Grouping Sensory Primitives for Object Recognition and Tracking[*]

R. Madhavan, M. Foedisch, T. Chang and T. Hong
Intelligent Systems Division
National Institute of Standards and Technology
Gaithersburg, MD 20899-8230.
Email: $raj.madhavan@ieee.org$, $\{mike.foedisch, tommy.chang, tsai.hong\}$ $@nist.gov$

Abstract

In this paper, we describe our recent efforts in grouping sensory data into meaningful entities. Our grouping philosophy is based on perceptual organization principles using gestalt hypotheses where we impose structural regularity on sensory primitives stemming from a common underlying cause. We present results using field data from UGVs and outline the utility of our research in object recognition and tracking for autonomous vehicle navigation. In addition, we show how the grouping efforts can be useful for constructing symbolic topological maps when data from different sensing modalities are fused in a bottom-up and top-down fashion.

1. Introduction

The 4D/RCS (Real Time Control System) reference model architecture [1] developed at the National Institute of Standards and Technology (NIST) has been used for the development of a wide array of intelligent systems in both government and commercial applications, including control of Unmanned Ground Vehicles (UGVs) in unstructured and unknown environments. It consists of a multi-layered multi-resolutional hierarchy of computational nodes each containing elements of Sensory Processing (SP), World Modeling (WM), Value Judgment (VJ), Behavior Generation (BG), and a Knowledge Database (KD). An important competence required within the SP and WM modules is the ability to group various features (observed in the sensor data) into patterns, objects, events, and situations.

Grouping sensory data may seem quite straightforward and simplistic but it is a difficult problem for several reasons. The environment in which the sensor is intended to operate, the noise (either due to the physics of the sensors or due to vehicle travel) in the resulting measurement, occlusions and shadows that are introduced in the range data are few of the factors in an unknown and uncontrolled domain that make the realization of perfect grouping quite challenging. The output of the grouping algorithms can be used to find desired objects in a scene towards the segmentation and classification needs imposed by autonomous driving. By establishing relationships between hypothesized groupings, both at temporal and spatial levels, we can classify, recognize, and evaluate these objects, events, and situations by linking them to named symbolic data structures.

In general terms, perceptual grouping can be defined as the visual ability to extract image relations from lower-level image primitives (features) independent of the image content and group them to obtain meaningful higher-level structure. In computer vision, it is a mechanism for data-directed image processing. In the 1920s, research in perceptual grouping was initiated by Gestaltic psychologists who subscribe to the view that the *whole* is greater than the sum of its parts. Gestalt theory is concerned with the grouping of elements into wholes subject to four main types of factors: similarity, proximity, continuity and closure [10, 11]. An excellent collection of papers and discussion on perceptual organization can be found in [2].

In this paper, we describe our recent efforts in grouping sensory data into meaningful entities at the signal (raw data) level [13]. Our grouping philosophy is based on perceptual organization principles using gestalt hypotheses where we impose structural regularity on sensory primitives stemming from a common underlying cause. We present results using range data from UGVs and outline the utility of our research in object recognition and tracking for autonomous vehicle navigation. We also discuss how the grouping efforts can be useful for constructing symbolic topological maps when data from different sensing modalities are fused

[*] Commercial equipment and materials are identified in this paper in order to adequately specify certain procedures. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

in a bottom-up and top-down fashion.

This paper is organized as follows: Section 2 details the low-level grouping process. Section 2.1 describes a modified curb detection algorithm that was originally developed for generic obstacle detection on range images followed by the geometric grouping algorithm in Section 2.2. The associated results on 2D range data collected on a UGV are then presented in Section 2.3. Section 3 outlines a high-level symbolic grouping approach that complements the low-level grouping algorithm. Section 4 discusses potential application areas for the grouping algorithms developed in this paper. Section 5 provides conclusions and areas of continuing work.

2. Low-level Geometric Grouping

2.1. Curb Detection

In order to detect curbs, we treat curbs as obstacles and use a slightly modified generic obstacle detection algorithm designed for range data. The original obstacle detection algorithm is described in this section followed by a modified algorithm for curb detection.

