

mobiles in a parking lot. We computed the consistency and confidence measures for an Audi A4 car model positioned at every 20 cm in the horizontal and vertical directions, assuming the model is major-axis-aligned with the parking space lines and located on the ground plane.

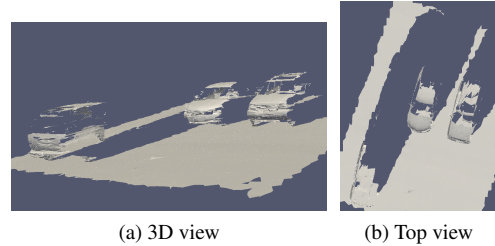


Figure 8: Parking lot scene with three cars.

Figure 9a shows a “heat map” of consistency values over the scene. We see that positions in the LiDAR shadow of the automobiles have high consistency. Figure 9b shows a heat map of the confidence values over the scene. There are several false positives. These can occur when significant parts of the model align with the scene, due to symmetries and the fact that any two near-planar objects tend to look alike. In Figure 9c, we thresholded the consistency map with a value of 0.75 and the confidence map with a value of 0.3 and boolean ANDed the resulting images. The position of all three automobiles are clearly verified with no false positives. However, we believe that considering both measures together leads to better-informed decisions than combining them into a single scalar value.

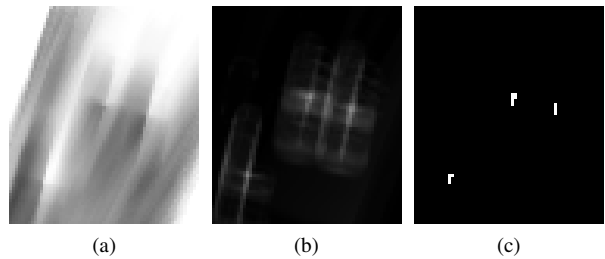


Figure 9: Consistency/confidence heat maps. (a) Consistency heat map (b) Confidence heat map (c) Dual thresholded with consistency > 0.75 and confidence > 0.3 .

6. Discussion and Conclusions

We presented a dual metric for deciding whether a 3D object exists at a hypothesized location in a LiDAR scan. A set of such locations produced by any registration method can be verified using these measures, which together are able to provide physically meaningful values for a user to

interpret. The experiments demonstrated the feasibility and accuracy of this method.

The consistency calculation is currently somewhat slow e.g., an average of 8 seconds for each position in Figure 8 on a Pentium 4, 3GHz computer with 2GB of RAM. This is primarily due to using a very high resolution model ($\approx 500,000$ triangles). Standard mesh decimation techniques from computer graphics tend to fail to maintain the overall structure of the mesh when it is not topologically equivalent to a sphere, which is typical of models that were not designed with resolution variability in mind. For this reason we chose to use the high resolution mesh throughout the experiments. We also plan to speed up the consistency calculation by employing a coarse-to-fine strategy. For example, we could evaluate the consistency using a uniformly downsampled set of the scene points. If the downsampled scene points are inconsistent with the model hypothesis, the probability that the entire set is also inconsistent is extremely high and further computation can be avoided. We could also use a depth buffer comparison rather than a ray-wise comparison to tremendously speed up this computation. However, there are several difficulties with this approach. The scan is a point cloud, not a triangulated mesh, so a point rendering system such as [12] must be employed. The resolution of the rendering window must be chosen such that there are similar numbers of corresponding pixels as there are scene points (and thus rays).

Currently, our dual metrics return similar values for a given amount of occlusion without considering the contiguity of the occlusion. For example, in Figure 5, the “net” occlusion produces almost identical values to the “monitor” occlusion. A possible solution is to remove the independence assumption on the collection of 1D problems along each ray, e.g., by using a first order Markov random field. This approach would favor neighborhoods of scene points that had similar consistencies.

Our consistency calculations assume that multiple registered scans come from a perfectly static scene. Relaxing this assumption would open up new research questions. For example, we could determine that an object was present and still for one scan, and then was either moved or occluded before the next scan was acquired.

Finally, we note that an accurate 3D model is frequently not available for the objects we might want to locate in the scene. This calls for a non-model based approach, in which the consistency and confidence for a scan are determined with respect to several example pictures or scans of a model.

References

- [1] J. Bentley. Multidimensional Binary Search Trees Used for Associative Searching. *Communications of the ACM*, 18(9):509–517, 1975.
- [2] T. Chevalier, P. Anderson, C. Gronwall, and G. Tolt. Methods for ground target detection and recognition in 3D laser data. Technical report, Swedish Defense Research Agency, 2006.
- [3] B. Horn. Extended Gaussian images. *Proceedings of the IEEE*, 72(12):1671–1686, 1984.
- [4] D. Huber, A. Kapuria, R. Donamukkala, and M. Hebert. Parts-based 3D object classification. In *IEEE Computer Society Conference On Computer Vision and Pattern Recognition*, volume 2, 2004.
- [5] D. F. Huber and M. Herbert. Fully automatic registration of multiple 3D data sets. *Image and Vision Computing*, 21(7):637–650, 2003.
- [6] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433–449, 1999.
- [7] L. Linsen. Point cloud representation. Technical Report 2001-3, Fakultt für Informatik, Universität Karlsruhe (TH), 2001.
- [8] B. Matei, Y. Shan, H. Sawhney, Y. Tan, R. Kumar, D. Huber, and M. Hebert. Rapid object indexing using locality sensitive hashing and joint 3D-signature space estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1111–1126, 2006.
- [9] A. Mian, M. Bennamoun, and R. Owens. Three-Dimensional Model-Based Object Recognition and Segmentation in Cluttered Scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1584–1601, 2006.
- [10] D. Nehab and P. Shilane. Stratified point sampling of 3D models. In *Eurographics Symposium on Point-Based Graphics*, 2004.
- [11] A. Patterson, P. Mordoi, and K. Daniilidis. Object detection from large-scale 3D datasets using bottom-up and top-down descriptors. In *European Conference on Computer Vision*, volume 4, pages 553–566, 2008.
- [12] S. Rusinkiewicz and M. Levoy. QSPat: A multiresolution point rendering system for large meshes. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pages 343–352, 2000.
- [13] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. In *Proceedings of the Third International Conference on 3D Digital Imaging and Modeling*, pages 145–152, 2001.
- [14] H. Samet. Implementing ray tracing with octrees and neighbor finding. *Computers And Graphics*, 13(4):445–460, 1989.
- [15] E. R. Smith, B. J. King, C. V. Stewart, and R. J. Radke. Registration of combined range-intensity scans: Initialization through verification. *Computer Vision and Image Understanding*, 110(2):226–244, 2008.
- [16] A. N. Vasile and R. M. Marion. Pose-independent automatic target detection and recognition using 3D laser radar imagery. *Lincoln Laboratory Journal*, 15(1), 2005.
- [17] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.