International Master’s Thesis

Hauler Detection For An Autonomous Wheel Loader

Mano Prasanth Nandanavanam
Technology
Hauler Detection For An Autonomous Wheel Loader
Mano Prasanth Nandanavanam

Hauler Detection For An Autonomous Wheel Loader

Supervisors: Prof. Achim Lillienthal
Dr. Martin Magnusson
Examiners: Dr. Todor Stoyanov
Dr. Henrik Andreasson
Title: Hauler Detection For An Autonomous Wheel Loader

ISSN 1404-7225
Abstract

In this thesis work, we evaluate an object recognition system for an autonomous wheel loader, to detect objects in its vicinity, in particular an articulated hauler truck, by using an interest point extraction method that explicitly considers object borders information, combined with a feature descriptor known as Normal Aligned Radial Features (NARF) in 3D point cloud data. The object recognition technique relies on extraction of NARF from range images (computed from point clouds) for both model(hauler) and the scene. The technique used is robust feature matching where the extracted model features are mapped on to the scene containing the model and then seeking for a best transformation that aligns the model with respect to the scene.

In this context we conducted several experiments with many number of 3D scans obtained from the laser scanner mounted on the top of an autonomous wheel loader to analyze the accuracy of the object recognition system. Finally we demonstrated the results, as the system is able to recognize the hauler from any view point.
It gives me immense pleasure in acknowledging my supervisor Prof. Achim Lilienthal for offering me this amazing thesis.

My utmost gratitude goes to my supervisor Dr. Martin Magnusson for his patience, guidance and the time spent during the development of this thesis work.

Special thanks to all classmates and my close friends for their encouragement in these two years at the Örebro University.

Furthermore, I would like to thank Bastian Steder for his outstanding work, which benefited me a lot.

Finally, I would like to thank my family for their blessings and support they provided me. No matter the distance, they are always there for me.

Máno Prasánth Nandanavanam.
September, 2011. Örebro.
Contents

1 Motivation 3
  1.1 Problem Summary ............................................. 3
  1.2 Contributions .................................................. 4
  1.3 Thesis Outline .................................................. 4

2 Literature Review 5
  2.1 Introduction ...................................................... 5
  2.2 IP Detectors ....................................................... 5
  2.3 Feature Descriptors .............................................. 6
  2.4 NARF ............................................................... 9
    2.4.1 Border Extraction ........................................... 10
    2.4.2 Interest Point Extraction .................................. 12
    2.4.3 NARF Descriptor Calculation .............................. 13
  2.5 Comparison of NARF with Related Work ...................... 14

3 Object Recognition Using NARF 17
  3.1 Introduction ...................................................... 17
  3.2 Computing the Transformations ................................ 18
  3.3 Assigning Score to the Candidate Matches .................. 18
  3.4 Rejection of False Positives .................................. 19

4 Evaluation of Results 21
  4.1 Optimal Parameters ............................................. 23
  4.2 Success Rate ..................................................... 25
  4.3 Results Obtained For Complete Model ....................... 26
  4.4 Results Obtained For Single View of the Model ................ 28
  4.5 Comparison with Template Matching .......................... 30
  4.6 Timing Comparison .............................................. 31
  4.7 Results from Other Views ..................................... 32

5 Conclusions and Future Work 33
List of Figures

2.1 Shape Index ......................................................... 6
2.2 Spin Images ......................................................... 7
2.3 Integral Volume Descriptor ................................. 8
2.4 Point Cloud ......................................................... 9
2.5 Range Image ......................................................... 10
2.6 Borders .............................................................. 10
2.7 Range Image ......................................................... 14
2.8 NARF descriptor visualization .............................. 15
4.1 Autonomous Wheel Loader ................................. 21
4.2 Volvo A25 articulated hauler ............................... 22
4.3 Support Sizes for Model ........................................ 24
4.4 Support Sizes for Single View ............................. 24
4.5 Hauler Range Image ............................................. 26
4.6 Range image of the scene containing the model .... 26
4.7 Scene Interest Points, the points marked in circles 27
4.8 Model of the hauler (object in green) recognized in the scene shown in a 3D viewer 27
4.9 Final result with timings for each step .................. 27
4.10 In this figure the hauler appears to be in the same location but with an false orientation 28
4.11 Range image of the Single View hauler with interest points 29
4.12 Range image of the scene containing the model .... 29
4.13 Scene Interest Points ........................................... 29
4.14 Single view model of the hauler recognized in the scene shown in a 3D viewer 30
4.15 Final result for a single view model with timings for each step 30
LIST OF FIGURES
Chapter 1
Motivation

Fully Autonomous Wheel Loaders for Efficient Handling of Heterogeneous Materials (All-4-eHAM): In the construction and mining industry, there is a need to develop a generic, modularized system for autonomous wheel loaders that carries out all parts of the material handling cycle in the context of an asphalt production site. The ALL-4-eHAM project aims to solve the basic technical and scientific challenges required to produce such a system. Apart from the difficulties that arise from perceiving and navigating in a dynamic, unstructured outdoor environment, one important aspect lies in the unloading. The wheel loaders need to load heterogeneous materials from piles continuously heaped up by human operators and to transport the material, to unload it into pockets at predefined place or hauler. In this scenario the recognition of hauler is more challenging than the recognition of pockets, as the location of hauler is not predefined.