2.1.1. Generic Obstacle Detection Within a path planning context for mobile robot navigation, obstacles can be classified into two kinds: those that stand out of the ground (positive obstacles), and those that lie below the ground (negative obstacles). The robot needs to know where the obstacles are located and how big they are so that it can plan a path around them. Finding obstacles in 3D point clouds resulting from LADAR or stereo involves determining the ground plane first. In man-made environments or where the ground is reasonably flat, a surface can be fitted to points believed to be on the ground (e.g., points close to the vehicle that are assumed to be on the ground). This surface can then be used as a reference against which points can be measured. Obstacles are defined as objects that project more than some distance d above or below the ground.

The positive obstacle detection algorithm works column by column in the LADAR range image [3]. The algorithm starts with a point, g, known to be on the ground. An initial ground value is assigned at the location where the front wheels of the vehicle touch the ground, known from Inertial Navigation System (INS) and Global Positioning System (GPS) sensors. Given point g, the algorithm processes upwards from the bottom pixel in the column to the top pixel, as follows:

- Let p_i be the i^{th}pixel in the column, where pixel 0 is at the bottom of the column. Let x_i, y_i, z_i be the Cartesian coordinates of p_i. Let g be the last known ground pixel in the column, initially obtained from the vehicle's position sensors.
- Compute the slope between the ground point, g, and the next pixel p_k. Pixel p_k is labeled a positive obstacle if

$$\frac{(z_k-z_g)^2}{(x_k-x_g)^2+(y_k-y_g)^2+(z_k-z_g)^2} \geq \sin^2(\alpha)$$

where α is a predefined constant representing the maximum allowed slope. The value of $\sin^2(\alpha)$ is constant, and is precomputed for efficiency.

Pixel p_k may fail the above test but still be a positive obstacle. This is because the slope test is a function of distance. The obstacle can be far from the current ground point due either to occlusion or to the resolution of the sensor which degrades as a function of distance. To resolve this ambiguity, the height of the obstacle is required to be greater than a constant, H. i.e., $|z_k - z_g| > H$. If p_k is not an obstacle, it is assumed to be ground and replaces g as the current ground pixel. The process iterates up the column with each pixel being compared to the closest ground pixel. If p_k is an obstacle, g is unchanged, and is compared with pixels as above, until another ground pixel is found. When this occurs, point g is set to the new ground pixel value and the process repeats. For example, in Figure 1, pixel 0 corresponds to the bottom of the vehicle wheel. Pixels 1, 2, 3, 8 and 9 are ground pixels. Pixel 4, 5, 6 and 7 are positive obstacles and the direction vectors shown on the bottom of Figure 1 indicates the vectors in which the slopes are determined. In a way, the algorithm is analogous to flooding; pixels 1, 2, 3, 8 and 9 are flooded because they have shallow slopes.

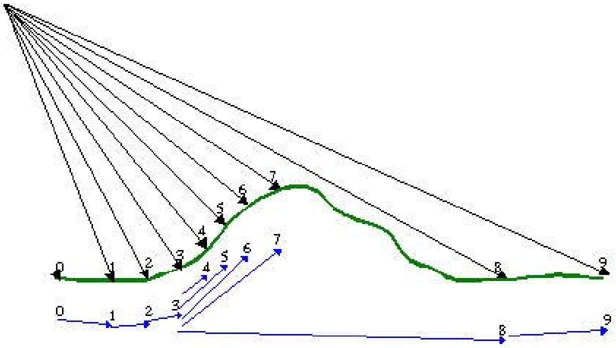


Figure 1. Positive obstacle detection.

2.1.2. Modification for Curb Detection In order to use the obstacle detection algorithm for detecting curbs in 2D laser rangefinder (SICK) data, we need to modify the algorithm slightly. First, the original algorithm assumes a range image data, corresponding to the road in which the vehicle travels. A column of pixels in this range image captures distances to the road surface and in the vehicle's traveling direction. In the case of curb detection, the 2D SICK data also capture distances to the road surface, but in the direction perpendicular to the vehicle's traveling direction. The original algorithm assumes the bottom pixel in the image column is ground (where the front vehicle wheels touch the road). This assumption is no longer valid in the curb detection application. Specifically, the bottom pixel in the SICK data corresponds to the left side of the road.

Thus it is necessary to split the SICK data into two at the middle. The middle of the SICK data corresponds to the center of the road (assuming the road is straight and SICK is sensing the cross-section of the road near the vehicle). The original algorithm without modification can now process each segment from the split SICK data from the middle. Finally, the curb is simply defined as the first obstacle detected in each of the two SICK data segments. The data points in between the two curb points are taken to be the road.

