Obstacles and Foliage Discrimination using Lidar

Daniel D. Morris

Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA

ABSTRACT

A central challenge to autonomous off-road navigation is discriminating between obstacles that are safe to drive over and those that pose a hazard to navigation and so must be circumnavigated. Foliage, which can often be safely driven over, presents two important perception problems. First, foliage can appear as a large impenetrable obstacle, and so must be discriminated from other objects. Second, real obstacles are much harder to detect when adjacent to or occluded by foliage and many detection methods fail to detect them due to additional clutter and partial occlusions from foliage.

This paper addresses both the discrimination of foliage, and the detection of obstacles in and near foliage using Lidar. Our approach uses neighboring pixels in a depth image to construct features at each pixel that provide local surface properties. A generative model for obstacles is used to accumulate probabilistic evidence for obstacles and foliage in the vicinity of a moving platform. Detection of obstacles is then based on evidence within overlapping cells of a map without the need to segment segment obstacles and foliage. High accuracy obstacle and foliage discrimination is obtained and compared with the use of a point scatter measure.

Keywords: Foliage, Obstacles, Trees, Discrimination, Hough Transform, Lidar, Mobile Robots

1. INTRODUCTION

Lidar has proven to be a versatile sensor for robust environment analysis for mobile robots. Methods have been built for terrain classification, ground surface estimation, object segmentation, classification and tracking. All of these sensing capabilities are needed to enable autonomous operation of mobile robots in both urban and natural environments.

Much work focuses on urban terrain and on-road driving. These environments are characterized by mostly flat terrain, large buildings with flat surfaces, and scattered objects including vehicles and pedestrians. Since objects lie on the ground plane and tend to be spaced apart, segmenting Lidar returns into contiguous objects can be achieved and then followed up with classification.

This paper addressed the problem of off-road, Lidar-based sensing for mobility. These environments provide quite different challenges to urban terrain. Obstacles can be small or extended, and may directly abut other objects, making the segmentation difficult. Foliage provides a particular challenge with features typically much finer than the Lidar sampling. It can be both clumped or extended such as a field of tall grass, and in these cases there may be no meaningful shape to be obtained by clustering points. In addition foliage may abut objects or surround them such as tree trunks. In order for robots effectively and safely move through off-road environments that have dense foliage, it is essential that they are able to discriminate non-traversable obstacles from foliage. This paper provides a method for performing this discrimination.

A commonly used cue for foliage discrimination is the local point scatter expressed using the eigenvalues of point covariance matrices. To be effective this requires a high surface sampling density on obstacles so that they can be modeled as locally planar. As a result methods using this typically accumulate accumulate Lidar points over time from a moving platform followed by discrimination. While this enables detailed scene analysis, it comes with the drawback, especially at longer ranges, that it can take many scans to acquire sufficient sampling density, significantly constraining the speed of the robot, or else requiring large, expensive and high-speed Lidars.

Further author information: (Send correspondence to D. Morris)
E-mail: dmorris@msu.edu, Telephone: +1-517-432-4427
In this paper we seek new features for foliage discrimination that are effective at lower resolutions and so enable discrimination at longer ranges than scatter-features. The key geometric knowledge that will enable our new features are the known viewing rays between the sensor and points, as well as pixel adjacency in the range image. With this richer pixel information we seek to robustly discriminate obstacles and foliage based on local evidence without the need for object segmentation.

2. RELATED WORK

Terrain classification work such as Lacaze et al. builds a local map around a robot with a height-based traversability measures. This can enable path planning and robot mobility that avoids obstacles and unsafe slopes. Its limitation is that apart from height and extent, obstacle types are not discriminated and both a tall bush and a tree trunk are equally to be avoided. Permeability of voxels or columns to Lidar rays has been used to discriminate load-bearing surfaces from ground-surface foliage. This works best for sparse foliage where some Lidar beams penetrate to the ground. While foliage can be represented with voxels, this is limited in its ability to distinguish it from obstacles.

A statistical decay function is used by Manduchi et al. to model Lidar penetration into tall grass. Obstacles in tall grass can be detected by changes in the covariance. This is limited short ranges in tall grass and requires many pixels on target.