1.1 Problem Summary

In object recognition or mapping the ability to detect the object in an uncontrolled outdoor environment is a highly relevant problem. To solve this problem, many object recognition algorithms in raw 3D data use feature extraction for the purpose of object recognition. These features describe a set of data that can be used efficiently to perform comparison between different data regions. Such features are usually local around a point in the sense that for a given point in the scene its surrounding neighborhood is used to determine the corresponding feature. These features are normally computed only for specific so-called interest points (IP’s), and not all points of a scan. Important advantage of these IP’s are that they substantially reduce the search space and time required for finding the correspondence between two scenes and they focus the computation on the areas that are more likely relevant for feature matching process.

In this thesis work, we introduce an object recognition system for the autonomous wheel loader for recognizing the hauler in 3D scan data obtained
from a laser scanner mounted on the top of the wheel loader. This method relies on range images obtained from raw 3D laser data and is based on the extraction of NARF features from range images. Robust feature matching is performed by using a variant of GOODSAC [7] algorithm.

1.2 Contributions

The main contributions of this thesis work:

- Evaluation of NARF
- Object recognition for the autonomous wheel loader using NARF
- Comparison of results with template matching.

1.3 Thesis Outline

The organization of this work is as follows:

- **Chapter 2:** Introduces key concepts related to features and feature descriptors
- **Chapter 3:** Describes the object recognition technique that is implemented
- **Chapter 4:** Presents a discussion and comparison of the results obtained by the object recognition system.
- **Chapter 5:** Presents a summary of this thesis, along with the final remarks and directions for future work.
Chapter 2
Literature Review

2.1 Introduction

This chapter provides a brief overview of the research related to interest point detection and feature description for the purpose of object recognition in 3D range data. A number of reasons can be given for performing interest point detection but IP detection is usually an early processing step and can be seen as

- A way of discarding information which is not relevant at an early stage
- A way of saving computation time required for finding correspondences between two scenes

From the above context several IP detectors were proposed by considering certain points as interesting points in 3D range data.

2.2 IP Detectors

**Extremal points:** To obtain precisely located features, Thirion [12] introduced extremal points, which are points of locally maximal curvature in both principal directions of the surface. These points are specific points on the crest lines and can be seen as the generalization of corner points for smooth surfaces in 3D.

**Line-type Features:** Pauly et al. [8] presented a multi-scale feature extraction on point-sampled surfaces. For each point, the covariance matrix of the points in the local neighborhood was computed and its eigenvalues were calculated. By dividing the smallest eigenvalue by the trace of the matrix, a surface variation measurement was established. A scale-space was then be built by altering the size of the local neighborhood. At each data position a feature line that approximately passes along a ridge of maximum inflection was found by estimating the surface variation keeping all maxima above a certain threshold.
**Shape Index:** H. Chen and B. Bhanu [2] proposed a feature detector based on 'Shape index' a quantitative measure of the shape of a surface at a point defined by maximum and minimum principal curvatures. The feature points were extracted in areas with large shape variation measured by shape index, calculated from principal curvatures. In order to estimate the curvature of a point on the surface, a quadratic surface was fitted to a local window centered at that point and the least square method was used to estimate the parameters of the quadratic surface, and then differential geometry was used to calculate the surface normal, Gaussian, mean and principal curvatures. By their definition, Shape index \( S_i \), a quantitative measure of the shape of a surface at a point \( p \), defined by

\[
S_i(p) = \frac{1}{2} - \frac{1}{\pi} \tan^{-1} \frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)}
\]  

(2.1)

where \( k_1 \) and \( k_2 \) are maximum and minimum principal curvatures, respectively.

![Figure 2.1: A Range image and its shape index image (Note: Figure taken from [2])](image)

### 2.3 Feature Descriptors

The feature descriptors describe the area around a point in a way that makes efficient comparison regarding similar points. In recent years, many feature descriptors were proposed in 3D range data, most of them were invariant to rotation around the normal. The invariance in rotation can be considered as useful property as the same features can be detected even in multiple viewpoints.

**Spin Images:** Andrew E. Johnson and Martial Hebert [4] proposed a feature descriptor known as Spin-Images for surface matching. They described the sur-
face by a dense collection of oriented points (3D points with surface normal). A spin image was constructed in each data point by first selecting the points defining the local neighborhood. The normal of the current center point was then used to setup a grid of bins, which can spin around the normal. Spinning the grid around the normal covers a cylinder with a base equal to the support region. When the grid spins around the normal, all oriented points where the normal deviates from the center normal less than a specified angle are added to a 2D representation with the size of the grid. This produces a spin map. The resulting spin maps were divided into regions, one region was one pixel in the resulting spin image. These images, which were localized descriptions of the global shape of the surface, were invariant to rigid transformations and point correspondences between two surfaces were established through their correlation. When two surfaces have many point correspondences, they match. Taken together, the oriented points and associated images make up the surface representation. Since the image generation process was shown as a sheet spinning about the normal of a point, the images in their representation were called spin-images shown in 2.2.