2.2. Geometric Grouping Algorithm

Once the curb is detected in the range sensor data, we proceed to identify other potential groups in non-curb data. The geometric grouping algorithm consists of the following three main steps:

G1. We determine *clusters* in the range data based on i) a distance threshold, d_{min}, and ii) number of points, n_{min} that satisfy d_{min}. The range scan is clustered based on discontinuities of the scanned range profiles. A new cluster is started whenever there is a significant discontinuity (indicative of object transition) as determined using the predefined thresholding distance, d_{min}. A small thresholding distance results in a large number of clusters. On the other hand, if a larger thresholding distance is chosen, some significant parts of the scan might be lost. So a trade-off between what to keep and what to throw away is to be made. After considerable trial and error analysis of the laser scans for the particular environment under consideration, d_{min} was arrived at. The clusters are identified by their centroids, $\mathbf{p_c} : (x_c, y_c)$.

G2. We determine *groups* by splitting or merging the clusters. For every cluster centroid in one scan, we compare its Euclidean distance with its counterpart in the next scan. If the distance between the centroids of the two clusters are within a predefined threshold, d_g, then these clusters are merged to form a group whose centroid $\mathbf{p_g} : (x_g, y_g)$ is the mean of the centroids of the group. This process is termed *straightforward association*.

G3. If the clusters are not within d_g, this might be a result of 1) a new group that is to be initiated or 2) the group (cluster) under question has been split due to occlusions or noise that may be introduced due to the pitching and rolling of the vehicle from which the scan was obtained and has to be so verified (meaning if the current cluster has to be merged to an existing group).
To verify which of the above two scenarios hold, we define an *Association Matrix*. This matrix computes an association of the clusters in the current scan with those in the (a) last and (b) second last scans by examining the distance d_g between them. If the association matrix for both cases (a) and (b) are the same, then the current cluster is part of an existing group; else a new group is initiated.
The association matrix enables spatio-temporal grouping thereby disambiguating association uncertainty that arises during the merging of clusters. The geometric grouping algorithm uses the perceptual grouping principles of proximity, similarity and continuity in a gestaltic sense.

An illustrative example is presented in the next section to enable the reader to better appreciate the geometric grouping algorithm.

2.3. Results and Discussion

We present results using range data obtained from a UGV during on-road driving. The environment is comprised of curbs, lane markings and intersections as the UGV shown in Figure 2 traversed relatively flat terrain. The SICK is a scanning laser rangefinder based on the time-of-flight principle. It has one scanning degree of freedom and is classified as a line ranging sensor. In our trials, the laser beam scanned through a $180°$ arc to give the distance to a target in cm at angular intervals of $0.5°$ with a maximum range of 50 m thus providing 361 range scan points.