In work by Vandapel et al. and Lalonde et al., Lidar points are accumulated and their local distribution characterized by the eigenvalues of a point covariance matrix. Linear structures have one large and two small eigenvalues, planar structures have two large and one small eigenvalue, and scattered points have three large eigenvalues. Obstacle surfaces such as tree trunks or branches are modeled as locally planar or linear surfaces, while foliage is modeled as scatter. This distinction can be captured by the ratio of the smallest to largest eigenvalue.

Building on local feature classifiers, nearby context has been used to improve 3D point classification. Anguelov et al. applied an Associative Markov Network to point clouds features, and demonstrated segmentation of objects into ground, building, trees and shrubbery. Good results are reported for dense Lidar point clouds. Munoz et al. also use a Markov Network and include higher-order interactions between features to enable features such as power-lines to be estimated. Lim and Suter partition dense point clouds into clusters and use a Conditional Random Field to leverage multiple-scales of context in classification. While context improves overall results, its downside is that objects close to foliage can be grouped with the foliage and misclassified, such as vehicle under trees, or trees next to buildings. Contextual classification has also shown its utility in aerial-based Lidar scene analysis. While contextual information certainly has value, it is not the focus of this research, and these contextual methods can be applied to the features proposed in this paper.

Point clustering and segmentation has been used as a first step in some scene analysis methods. Buildings and other structures separated on the ground plane can be segmented. Tree stems are modeled as cylinders and cones and fitted to clusters of points. However, methods that depend on clustering can be fragile as objects can be merged when clutter is close to targets as is often the case in natural terrain.

There methods using other modalities for foliage detection such as color plus NIR, although these have their limitations particularly for dried out foliage which lacks Chlorophyll. A real system is likely to combine multiple modalities and algorithms. What is important is to maximize the information content of each modality. In this regard we believe there is more useful information available in Lidar depth images that is not captured by the local eigenvalue-based features. Thus this paper focuses on extracting new robust features from depth images, modeling their statistics and leveraging them to discriminate obstacles from foliage.

3. LIDAR SENSOR AND DATA

For this project a Velodyne 32 Lidar was mounted on an Husky robot at 1.6m height. 32 return laser beams scan 360 degrees in azimuth at 10 Hz. The azimuth angular resolution at 0.17° is 8 times the elevation resolution of 1.34°. For fast object detection and discrimination the method here operates on each frame at ranges up to 30m without the need to accumulate data.
4. DEPTH IMAGE SURFACES

This paper seeks new features for improved obstacle and foliage discrimination. Current methods\textsuperscript{7,9,10} still rely on point scatter measures, such as the ratio of the minimum to maximum eigenvalue of 3D pixel covariance matrices.\textsuperscript{1,2} These statistics are simple to calculate and combine, but are limited in their modeling of the world including that they do not provide an explicit surface model of objects in the environment. In contrast, the approach here builds features based on explicit surface models and normals.

While local surfaces can be estimated from raw 3D point clouds using mesh methods,\textsuperscript{14–17} these typically require high surface sampling compared to the distance between objects. In natural terrain scenes with complex objects, such as trees, the Lidar sampling is often sparser than the structure of the objects. Compensating by increasing the resolution through multiple scans or larger and more expensive Lidar sensors have negatively impacts robot performance.

The alternative to starting from raw 3D point clouds, is to obtain surface estimates directly from the Lidar range-images by leveraging pixel adjacency. Key challenges that must be addressed include how to obtain normals that are robust to sensor noise and target roughness, and to avoid smoothing across object depth discontinuities. The following section describes how these surface features are used in modeling obstacles and foliage.

4.1 Lidar Pixel Edge Features

A Lidar Pixel Edge (LPE) feature is defined by up to four edges between the pixel and its four neighbors. The edges represent the object surface and should only be included if they approximate the underlying surface. The simplest construction is a direct connection from a pixel to its four neighbors in the range-image: above and below, to the left and to the right, as illustrated in Fig. 1(a). This, however, has two problems. At close range where sampling is dense, noise in range as well as surface roughness results in noisy, non-representative edges as illustrated in Fig. 3(a) and (c). Noisy surface normals add to the difficulty in characterizing obstacle shape and discriminating obstacles from foliage. In addition at boundaries of objects these edges join objects rather than approximating an object surface. Thus the first step of our method is to determine a better set of neighbors for each pixel, such that the edges between each pixel and its neighbor approximate the underlying surface at the desired scale. The result is a LPE, as illustrated in Fig. 1(b).