Figure 2.2: A surface described by a polygonal surface mesh can be represented for matching as a set of 3-D points, surface normals and spin-images. (Note: figure taken from [4])

Point Signature: Beaudet [1] presented a point feature descriptor known as 'Point Signature' that was constructed around a point by encoding the distance from a locally fitted plane to 3D space curves around the point. The 3D space curves are the result of intersections between the surface and spheres with varying radii around the center point. The proposed method does not use any derivatives to estimate surface properties but establishes a local frame, which was referred as directional frame, for each point. The frame consists of a normal resulting from local plane fitting, a reference direction where the distance
curve was the largest and the cross product between the normal and directional frame.

**Integral Volume Descriptor:** Gelfand et. al [3] presented an approach to global registration using Integral Volume Descriptors (IVD’s) which were estimated at certain interest points in the data. The interest points are extracted using a self similarity approach in the IVD space, meaning the descriptor of a point was compared to the descriptors of its neighbors to determine areas where there is a significant change. In their method they developed local shape invariant descriptors that are based on integration instead of Differentiation of shape (which were less robust because of noise present in the input shape gets amplified when derivatives are computed). This invariant was defined at each vertex \( p \) of the input shape as follows,

\[
V_r(p) = \int_{B_r(p) \cap S} dx
\]  

(2.2)

Figure 2.3: Illustration of the volume integral descriptor in 2D. (a) Intersection of a ball of radius \( r \) centered at point \( p \) with the interior of the surface. (b) Discretization of the volume descriptor as computed. The cell size of the grid is \( \rho \). (Note: figure taken from [3])

Here the integration kernel \( B_r(p) \) is a ball of radius centered at a point and \( S \) is the interior of the surface represented by \( P \). The quantity \( V_r(p) \) is the volume of the intersection of the ball \( B_r(p) \) with the interior of the object defined by the input mesh. The invariant is illustrated in 2D in Figure 2.3. Assuming the intersection of \( \tilde{S} \) and \( B_r(p) \) is simply-connected, the volume descriptor related to mean curvature at \( p \) as follows,

\[
V_r(p) = \frac{2\pi}{3} \cdot r^3 - \frac{\pi H}{4} \cdot r^4 + O(r^5)
\]  

(2.3)

The leading term is the volume of the half-ball of radius \( r \), and the correction term involves the mean curvature \( H \) at the point \( p \).

The descriptor was calculated by a convolution of the occupancy voxel grid \( G_O \) with the ball grid \( G_B \). The value of the volume descriptor at each cell \( c \) is given by \( V(c) = (G_B \ast G_O)(c) \). The convolution can be computed efficiently by...
using the Fourier transform of the ball grid and the occupancy grid. The occupancy grid $G_0$ can be computed using scan conversion algorithms for meshes or ray shooting algorithms for point clouds.

## 2.4 NARF

Steder et al. [11] presented a novel IP extraction method that operates on range images generated from 3D point clouds see fig 2.4, (used to represent 3D information about the world and most often created by 3D scanners representing an ordered set of scan lines, providing range values). which explicitly considers the borders of the object found by transition from foreground to background. In their work they used range images see fig 2.5, which are visual representations of 3D scenes. Each point in a range image has both 2D position (the position of pixel) and 3D position (the measured position in the world coordinate frame). In addition to this, they proposed a feature descriptor NARF (Normal Aligned Radial Feature) that enables to extract a unique orientation around a normal. In their work, they developed NARF descriptor in a way that it should capture the existence of both occupied and free space so that the parts on the surface and outer shape of the object can be described and also the descriptor can be robust against the noise on the interest point position.

![Figure 2.4: Point cloud image of an office scene](image)

In their method they followed an accurate procedure to compute the NARF descriptors.

The following sections describe the entire procedure:
CHAPTER 2. LITERATURE REVIEW

2.4.1 Border Extraction

To extract borders for the range images they considered 3 kinds of borders: object borders, which were outermost points of an object, shadow borders, points that were part of the object’s background and veil points, points between the object and shadow caused by lidars. The occurrence these veil points is because of the scan lines at borders or edges of an object were considered as imaginary points as shown in 2.6) where the points that were considered at bottom of the chair.

At first in every image point, they applied a heuristic to find the 3D distance (Euclidean) from its 2D neighbors that belong to the same surface. The employed heuristic was:
• For each point \( p_i \) in the range image, its neighboring points \( \{ n_1, ..., n_{s^2} \} \) that lie in the square of size \( s \) are selected with \( p_i \) being in the middle. Then with respect to \( p_i \), all the 3D distances \( \{ d_0, ..., d_{s^2} \} \) of the neighboring points are calculated and sorted in an increasing order \( \{ d_0', ..., d_{s^2}' \} \). Assuming that at least a certain number \( M \) of the points lie on the same surface as \( p_i \), and selecting \( \delta = d_M \) as a typical distance to \( p_i \)'s neighbors so that points beyond a border are not included.