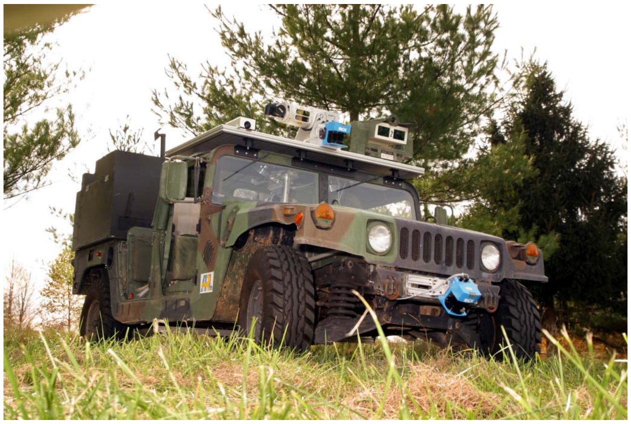
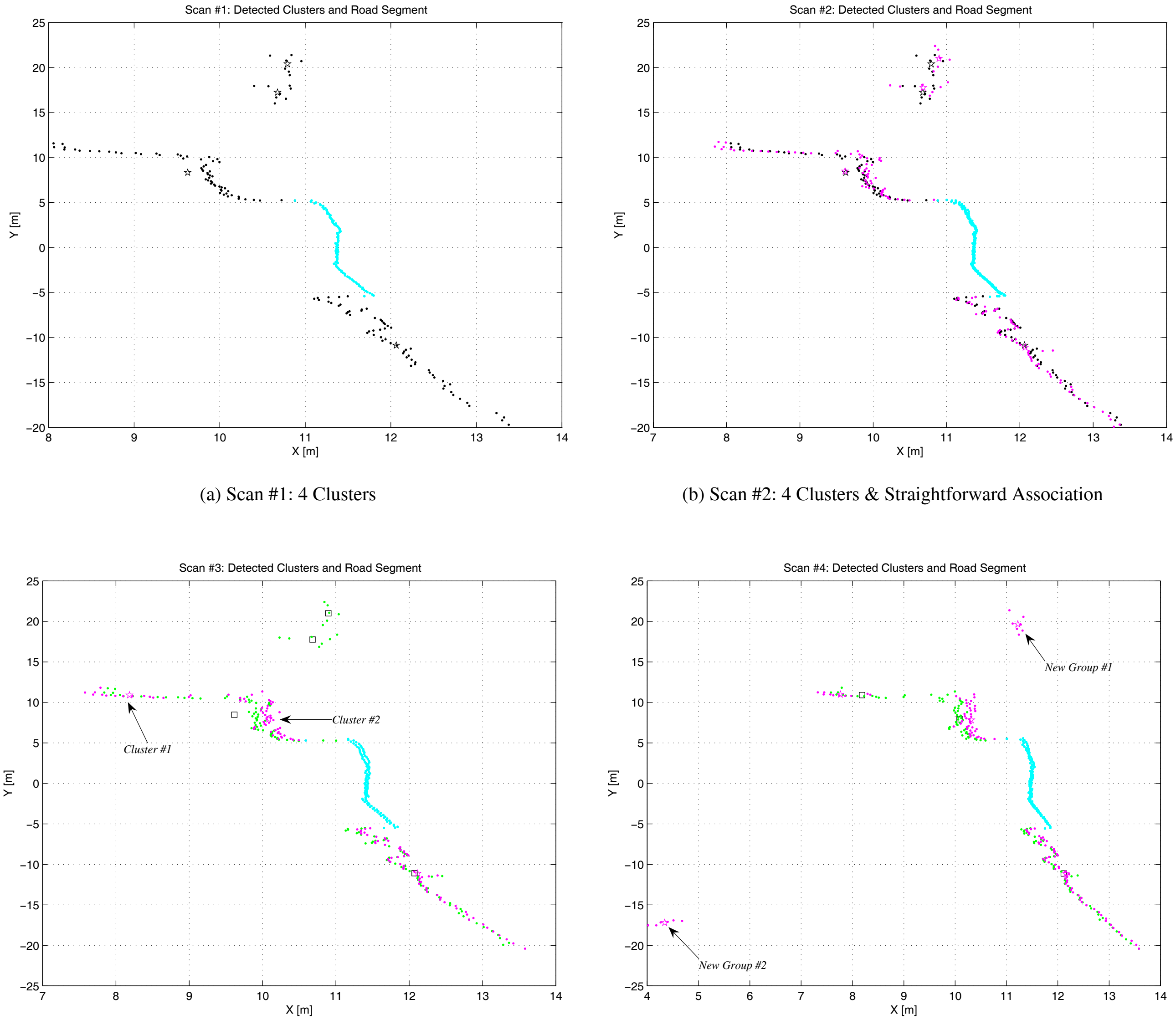


Figure 2. The roof-mounted laser rangefinder. The bumper-mounted rangefinder was not used in the results reported in this paper.

Figures 3 shows a sequence of laser scans based on which we will now illustrate the grouping algorithm. In all the scans, the laser is at (0,0) and the road (detected us-



(a) Scan #1: 4 Clusters

(b) Scan #2: 4 Clusters & Straightforward Association

(c) Scan #3: 3 Clusters & No New Group

(d) Scan #4: 5 Clusters & 2 New Groups

Figure 3. An Illustrative example for the geometric grouping algorithm.

ing the curb detection algorithm in Section 2.1) in front of the vehicle (sensor) is shown in cyan.

Figure 3(a) shows the clusters that have been extracted from a laser scan subject to the two conditions in Step **G1** (d_{min} and n_{min}). Four clusters have been identified in this scan #1 and are shown by black dots with their respective centroids denoted by black pentagrams. The next scan #2 and the extracted clusters are shown in Figure 3(b) by magenta dots and their centroids by magenta pentagrams (scan #1 is also shown in black for comparison). This scenario is a case of the aforementioned straightforward association where the previous four clusters are associated with the current four clusters and these are merged to form a new group as described in Step **G2**.