We improve pixel feature edges by a combination of two processes: removing edges that connect separate objects and increasing the edge length within objects by connecting them to more distant pixels. This edge expansion seeks edges that stay within a threshold distance of the surface that is based on surface roughness and range noise. In this way edges capture the underlying object surface normals by reducing the effects of noise and surface roughness, and at the same time maintaining sharp boundaries between objects.

4.2 Pairwise Connections

Before extending edges, an initial connection is determined between all pairs of adjacent pixels in the depth image. The goal is to connect pixels on the same objects (or obstacles) while not connecting pairs of points on objects that are spatially separated. A full segmentation of obstacles is not needed as local features will be used. The two primary cues used in determining connectivity are spatial proximity and surface smoothness, as described next.

First the easy extreme connectivity cases are considered. Adjacent pixels whose range difference is below a small threshold, $t_{r1}$, are assumed to be on the same surface and so are connected. Also if the range difference between adjacent pixels is greater than a large threshold, $t_{r2}$, they are not connected. Intermediate cases are more difficult and correspond to edge points on obstacles close to other obstacles or foliage, as well as points on flat surfaces viewed at a grazing angle. In these cases a threshold $t_{r3}$ is used on the range difference between a pixel and the mid-point between its left and right neighbors for horizontal connectivity, and below and above neighbors for vertical connectivity. This connects widely spaced points on flat surfaces (when viewed at steep grazing angles), while breaking connectivity at edge discontinuities. The result of these cues is a graph indicating the connectivity between each pixel and its four adjacent neighbors.
4.3 Extending Connections

Once adjacent connections are established, edges from each pixel can be extended to more distant pixels. The goal is that edges will capture obstacle shape rather than surface roughness or sensor range noise. Edge extension is directional in the range image and uses the same method in four directions, namely to the left, right, above and below each pixel. This is illustrated in Fig. 2 and described in this section.

A measure of maximum surface roughness plus sensor noise is captured in a threshold $t_{r4}$. Edges are restricted to adhering to the surface samples with this measure. For each pixel extend its four edges the maximum distance in the range image that satisfies these three conditions. (1) The connecting edge maintains a minimum perpendicular distance of at most $t_{r4}$ to all its intermediate pixels. (2) All consecutive pairs of points are connected as described in section 4.2. (3) A maximum of $N_e$ intermediate edges are included. The first condition limits from deviating from an underlying surface. The second condition is needed at object boundaries to prevent an edge following a surface from connecting to another object. The third condition ensures that for flat surfaces the end-points of edges are distributed along the surface rather than all at the boundary pixel. The result of doing this is illustrated in Fig. 3 where we see a significant reduction in surface normal noise.

4.4 Initial Pixel Categories

In preparation for evidence accumulation pixels are categorized using their slope and a ground model. Any appropriate ground detection method could be used. Here a simple robust surface is fit to the nearby points to identify ground pixels within 20 meters. At ranges beyond this, pixels whose normals all point up are labeled as ground. In some cases these pixels may be on obstacles, but since they are viewed from a grazing angle they are not near the front edge of the obstacle.

In addition to ground we categorize steep and sloped pixels based on their slope. Each pixel feature has up to four facets illustrated in Fig. 1(b) plus an edge to the above and below neighbors. (Narrow objects such as
Figure 3. Illustration of the effect of extending the left-right edges. Points are from a single horizontal slice from the Lidar observing a flat section of brick wall, (a) and (b), and a tree trunk, (c) and (d), with the sensor at the origin. Edges connect neighbors normals in the plane are shown in red. When adjacent pixels are used to define edges, (a) and (c), the normals have high variance. After edge extensions, (b) and (d), the resulting normals represent the surface much better. Also perpendicular lines through the edge mid points intersect much better at the center of the trunk (dashed green).

thin poles may have only a single column of pixels and so there will be no left-right neighbors and hence no facets). Each facet has a unit normal defining an angle from the horizontal. Also the normals to the above and below edges define an angle with the horizontal ray to sensor. The minimum of all these angles is selected as the steepness of a pixel. Steep pixels are defined to have angle under 12.5° and sloped have an angle less than 45°.