• In next step four scores for each image point were calculated based on \( \sigma \) and direction, describing the probability of having a border on top, left, right or bottom. The procedure for the border on right is described below: Assuming a point \( p(x, y) \) at position \( x, y \) in the image, the average 3D position of its neighbors to the right can be calculated by the following equation

\[
\bar{p}_{\text{right}} = \frac{1}{m_p} \sum_{i=1}^{m_p} p_{x+i, y},
\]

(2.4)

here \( m_p \) is the number of points used to calculate the average 3D position, the average is taken to consider the noise and possible existence of veil points. As the next step the 3D distance is calculated w.r.t to \( p(x, y) \) by

\[
d_{\text{right}} = ||p_{x, y} - \bar{p}_{\text{right}}||.
\]

Then a score is calculated based on the quotient of \( d_{\text{right}} \) and \( \delta \)

\[
s_{\text{right}} = \max(0, 1 - \frac{\delta}{d_{\text{right}}})
\]

(2.5)

This gives a value in \( [0, 1) \) where high values implies a substantial increase between the typical neighbor distance and the distance to the points on the right thereby indicating a probable border. A smoothing operation was applied on the score values to achieve continuous borders. To determine whether a given point \( p(x, y) \) was in the foreground or in the background its range value (distance to the original sensor position) is checked. If the range value of \( p(x, y) \) is higher than the range of \( \bar{p}_{\text{right}} \) then the border found is a shadow border and in other case its an obstacle border. To determine a shadow border, the point with highest score in a maximum 2D distance was selected. Depending on the score of this potential shadow border \( s_{\text{right}} \) is slightly decreased by

\[
s'_{\text{right}} = \max(0.9, 1 - (1 - s_{\text{shadow}})^3) \cdot s_{\text{right}}
\]

(2.6)

• As the last step, \( s'_{\text{right}} \) is checked if it is above a threshold (0.8 in their implementation) and if it appears maximum regarding \( p_{x-1, y} \) and \( p_{x+1, y} \) then \( p(x, y) \) is an obstacle border. If \( s'_{\text{right}} \) appears as minimum value corresponding to the neighbors of \( p(x, y) \) then \( p(x, y) \) is a shadow border and all the pixels in between are considered as veil points.
2.4.2 Interest Point Extraction

In general IP’s were extracted from corners and edges but in such regions, nearby points may have quite different appearance that could lead to the extraction of unstable interest points. As a solution, this method considers the points that are mainly in stable position of the surface, where there are sufficient changes in the local neighborhood, so that the point can be robustly detected even if observed from different perspectives. Since this method also considers the changes on the surfaces that are not related to borders, at the local neighborhood of each point the changes in the surface were determined by calculating the principal curvature directions at that point (principal direction and magnitude \( \lambda \)). Every point in the image was associated with a main direction (border direction in every border point and principal direction in other points) and a weight which is 1 for border point and \( 1 - (1 - \lambda)^3 \) for every other point.

In their approach, for every image point \( p \) they considered all its neighbors \( \{n_1, \ldots, n_N\} \) that are inside of a support size (diameter of a sphere around the interest point) which makes their method less sensitive to resolution, viewing distance and non-uniform point distribution. Since each of these points \( n_i \) has a main direction \( v_{n_i} \) and a weight \( w_{n_i} \), to reduce the influence of noise from normal estimation, the directions are projected onto a plane perpendicular to the direction of the sensor to \( p \) leading to a one dimensional angle \( \alpha_{n_i} \) for each \( n_i \). Since two opposite directions do not define a unique position and also the principal curvature does not provide a unique direction, the angles are transformed in the following way:

\[
\alpha' = \begin{cases} 
2 \cdot (\alpha - 180^\circ) & \text{for } \alpha > 90^\circ \\
2 \cdot (\alpha + 180^\circ) & \text{for } \alpha \leq -90^\circ 
\end{cases}
\]  

(2.7)

Then all the weights and angles are smoothed by applying a bounded Gaussian kernel and the interest value \( I(p) \) of point \( p \) is defined by following equations:

\[
I_1(p) = \min_i (1 - w_{n_i} \max(1 - \frac{10 \cdot ||p - n_i||}{\sigma}), 0) \quad (2.8)
\]

\[
f(n) = \sqrt{w_n (1 - 2 \cdot ||p - n||/\sigma - 1/2)} \quad (2.9)
\]

\[
I_2(p) = \max_{i,j} [f(n_i)f(n_j)(1 - |\cos(\alpha'_{n_i} - \alpha'_{n_j})|)] \quad (2.10)
\]

\[
I(p) = I_1(p) \cdot I_2(p) \quad (2.11)
\]

Here, one of the criteria of this method, putting the interest points in locally stable surface positions is achieved, as the term \( I_1 \) scales the term \( I_2 \) downwards.
if \( p \) has neighboring points with strong surface changes. The interest value is increased by the term \( I_2 \) if there exists a pair of neighbors with very different and strong main directions. Additional smoothing of the interest value is performed in every image point after the calculation. Finally, interest points are selected by considering all the maxima of \( I \) above a threshold.

### 2.4.3 NARF Descriptor Calculation

After extracting the interest points, for each of these points \( p_i \) a descriptor vector \( f_i \) is computed, that captures the structure of the object in the neighborhood of \( p_i \). In their previous work [10] they computed these descriptors by the following procedure. \( N(p_i) \) is a set of 3D points of the scene whose distance from \( p_i \) is below a given threshold and the normal \( n_i \) of the planar approximation of the \( N(p_i) \) is computed. In the next step, a point \( v_i \) that passes along the line through \( p_i \) is selected and is oriented according to \( n_i \). Then the observer position and direction are set at \( v_i \) and \( -n_i \). As the final step, \( f_i \) (descriptor vector) at a point \( p_i \) is computed by generating a range image that contains all the points in \( N(p_i) \) according to the computed observer position. By extracting the image patch from a viewpoint along the normal vector (computed by using PCA on all points, to increase the stability of descriptor), this method solves the 5 of 6 degrees of freedom of the relative transformation between two range images. The remaining degree of freedom is resolved by orienting the image patch along the z-axis so that the x-axis of the image patch will be always orthogonal to z-axis in the world coordinate.