In Figures 3(c)-(d), the previous scans (clusters) are shown by green dots and their centroids by black squares for illustrative purposes. Figure 3(c) shows the three new clusters of scan #3 by magenta dots with their centroids shown by magenta pentagrams. For this scenario, we need to decide if two clusters of the current scan are part of the previous group (merging) or a new group has to be initiated (see area marked by arrows in Figure 3(c)). When the association matrix (described in Step **G3**) for this scan compared with the last scan and the second last scan are computed, it returns the same association which enables us to decide that the two clusters of scan #3 are to be associated with the already existing group that was formed in scan #2.

In scan #4 shown in Figure 3(d), five clusters are found (shown again by magenta dots and their centroids by pentagrams). Two new groups are initiated (marked by arrows) according to their association matrices that provide different associations with the immediate last and second last scans. It is evident that these clusters are to be initiated as new groups as there are no scan points in the their immediate neighborhood. This is a result of the UGV traversing the environment and new features coming into the view of the range sensor. The other clusters were associated with the already existing groups as they were found to be within the predefined inter-group distance, d_g.

3. High-level Symbolic Grouping

While current ladar/laser sensors provide a highly accurate but limited view of a part of the world, common video cameras allow a more holistic view of the scenery. Our previous work on road detection on color images (see [6, 7]) suggested the use of background knowledge (in terms of models) in order to improve the recognition results. In the following, we will describe our approach of a model-guided road recognition process. We will discuss the type of extracted features, the representation of models, and recognition process itself.

3.1. Feature Extraction

Assuming normal orientation of the vehicle on the road (see Figure 4), a simple set of features, which are easily extracted and well-understood, can be derived [5]. The features are based on "slices" of the road perpendicular to the direction of the vehicle. They can be extracted by applying one of several approaches for detecting the road area in images or road edge detecting algorithms (e.g. [4, 5, 6, 7, 12]).

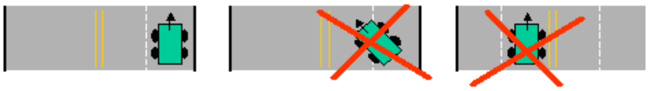


Figure 4. Normal orientation of the vehicle on the road. Legal (left) and illegal (center, right) orientations.

Figure 5 gives an example of the features. Starting at the bottom image row, the left and right road edge points in each row are determined. A pair of road edge points described in both image and world coordinates (through camera calibration) describes one feature item. The process continues bottom-up row-by-row until the world coordinates of the road edges reach a given maximum distance in front of the vehicle (e.g. more than 55 m). Furthermore, additional data will be associated with a feature item, e.g. information about lane markings.

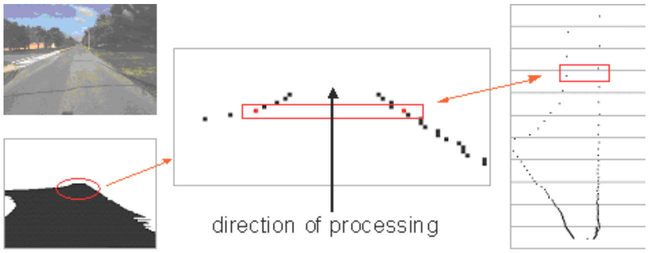


Figure 5. Feature Extraction - original image and road areas (left), road edge points (center), and projection into world coordinates (right).

3.2. Model Representation

Figure 6 depicts our approach for representing road model primitives. A "slice" of road is described by its width (geometrical component) and lane structure in terms of number of lanes and their legal direction (topological component). This representation of road primitives is compatible with the type of feature data described in Section 3.1.

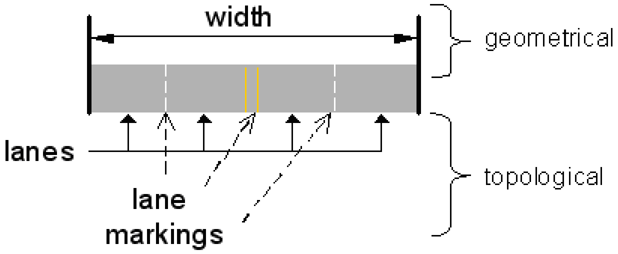


Figure 6. Geometrical and topological representation of a "slice" of road.