5. OBSTACLE EVIDENCE

This paper seeks to model and estimate local surface shape of obstacles or obstacle parts as a means to discriminating them from foliage. These obstacle parts are captured using a cell-based map.

5.1 Cells

Discrimination is performed using local information within cells. An important choice is the size of cells; the scale should be large enough to accumulate sufficient evidence while small enough that simple curved and flat shape models will suffice to model obstacle parts. Our choice was to use cylindrical cells with radius 0.5 m and height 1 m. This is sufficient to capture visible portions of poles, cones, barrels, tree trunks, parts of walls and other barriers that are typical obstacles to mobile robots. Each cell accumulates evidence independently for obstacle parts and foliage and there is no need to cluster or segment the objects as in.12 This is important as gives or full object segmentation is difficult in cluttered natural scenes.

Cells are only needed where there are Lidar points and so a sparse distribution of cells covering the 3D points is sufficient to model each depth image. Since the position of a cell relative to the pixels in it can impact detection, cells are distributed so that they overlap by 50% in each direction. The method for placing cells is as follows. Sloped points in the depth image are sorted by their distance to the sensor. Starting from the closest point, a cell is positioned with the pixel at its center and all other pixels within its boundary are included. Those pixels extending half way to the boundary in each direction are removed from the sorted list, and the process repeats until all pixels are covered by at least one cell. Finally we remove cells with three or fewer pixels. Fig. 4(c) illustrates cells distributed over a small region of a map.
Figure 4. (a) Lidar returns on a vehicle viewed at roughly 20m. (b) LPE features edges are shown. (c) The gray cylinders show all the overlapping cells. (d) Classification of cells: Blue is linear obstacle, red is curved obstacle, green is foliage. (d) Pixels classifications are inherited from cells. Pixels without a cell are marked as not classified. Notice that while most of the vehicle pixels are linear obstacles, the corner of the vehicle is classified as curved obstacle and the pixels around the wheel well are classified as foliage.

5.2 Cell Evidence

In classifying a cell we use the a posteriori probability ratio of a target, $T$, to foliage, $F$, given the Lidar data measurements, $D$ in the cell. This can be expanded using Bayes rule as:

$$\frac{P(T|D)}{P(F|D)} = \frac{P(D|T)P(T)}{P(D|F)P(F)}.$$  (1)

On the right hand side, the first term is the likelihood ratio of target to foliage and second is the prior probability ratio of target and foliage which is assumed to be unity. Now the likelihood, $P(D|T)$, for a target parameterized with a feature vector $\theta_T$ can be expressed as $P(D|T, \theta_T)$, and estimated with a probabilistic Hough transform. The Hough transform defines a log density function over a binned parameter space, $\theta_T$. Each pixel falling in a cell contributes towards the parameter density through its votes:

$$\log P(D|T, \theta_T) = B(\theta_T|D, T) + k.$$  (2)

Here the binning function $B(\theta_T|D, T) = \sum_j b(\theta_T|D_j, T)/N$ increments bins, $b$, of the discrete parameter space $\theta_T$ for each Lidar pixel $D_j \in D$. This is normalized by, $N$, the number of pixels falling in the cell, and is equivalent to modeling pixels as being fully correlated. Pixels that fall in a cell but do not contribute towards any bins still contribute towards this $N$. In a Hough transform the bin with the maximum value is selected, $\log P(D|T) = B_{max}(\theta_T|D, T) + k$ giving the maximum likelihood solution up to a scale factor (or offset in log space): $k$. For a single detector this offset can be accounted for in the threshold, but when combining multiple detectors it is important to account for it as is done below.
Figure 5. Average peak foliage response, \( B_f^{\max}(r) \), to each of the Hough detectors as a function of range from sensor, obtained by applying the detectors to labeled foliage data.

The same binned parameter model can be applied to foliage cells giving: \( \log P(D|F) = B_{\max}(\theta_T|D,F) + k \). Its average maximum response to foliage can be learned from example data. Since pixel sampling resolution affects the response, an average of the maximum binned response was calculated as a function of range to cell, \( r \), and is denoted: \( B_f^{\max}(r) \). The values calculated for this for the various detectors described in the next section are shown in Fig. 5. Using this, the log of the likelihood ratio for a target to foliage for a cell at range \( r \) is obtained as:

\[
\log \frac{P(D|T)}{P(D|F)} = B_{\max}(\theta_T|D,T) - B_f^{\max}(r).
\]  

(3)

The next section explores designs for these binned detectors.