After the computation of descriptor, all the points within the support size (\( \sigma/2 \)) are transformed in to world coordinate frame. The cell of the descriptor in which a point falls is defined by the resulting x and y coordinates and the value of cell is defined by minimum overall z values. The cell where no 3D points fall will get the maximum value of support size \( \sigma/2 \). As the next step, the image patch is blurred by Gaussian and a star shaped pattern with \( n \) beams is projected on to it, where \( n \) corresponding to the size of NARF descriptor. For each beam \( b_i \), a set of cells \( c_0, ..., c_n \) that lie under the beam are selected with \( c_0 \) being the middle of the patch and the rest were placed according to the distance to \( c_0 \). In this case, the value of the \( i \)th descriptor cell \( D_i \) can be known from below equations.

\[
\begin{align*}
    w(c_j) &= 2 - 2 \cdot \frac{\|c_j - c_0\|}{\sigma} \\
    D_i' &= \frac{\sum_{j=0}^{m-1} (w(c_j) \cdot (c_{j+1} - c_j))}{\sum_{j=0}^{m-1} w(c_j)} \\
    D_i &= \frac{\text{atan}2(D_i', \frac{\sigma}{2})}{180^0}
\end{align*}
\] (2.12) (2.13) (2.14)
where $w(c_j)$ is a distance-based weighting factor, where the weight in the middle of the patch is 2 and decrease to 1 when approaching the outer edges of the patch.

Now to extract one or more unique orientation for the descriptor, the beams are discretized in to 360° bins which is shown in fig 2.8 and a histogram is created and the maximum bin value of the histogram is selected as the dominant orientation of the patch. Finally the descriptor is shifted according to the extracted orientation to make it invariant to rotation. If the descriptor cell $D_i$ is closer to the center of the patch then the surface changes, stronger the change, more the beam value deviates from 0. This can be noted from figures 2.7 and 2.8, where beams lie on a flat surface have low values whereas the beams going over the border have high values.

![Figure 2.7](image)

Figure 2.7: A range image of an example scene with an armchair in the front. The black cross marks the position of an interest point. (Note: figure taken from [11])

### 2.5 Comparison of NARF with Related Work

As we seen the way of selecting the interest points, this method can be more accurate when compared to the other IP’s extraction methods that were discussed earlier. All these methods extracted IP’s from areas like corners in Thirion [12], ridges in Pauly [8] and large surface variations in Chen [2] while these areas are indeed interesting points in the scan, having interest points directly on corners or other positions that have a significant change in structure can lead to high inaccuracies in calculating descriptor, as these areas are mainly unstable. This method considered the points that were in stable surfaces along with borders because stable interest points ensure robust estimation of normals and in turn can make the overall process more robust.

The NARF descriptor computation described in this section has shown the way of capturing the existence of occupied and free space, so that the parts of the object and also the outer shape of the object can be described. This method can be more reliable when compared to Johnson [4] which does not take the
2.5. COMPARISON OF NARF WITH RELATED WORK

Figure 2.8: The top shows a range value patch of the top right corner of the armchair. The actual descriptor is visualized on the bottom. Each of the 20 cells of the descriptor corresponds to one of the beams (green) visualized in the patch, with two of the correspondences marked with arrows. The additional (red) arrow pointing to the top right shows the extracted dominant orientation. (Note: figure taken from [11])

space beyond object borders in to account. In [4] the spin images for the points in the center of a square plane and in the corners would be identical, on the other hand the method described in this section can discriminate between those points. In [3] they proposed the similar way of taking the descriptor information explicitly, but in their method it may become impractical and expensive to extract more complex descriptors. The descriptors presented in Beaudet [1], Johnson [4] and Gelfand [3] were normally invariant to rotation around normal. But in the case of NARF descriptor, can extract a unique orientation around the normal, which is an added advantage as the unique orientation can provide additional filtering for consistent local frames between features in turn improving the overall robustness.
Chapter 3
Object Recognition Using NARF

3.1 Introduction

The previous chapters introduced the main concepts of IP detectors and descriptors, as well as the extraction of NARF features from range images for an object. As it was discussed before, the NARF features were extracted by detecting the interest points (also includes the information from borders, which can be extracted by border extraction method) and then computing the NARF descriptor in an interest point in a way to make them invariant to rotation. These NARF features can be used for detecting several kind of objects. For the purpose of object detection using NARF features, an object recognition algorithm must be implemented.

This chapter introduces an object recognition technique in 3D point clouds developed by Steder et al. [10] for the purpose of object detection in the ALL-4-eHAM project. Specifically, the goal is to detect a Volvo A25 articulated hauler in laser scans made from a slowly moving platform. The basic idea of this method is feature matching, where the extracted features from the model are mapped on to the features in a scene containing the model. The main advantage of this method is it operates directly on range images and requires less computation time while retaining the robustness of feature matching.