A road type consists of an ordered group of primitive road model items. For such groups additional constraints apply. A road type might require a minimal and/or maximal lateral length or, in the case of road widening and narrowing, a certain monotonic behavior. Other constraints limit the connectivity between (primitive) road types, e.g. a two-lane road segment can connect to a three-lane road segment only through a transitional segment. Primitive road items and road types are organized hierarchically. Additionally, primitive model items are grouped by the type of driving environment, e.g. highway driving, rural road or urban road driving. Appropriate connectors describe transitions from one environment to another one (e.g. a highway exit transfers the vehicle from highway driving to rural road driving).

3.3. Recognition Process

The goal of the recognition process is to find associations between feature items and (primitive) road models and eventually an interpretation of the scene. The application of a tree search algorithm spans potentially all possible associations of feature items and road models [8]. However, this process is computationally expensive and must therefore be constrained. We define constraints on three different levels, the primitive associations level, the group level, and the symbolic-level interpretation.

On the primitive level, potential associations must comply to unary constraints. For example, in order to associate a feature item to specific road model, the width of the road has to be similar for both entities.

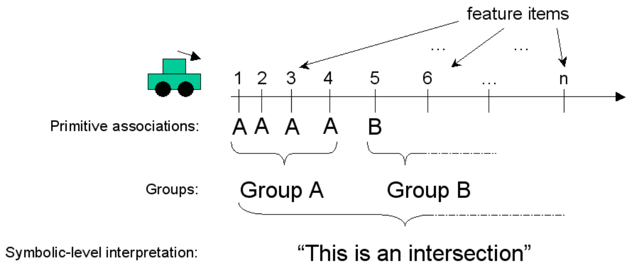


Figure 7. Recognition process levels: primitive associations level, group level, and symbolic-level interpretation.

Whenever a feature item is associated with the same model as the previous feature item, group-related constraints apply. Assuming that feature items *1*, *2*, and *3* in Figure 7 are already associated with a model *A*, the association of feature item *4* also to model *A* requires compliance of the extended *Group A* to group-related constraints. For example, in the case of a road widening (as part of an intersection) the group should comply to a certain monotonic behavior and the group's length should be within the maximal length of the model. Assuming another situation where feature items *1-4* are already associated to model *A*, associating feature item *5* to model *B* would trigger additional constraints. Starting a new group *B* causes the previous group *A* to be closed. This, for example, requires compliance to the minimum length constraint.

Finally, the set of (locally) consistent groups may allow a high-level interpretation of the scene. For example, the occurrence of a regular road segment, a widening segment, a narrowing segment, and another regular road segment (in this order) can be a strong indicator for the existence of an intersection.

4. Potential Application Areas

4.1. Integration of Geometric and Symbolic Grouping

While low-level grouping is data-directed (bottom-up), high-level grouping is model-directed (top-down) [9]. Combining such perceptual grouping provides advantages that are not achievable by either of them alone.

High-level processes gain confidence from accurate low-level grouping, e.g. a road recognition algorithm on video data can use accurate measures of the road (taken close to the vehicle) to extract distant road data. However, the interpretation of low-level data often depends on the context and its accuracy can be lost by mere local data processing. The integration of high-level symbolic data provides an opportunity to guide low-level processes, e.g. symbolic knowledge about an upcoming intersection can prevent a curb detection algorithm from erroneously extracting linear segments in curved areas.

4.2. Object Recognition and Tracking

Moving object recognition and tracking is critical, especially for on-road driving as the autonomous vehicle needs to make intelligent decisions on avoiding the obstacles and to plan appropriate routes. Detecting, recognizing (classifying) and tracking moving obstacles from a moving platform is a challenging task particularly in urban environments. The grouping algorithms presented in the previous sections of this paper provide a good starting point and are essential for tracking and recognition at the structural and primitive levels.

For example, we can group objects in the environment and classify them as either stationary or moving by tracking groups in continuous scans. Such information is valuable for avoiding collision with obstacles in the environment [14]. At the time this paper was written, we have demonstrated tracking moving objects in a stationary environment (i.e. the vehicle did not move between scans).

4.3. Efficient Path Planning

Using the grouping approaches presented in this paper, we can generate efficient routes for UGVs in unknown environments using noisy sensor data. The geometric grouping algorithm can provide a polygonal world model. Once we have such a world model, we are able to formulate optimal plans based on the incremental construction and analysis of a visibility graph. This is an area we are currently investigating using range data from a high-fidelity simulation environment.