5.3 Obstacle vs Foliage Detectors

Since obstacle parts can have significant variation in shapes, a bank of detectors for a variety different obstacle shapes is created, and their outputs are combined to produce a likelihood ratio for obstacle versus foliage in each cell. This section describes these detectors. We observe that Lidar rays penetrate foliage in a randomized fashion. This penetration has been modeled as an exponential decay\(^6\) and simply as scatter.\(^1\) Here we instead model foliage as a constant response, at a given range, to the obstacle detectors.

5.3.1 Curved Obstacles

Curved object filters are designed to detect obstacles such as poles, tree trunks, barrels, boulders and cones. These detectors are modeled with a length-two feature vector, \( \theta = (x,y) \), equal to the position of the object center in the horizontal plane. Assuming a known radius, \( R \), each horizontal edge of a LPE feature can predict the obstacle center, as illustrated in Fig. 6. Since the radius is not known, four detectors are created with radii 0.1, 0.2, 0.3 and 0.4m. An alternative would be to use all three points to predict the circle center, but this is more sensitive to noise. For each detector, the feature space, \( \theta \), is defined as a square region extending beyond the cell boundary by roughly \( R \), and discretized into uniform bins with size proportional to \( R \).

Before accumulating votes from pixel edges, some pre-filtering is applied to the pixel features. First only steep pixels predict obstacle centers and the remaining pixels within a cell are treated as non-target pixels. Second, edges whose length are too long are also treated as non-target pixels. If an edge is greater than \( 2R \) both ends cannot be on the target, but even this length is too large as it entails an edge cutting through the center of the obstacle. We settled on a maximum edge length of \( \sqrt{2R} \), corresponding to subtending 90 degrees at the obstacle center. Finally in some cases, such as thin vertical poles, only a single column of pixels are obtained, and the pixels have no horizontal edges. These are modeled by adjusting the smallest radius detector, \( R = 0.1m \), so that pixel features with no horizontal edges predict a center at radius \( R \) along the viewing ray behind the pixel.

Once pixel filtering is complete, the remaining edges predict obstacle centers for each of the four detectors and those are interpolated into the binned parameter space. Examples of these curved Hough-transform accumulators are shown in Figs. 7 and 8.
5.3.2 Linear Obstacles

Linear obstacle filters are designed to detect a flat, vertical or steeply sloped obstacle surfaces. These detectors are parameterized using 2D polar coordinates: $\theta = (d, \phi)$, where $\phi$ is the azimuth orientation of the normal relative to the ray to the sensor, and $d$ is the perpendicular distance to the center of the cell.

While all left-right edges can parameterize a wall object, we gather evidence from those with discriminative power. Two different tests are useful for this. (1) Edges belonging to steep pixels, and (2) edges from pixels where the left and right edges are parallel. While both of these conditions are often satisfied for wall obstacles, in some cases such as near corners only the first applies, and in other cases such as low walls or sloped walls only the second applies. Hence two separate linear detectors are created, the first collecting evidence from LPE features that satisfy condition (1), and the second condition (2). In both detectors the left and right edges of the LPE add votes to the parameters space based on their horizontal normals and their perpendicular distance to the cell center. Examples of these linear obstacle Hough-transform accumulators are shown in Fig. 7.

5.4 Object Discrimination

The bank of detectors produces an array of log likelihood ratios for different obstacle types. These could be combined through a discriminative classifier, but here for simplicity the highest response is selected. This produces a likelihood ratio of obstacle to target for the cell. Individual pixels are classified according to the cells in which they fall.

6. RESULTS

Data were collected from the Velodyne-32 mounted on a Husky mobile robot in a variety of outdoor environments. All-foliage environments included a corn field, soybean field and tall grasses. An all-obstacle environment was a parking lot with stationary vehicles. Mixed environments included trees, tree-trunks, lamp posts, cones, poles, pillars, and building facades. The data points were labeled as wall-obstacle, curve-obstacle or foliage to enable filter response measurement on foliage as well as quantitative evaluation.