The overall object detection procedure works as follows:

- When new scan is acquired, the corresponding range image of the scan is computed. Then the features are extracted from the range image for both scene and the model (extracted only once and stored).

- As the next step, the extracted NARF features in the scene are compared with the model features. A set of potential alignments are made from the set of corresponding features (best matches) by ranking the solutions based on a score function.
Finally, the pose of the object is estimated by validating each solution based overlap between the range image of the scene and range image of the model.

All the above mentioned steps are discussed in detail in the following sections.

Before going into details a brief description of feature matching procedure is as follows: The feature matching is made by treating the small range image patches as vectors and searching for the closest neighbors in feature space. For each feature in the scene $f^s_i$ and in the model $f^m_j$, the Euclidean distance $d(f^s_i, f^m_j)$ between their descriptors is computed. Based on the distance, a set of potential correspondences $C = c_{ij}$ (given by all feature pairs whose distances are below a given threshold) are created. To reject the false feature matches a method similar to GOODSAC [7] algorithm is used, which selects only good feature matches by sorting them based on the Euclidean distance between the descriptors.

### 3.2 Computing the Transformations

The alignment of range image of model with that of the scene can be done by knowing at least three correspondences between the features in both scene and the model, so that the 3D transformation between them can be computed. Then this procedure works as follows: Since a possible object pose can be determined by three correspondences, the number of possible transformations is proportional to $|C|^3$. The method described in this section provides a way to limit the computational requirements by sorting the correspondences based on their descriptor distances and also by considering the best $n$ correspondences. Triples of correspondences can be selected from $C'$, which is an ordered set of elements $c'_1, ..., c'_n$ by means of an algorithm that searches the best correspondences based on the descriptor distance and generates the sequence of correspondences like $(c'_1, c'_2, c'_3)$, $(c'_1, c'_2, c'_4)$ in this case the search will not get stuck if one of the correspondences with a low descriptor distance found to be a false match. Then the current triple of correspondences $c'_a, c'_b, c'_c \in C'$ is used to compute a candidate transformation. The next step is to validate this transformation which is described in following section.

### 3.3 Assigning Score to the Candidate Matches

The score of a transformation is computed by sampling different views of the object and selecting a fixed number $n$ of uniformly distributed 3D points (validation points) from the resulting range images. These validation points, whose viewpoint is closest to the transformation are considered for each candidate transformation. To check if the candidate transformation is correct, the depth values of the original points in the scene range image $D^s = \{d^s_1, ..., d^s_n\}$ should be same as the depth values of transformed validation points $D^v = \{d^v_1, ..., d^v_n\}$. 
3.4. REJECTION OF FALSE POSITIVES

So for each validation point, a scoring function is chosen which is explained by the following equation.

\[
 s(d^y_i, d^s_i) = \begin{cases} 
    d^y_i - d^s_i < -\epsilon : 0.0 \\
    |d^y_i - d^s_i| < \epsilon : 1 - \frac{|d^y_i - d^s_i|}{\epsilon} \\
    d^y_i - d^s_i > \epsilon : -p 
\end{cases} 
\]  

(3.1)

where in [10], this equation was found to be incorrect, as in their paper they mentioned the second case of the above equation as \(|d^y_i - d^s_i|\), which means the highest scores would be considered from 0, which would be a wrong value.

In the above equation the term \(\epsilon\) is treated as maximal error value where the validation point can still be considered in right place. The explanation of the above equation is as follows: If \(d^y_i - d^s_i > \epsilon\), then the point in the scene is considered behind a point in the model, so to solve this situation a negative reward of \(-p\) (penalty of \(-p\) to the points in the scene as they are occluded by matched object) is given. The condition \(d^y_i - d^s_i < -\epsilon\) meant that something blocked the view of the object. Depending on the above conditions the score for the complete set of validation points is defined as

\[
 S(D^s, D^y) = \max(0.0, -\frac{\sum s(d^y_i, d^s_i)}{n}) 
\]  

(3.2)

The value of the score is always between 0.0 and 1.0 and once the score is computed, the same procedure is repeated for the next three correspondences. Since it requires only three correspondences to calculate a potential alignment, this algorithm can handle partial occlusion. This algorithm can return a possible position of an object in the scene for every transformation with a score above a certain threshold \(\gamma\).

3.4 Rejection of False Positives

The above procedure may return many instances of an object at similar locations. To reject the wrong instances, pruning of solutions is done by keeping only the objects with highest scores that are found in the scene. Then these pruned solutions are validated based on following criteria: the absence of collisions between neighboring solutions and the similarity between range images. The collision check can be efficiently performed by using kd-tree. If any collisions are found, the solution with lower score is rejected. To refine the pose of the object one of the variants of Iterative Closest Point (ICP) method known as FastICP [9] is used.
Chapter 4
Evaluation of Results

In this chapter, we present the results from the object recognition algorithm discussed in previous section. The object that needs to be recognized, in this case, is a Volvo A25 articulated hauler (shown in Figure 4.2). The scanner is mounted on the top of an automated wheel loader (shown in Figure 4.1), and the wheel loader should be able to reliably detect the hauler even when scanning while moving.

Figure 4.1: Autonomous Wheel Loader
Figure 4.2: Volvo A25 articulated hauler
4.1  Optimal Parameters

The developed object recognition system depends on various parameters mostly related to feature extraction such as support size, feature descriptor size and the maximal descriptor distance. The influence of these parameters is as follows:

- An increase in the support size makes the feature extraction phase more expensive during runtime, since larger radius has to be considered for every feature.