5. Conclusions and Further Research

In this paper, we have described our recent efforts in using gestaltic principles of proximity, similarity and continuity for perceptual grouping of sensory data into meaningful entities. The low-level geometric algorithm was shown to be efficient in extracting the road in front of the vehicle using a curb detection algorithm and for grouping range data both temporally and spatially. The development of an association matrix within the geometric grouping process enabled spatio-temporal grouping by disambiguating association uncertainty for splitting and merging of groups. The high-level symbolic grouping described a model-guided road recognition process using background knowledge. The geometric and symbolic grouping processes were illustrated using experimental data collected from a UGV traversing urban environments. We discussed the advantages of integrating the geometric and symbolic grouping processes and how they can be complementary to one another. We also outlined the utility of the grouping algorithms in object recognition and tracking and for planning efficient routes for UGVs in unknown environments.

Further research work includes casting the geometric grouping algorithm in an extended Kalman filter (EKF) framework. From a given scan, the future scan can be predicted using vehicle speed and orientation and the difference between the predicted and the actual scan can be used to generate an error that could weight the grouping assignments via covariance matrices provided by the EKF.

For the high-level grouping, the results of the first experimental implementation have been encouraging. The current set of simple constraints have already vastly improved the efficiency of the tree search algorithm. Current research is investigating the enabling of real-time processing by taking advantage of the results of previous frames. For instance, pre-sorting of primitive models could guide the search algorithm and make it more efficient.

The integration of the geometric and symbolic grouping algorithms needs to be experimentally verified for an on-road driving scenario. We would also like to assess the performance of our grouping algorithms against ground truth.

References

[1] J. Albus et al. 4D/RCS Version 2.0: A Reference Model Architecture for Unmanned Vehicle Systems. Technical Report NISTIR 6910, National Institute of Standards and Technology, Gaitherburg, MD 20899, U.S.A., 2002.

[2] K. Boyer and S. Sarkar, editors. *Perceptual Organization for Artificial Vision Systems*. Kluwer Academic Publishers, 2000.

[3] T. Chang, T. Hong, S. Legowik, and M. Abrams. Concealment and Obstacle Detection for Autonomous Driving. In *Proceedings of the IASTED Robotics & Applications Conference*, Oct. 1999.

[4] R. Chapuis, R. Aufrere, and F. Chausse. Accurate Road Following and Reconstruction by Computer Vision. *IEEE Transactions on Intelligent Transportation Systems*, 3, Dec. 2002.

[5] D. DeMenthon and L. Davis. Reconstruction of a Road by Local Image Matches and Global 3D Optimization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1990.

[6] M. Foedisch and A. Takeuchi. Adaptive Real-Time Road Detection Using Neural Networks. In *Proceedings of the International Conference on Intelligent Transportation Systems*, Oct. 2004.

[7] M. Foedisch and A. Takeuchi. Adaptive Road Detection through Continuous Environment Learning. In *Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop*, Oct. 2004.

[8] W. Grimson. *Object Recognition by Computer: The Role of Geometric Constraints*. The MIT Press, 1990.

[9] H.-B. Kang and E. Walker. Multilevel Grouping: Combining Bottom-Up and Top-down Reasoning for Object Recognition. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, pages 559–562, Oct. 1994.

[10] G. Kanizsa. *Organization in Vision: Essays on Gestalt Perception*. Praeger Publishers, 1979.

[11] K. Koffka. *Principles of Gestalt Psychology*. Routledge and Kegan Paul Ltd., first published 1935, reprinted July 1999.

[12] M. Luetzeler and E. Dickmanns. Road Recognition with MarVEye. In *Proceedings of the IEEE International Conference on Intelligent Vehicles*, Oct. 1998.

[13] S. Sarkar and K. Boyer. Perceptual Organization in Computer Vision: A Review and a Proposal for a Classificatory Structure. *IEEE Trans. on Systems, Man, and Cybernetics*, 23(2):382–399, Apr. 1993.

[14] C.-C. Wang, C. Thorpe, and S. Thrun. Online Simultaneous Localization and Mapping with Detection and Tracking of Moving Objects: Theory and Results from a Ground Vehicle in Crowded Urban Areas. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 842–849, Sept. 2003.