The same fixed values for the pixel-feature thresholds were used throughout. These were pixel connection: $t_{r_1} = 0.2m$, pixel separation $t_{r_2} = 1.2m$, distance to mid-point of neighbors $t_{r_3} = 0.05m + 0.0125r$, where $r$ is the range to pixel, and surface roughness, $t_{r_4} = 0.04m$.

The cell-based classification process is illustrated in Fig. 4. Examples of the filter responses on a cone and tree obstacles are shown in Figs. 7 and 8. Results on a cluttered natural scene with a number of obstacles including boxes, cones, and tree trunks are shown in Fig. 9. Notices that despite partial occlusions from branches, the tree trunks and other obstacles are detected with few false positives in the foliage.

Finally the overall performance of the method is shown in Fig. 10 as precision recall curves and average precision for cells. A comparison is shown to using the ratio of minimum to maximum eigenvalues of the cell pixel covariances as a scatter measure and hence foliage detector. This performs fairly well overall, but quite
Figure 7. A cell with its bank of detectors centered on a cone corresponding to object $H$ in Fig. 9. (a) The LPE edges are plotted on the 3D points. Ground points in brown are also shown but these are ignored by the detectors. The accumulated evidence from the curved obstacle Hough detectors with their radii are shown in (b) $r = 0.1$, (c) $r = 0.2$, (d) $r = 0.3$, and (e) $r = 0.4$. A top-down view of the features is plotted along with the cell circumference. The red circles are the predicted centers from each edge. Darker grid elements indicate more interpolated weight from these centers. The horizontal axis labeled $r_{xy}$ is the radial distance to the cell center and the vertical axis is the perpendicular. The green line indicates the viewing ray. In this case the smallest radius detector, (b), has the strongest response as the predicted centers are consistent. The linear obstacle detectors are shown in (f) and (g) parameterized by normal orientation $\phi$, and perpendicular distance, $d$. Neither has a strong response here.

poorly for curved obstacles which, like the cone in Fig. 7, are not linear or planar. The method proposed here has high discrimination performance for all obstacles including curved obstacles.

7. CONCLUSION

A new feature class is proposed that enables improved discrimination between obstacles and foliage with Lidar data. There are two aspects to this class. The first is LPE, and edge-based pixel feature that captures the local surface shape by smoothing for object roughness and sensor noise while respecting object boundaries. The second aspect is a cell-based feature that uses a bank of detectors to accumulate evidence for local obstacle shapes versus foliage. By robustly detecting obstacle parts it is not necessary to segment obstacles, and obstacles can be discriminated even in close proximity to foliage. This paper demonstrates the advantage of using local surface properties including normals from Lidar range-images for object discrimination, even when the sampling is coarse.

There are numerous ways in which this work can be extended. Markov network methods that bring contextual information could be used to improve discrimination. The output of the detector banks could be used to train a discriminative detector. In addition, other features and detectors could be added to the detector bank.
Figure 8. One of the cells on tree trunk $B$ in Fig. 9. A large number of non-trunk points surround the trunk and are included in the cell. The curved obstacle detectors corresponding to those in Fig. 7 are shown along with their responses. In this case there are no valid edges for the smallest radius in (b) and just a few for $r = 0.2m$ in (c). The strongest response is for $r = 0.3m$ in (d). Linear detectors are not shown and have low response.

REFERENCES


Figure 9. (a) A natural terrain scene containing foliage and partially occluded obstacles. The nine obstacles are indicated and labeled in the image. The estimated ground points are shown in the remaining plots are brown pixels. (b) Pixels in cells classified as convex are shown in red. All of the obstacles are detected and three small false positive regions marked with $X$ are shown. (c) Pixels in cells classified as wall-slopes are shown in blue, along with some false positives. (d) Detected obstacle and foliage pixels are all plotted.
Figure 10. (a) A precision-recall curve for obstacle cells out to 30m is shown, along with average precision (AP). This is compared to using ratio of minimum to maximum eigenvalue of pixel coordinate covariance. There are significantly more linear obstacles from walls than poles, barrels, tree-trunks etc. If only convex obstacles are considered out to 30m, then the precision-recall curve is that shown in (b). This shows a moderate drop in accuracy for our method but a dramatic drop in accuracy for the eigenvalue ratio.