- Similarly an increase in the descriptor distance threshold (Euclidean distance between the descriptors vectors) increases the score value (best estimate) of the match as the number of candidate transformations increases.

- On the other hand an increase in feature descriptor size reduces runtime as it considers less number of candidate transformations.

To find the optimal values for these parameters, we tested the system with several scans of Articulated hauler. These scans were taken with an angular resolution of 1 degree by the scanner located at the top of autonomous wheel loader while it is moving.

**Final Parameters**  As discussed earlier our system depends mainly on maximal descriptor distance threshold, size of the descriptor and support size. For evaluating these parameters we used a very low score threshold $\gamma = 0.3$ (discussed in previous chapter) to reject false feature matches. As an initial step we evaluated each parameter individually by applying different values and then we checked all the parameters at a time to view their combined effect on the performance of the system.

**Effect of support size:** At first we evaluated the system with various support sizes from 2.5 m to 3.8 m, for each step we checked the number of interest points we obtained and also the number of true positives. As we increase the size from 2.0 m we observed the increase in the number of interest points and also in the total precision and recall rate of the system (explained in next section). We found the optimal value for the support size at 3.0 m which can be considered as approximately 25% of the total size of the object (considering the fact based on previous knowledge that 25% of the average object size is a reasonable value).

In the below (Figures 4.3 and 4.4) you can see the plot regarding various support sizes from 2.5 m to 3.8 m with corresponding precision and recall rates.

**Effect of descriptor distance threshold:** Since the maximal descriptor distance threshold would increase the score value (best estimate) of the match, to find the optimal value for this we tuned it with values between 0.1 (Score value of 0.571074) and 0.3 (Score value of 0.650524) and as we increased the values
CHAPTER 4. EVALUATION OF RESULTS

Figure 4.3: Plot that shows various support sizes tested for the complete model of the hauler on X-axis, Precision and Recall rates on y-axis

Figure 4.4: Plot that shows various support sizes tested for the single view of the hauler on X-axis, Precision and Recall rates on y-axis
we found that the final score increasing accordingly. Finally, we obtained the optimal value for descriptor distance threshold at 0.2 (Score value of 0.650524, after 0.2 the score value remained the same and the runtime started to increase as it leads to more number of candidate transformations).

**Effect of feature descriptor size:** The size of the feature descriptor has shown some minor effect on runtime. At first we tested with 6x6 size for the descriptor patch and we found out that at this size the feature is less descriptive as it is leading to more number of candidate transformations in turn increasing the runtime (0.170655 sec, tested for one scan). So we increased the size up to 8x8 and as we tested with this size, the features became more descriptive therefore reducing the candidate transformations along with the runtime (0.155343 sec tested with the same scan as above) and didn’t effect the scores. As we go on increasing the size of the patch, the scores dropped and so we finally considered 8x8 as an optimal value for the size.

### 4.2 Success Rate

In this experiment with the object recognition system, we tested 123 scans of the scene containing the articulated hauler with a single model of it. For each case, 10 views of the model were generated by the system where each view corresponds to the direction of model which can be judged by looking it in a 3D viewer. The success rate factor mainly depends on the accuracy of the model and also the optimal parameters discussed earlier. Here success rate can be determined in terms of precision and recall rates. These terms are explained in the following equations

\[
\text{Precision rate} = \frac{tp}{tp + fp} \quad (4.1)
\]

\[
\text{Recall rate} = \frac{tp}{tp + fn} \quad (4.2)
\]

In the above equations the term \( tp \) indicates true positives, \( fp \) indicates false positives and \( fn \) indicates false negatives. In this scenario a \( tp \) can be considered as a correct recognition of the model, if the model is found to be in the scene, the term \( fp \) can be considered as a scenario where the recognized model was not exactly at same location in the scene (or if the model is found to be in the same place with a false orientation) and finally the term \( fn \) can be defined as the indication which the system couldn’t be able to recognize the model, even the model was present in the scene.
4.3 Results Obtained For Complete Model

Out of 123 scans 17 scans were considered as true negatives (since there is no sign of hauler in those scans). So from the remaining 106 scans where the hauler in the scene appeared from right side with respect to the sensor position, the recall rate is 82.75% (with 72 true positives) and the precision rate is found out to be 79.12%.

The appearance of false positives in some of the scans was found to be the case, where the model in the scene is at the same location but with a false orientation. In All-4-eham project, considering the fact that the resulted false positives (where the hauler appears upside down) can be still considered as a true positives, in this case the precision and recall rates will be 100% and 85.84%. The figure 4.10 shows one of the false positive occurrence.

The following figures show the results obtained for complete model for of one of the scan where right side of the hauler appears in the scene with respect to the sensor position.

Figure 4.5: Range image of the hauler model

Figure 4.6: Range image of the scene containing the model
4.3. RESULTS OBTAINED FOR COMPLETE MODEL

Figure 4.7: Scene Interest Points, the points marked in circles

Figure 4.8: Model of the hauler (object in green) recognized in the scene shown in a 3D viewer

Figure 4.9: Final result with timings for each step
4.4 Results Obtained For Single View of the Model

In another experiment tested with the same 123 scans, where we used only a single view of the model, rather than the entire model for matching purpose. In this case the recall rate turned out to be 87.50% (only 12 false negatives) and the precision rate found to be 89.36% (with 10 false positives). If we consider the fact about false positives (which were upside down), mentioned in previous experiment, then the precision and recall rates will be 100% and 88.67%. The success rate is higher in this case because of the accuracy of the single view model and also considering the fact that system can easily detect if the orientation of the model and matched model in the scene were similar.

The following figures show the results obtained for single view of the model for one of the scan where right side of the hauler appears in the scene with respect to the sensor position.
4.4. RESULTS OBTAINED FOR SINGLE VIEW OF THE MODEL

Figure 4.11: Range image of the Single View hauler with interest points

Figure 4.12: Range image of the scene containing the model

Figure 4.13: Scene Interest Points
**CHAPTER 4. EVALUATION OF RESULTS**

Figure 4.14: Single view model of the hauler recognized in the scene shown in a 3D viewer

Figure 4.15: Final result for a single view model with timings for each step

4.5 **Comparison with Template Matching**

**Template matching:** Template matching is one of the object recognition method where the entire template or a sample of the object is matched to the corresponding object in the scene. This method is used when major part of the template image constitutes the matching image or when the features are not strong enough to compare.
**Brief Description of Algorithm:** In ALL-4-eHAM, the location of hauler in not predefined both in the asphalt-mill scenario and the short-cycle scenario. The algorithm for hauler detection is described below.

- First classify the current point cloud, looking for points in planar regions (using the same surface classification method described in the ALL-4-eHAM work on pile detection [5]). Using principal component analysis (PCA), the orientation of each substantial cluster of planar points can be found, and oriented in the same direction as the template model. A set of estimates is generated for each such planar cluster, with a small amount of noise added to the position and orientation.

- For each estimate, the current scan is registered (using NDT [6]) to the template, updating the pose of the estimate to the output of the registration algorithm.

- The confidence of each particle is measured using the Hessian matrix of the NDT score function at the solution pose. The Hessian corresponds to the covariance matrix of the six transformation parameters (3D translation and rotation). A threshold on the confidence is used to determine if an instance of the hauler was correctly localized or not. If no estimate reaches this threshold, the algorithm returns no detection.

**Success Rate:** In this experiment with template matching we tested the same 123 scans in which 17 scans turned out to be true negatives. The remaining 106 scans are tested with complete template of the object, the precision rate found out to be 96% (with 2 false positives) and recall rate is 41%.

In another experiment where we tested with above 106 scans with a single view of the model the performance increased a lot as the precision rate is 100% and the recall rate is 87%.

### 4.6 Timing Comparison

**Object Recognition using NARF**

- The average timing for the object recognition for complete model 0.14s
- The average timing for the object recognition for single view model 0.15s

**Object Recognition by Template Matching**

- The average timing for the object recognition for complete model 1.27s
- The average timing for the object recognition for single view model 0.92s
Finally, in comparison with template matching, NARF performed well in case of precision and recall rates and even in the case of timings as the object recognition with NARF took very less computation time. Since template matching considers mainly the planar surfaces, whereas NARF considers interest points so far the results from the NARF are much better than template matching. But, in few cases where the object is at a distance of 25m from the sensor position, the template matching algorithm performed better than NARF, which can be considered as a limitation to NARF.

4.7 Results from Other Views

To check whether the developed object recognition system can detect objects from other view points, we tested the system with only a few scans in each view point (hauler facing left, front and back to the sensor). We calculated the precision and recall rates for each view point:

- The precision and recall rates of the hauler, facing to its right according to the sensor position were found to be 80% and 100% (tested with 5 scans).

- The precision and recall rates of the hauler, facing to its back according to the sensor position were found to be 80% and 50% (tested with 10 scans).

- The precision and recall rates of the hauler, facing to its front according to the sensor position were found to be 66% and 50% (tested with 4 scans).

In all the three experiments the false positives were found to be upside down.
The goal of this thesis was to implement an object recognition system for an autonomous wheel loader for the recognition of hauler in an uncontrolled environment by using NARF (Normal Aligned Radial Features). This objective has been successfully achieved by implementing a feature based object recognition algorithm that mainly operates on 3D range images, derived from point clouds. These NARF’s that were extracted from range images along with the information obtained from border extraction was integrated to compute NARF descriptor.

By using the above mentioned NARF, the developed object recognition system was able to recognize the model of the hauler with respect to the scene. The developed system was invariant to rotation along the normal and also can recognize the hauler model from any view point. The performance of this system was put in to test by using several 3D scans of the hauler that was taken from the SICK LMS 291 laser scanner mounted on the top of the autonomous wheel loader. The total time taken to recognize the hauler was very impressive as it stands out to be 0.15 seconds on an average (using the complete model of the hauler).

The other goal of this thesis was to compare the results of developed object recognition system with another algorithm template matching. This objective was also achieved successfully as the developed object recognition system using NARF features performed better (with one limitation) than template matching with respect to accuracy of recognition and also time taken for the object recognition.

**Suggested future work directions are:**

- The accuracy of recognition after a certain distance 25m can be improved
- As the developed system takes less computation time, an object tracking system can be developed along with the object recognition.
Bibliography


