Best Practices for Surveying and Mapping Roadways and Intersections for Connected Vehicle Applications

Task 5 Report: Feature Extraction Method

Executive Summary:

Connected Vehicle applications require established methods of roadway feature representation and reference in the form of a map. Numerous methods exist to acquire and record geospatial data indicative of roadway relevant objects and terrain. The preferred existing acquisition methods for roadway mapping utilize GPS integrated LIDAR sensing to generate a three dimensional georectified point cloud data set. When data acquisition is conducted with multiple integrated sensors at a high frequency the resulting dataset, especially when considered at the national or global scales necessary for commercial success, is sufficiently large to require automated application specific data processing for roadway relevant feature mapping.

This section presents an automated feature extraction approach explaining the data processing steps utilized to transform a georectified point cloud representation of the roadway environment to relevant intersection features represented in a SAE J2725 map message. SAE J2735 is an accepted communication media and associated map message format to represent intersection geometry and features in connected vehicle applications. The feature extraction methodology presented is intended to exemplify a uniform approach applicable to standardized intersections meeting accepted roadway design criteria. As such, the feature extraction methodology can serve as a template for feature extraction beyond the scope of J2735 based applications.

The roadway feature extraction process consists of the following primary steps:

- Preprocessing to extract the georectified point cloud and associated trajectory portions of interest for given intersections.
- Identification and extraction of the road surface point cloud, road edge curves, and median edge curves.
- Conversion of the intersection road surface point cloud to an image to enable additional image processing.
- Image-based roadway feature extraction.
- Translation to a J2735 feature map for intersections of interest.

The methodology of the feature extraction and map representation procedure are provided with detailed explanation of data processing and integration. The performance of the automated feature extraction approach is demonstrated using the eleven intersections along El Camino Real in Palo Alto California. This detailed approach allows for future feature extraction of relevant roadway features in a connected vehicle environment.

TABLE OF CONTENTS

I. INTRODUCTION

The output of the Mobile Terrestrial Laser Scanning (MTLS) mapping platform is a georectified 3D point cloud of the scanned area which is a set of four dimensional points representing reflection intensity and centimeter level accuracy Earth-referenced position. Additionally, as a byproduct of the georectification process the trajectory of the MTLS platform during the data collection process is available.

The point cloud contains information useful for mapping road features. However, it also contains large amounts of redundant or non-relevant sensor data. The objective of the feature extraction task is to extract relevant road feature information from the available MTLS data in a format, such as the SAE J2735 map message, that is useful for connected vehicle applications. The J2735 map message is designed for convenience of storage and communication between infrastructure and users. Such formats include metadata information such as number of lanes and whether lanes are intersection ingress or egress, which can be extracted from the MTLS platform trajectory data.

This Task 5 report presents the method and results used to extract J2735 map representations from MTLS data for eleven intersections along the Palo Alto testbed in California. For this study, the features of interest near intersections include: lane divider markings, road edges, center of intersection, and stop bars.

Section II describes this testbed, the data, and data acquisition. Section III describes the data processing for feature extraction. Section IV describes the feature extraction results. Sections V and VI discuss future work and conclusions of the study, including a list of best practices.

II. DATA

The point cloud data used in this study was collected by the AHMCT research center at UC Davis. The data is obtained by scanning Palo Alto testbed along El Camino Real. The MTLS includes a Trimble MX8 a Riegl VQ450 laser scanner and an Applanix POS 520 platform positioning system. The MX8 system has two LIDAR scanners, the left laser scanner has vehicle relative azimuth, roll and pitch angles of 143, -19 and 23.5 degrees respectively. The right scanner has vehicle relative azimuth, roll and pitch angles of -143, 19 and 23.5 degrees respectively [1]. The platform mounted on the vehicle is shown in Fig. 1.

Figure 1: AHMCT MTLS vehicle.

The testbed contains 11 signalized intersections. See Table 1. At each intersection, the MTLS platform was driven along each ingress and egress section of the main thoroughfare (El Camino Real) multiple times and along the cross street at least once. The multiple passes along each section provide a dense points cloud on the surface of

Table 1: Attributes of the testbed intersections

Cross street name	Center coordinates		Number of points in million
	Latitude	Longitude	
Stanford Ave	37.4277640	122.1492539	77.57
Cambridge Ave	37.4255950	122.1467851	75.36
California Ave	37.4250803	122.1458624	66.03
Page Mill Rd	37.4230638	122.1420467	95.04
Portage Ave/Hansen Way	37.4210617	122.1382334	154.58
Matadero Ave	37.4191602	122.1345428	68.68
Curtner Ave	37.4176288	122.1315912	70.04
Ventura Ave	37.4168366	122.1300875	50.64
Los Robles Ave	37.4157300	122.1281376	70.48
Maybell Ave	37.4120037	122.1246158	59.78
Charleston Ave	37.4104459	122.1233336	83.45
		Total Size	**871.65**
		Total Data Size	**2559.70**
		Discarded Points	**1688.05**

the street, which will facilitate reliable feature extraction. A second reason for the multiple passes along a road section is that the effect of dynamic obstructions (e.g., other vehicles) is unlikely to be the same in the multiple passes, so their effect is a slightly lower density of returns, instead of the absence of data that would be caused by a single run. The resulting dataset provides both MTLS georectified point cloud and the trajectory of the platform. The platform trajectory is shown in Fig 2a (left). The entire point cloud acquired along this trajectory contains approximately 2,560 million points.

For each intersection, the J2735 map message requires definition of the intersection center point. This is a designated reference point that should be near the center and later provided in the J2735 map message. All intersections features described in the J27235 for that intersection will be specified as offsets relative to the defined center.

Table 1 lists the names of the intersection cross streets, the latitude and longitude defined as the center of the intersection, and the number of points within a radius R=60 meters of the intersection center. On average each intersection contains around 80 million points on average. Of the 2,560 million points in the full point cloud, approximately 872 million are within the radius R of at least one of the intersection centers. The remaining 1688 million points are discarded.

III. INTERSECTION FEATURE EXTRACTION PROCEDURE

The full geo-rectified point cloud *P* contains a large number of points obtained while driving the vehicle along the roadway during the MTLS data acquisition process. To facilitate data processing, each intersection is processed independently.

The roadway features of interest for this project include lane markings such as lane edges and stop bars. These features are painted onto the road surface using high-reflectivity paint. The point cloud

P_i for the i-th intersection is designed to contain Lidar reflection intensity data from which the desired roadway lane markings and their position information can be extracted. Many other unwanted items with high reflectivity also exist within the roadway environment (e.g., signs and portions of vehicles or buildings). Because the desired lane markings are painted onto the road surface, many unwanted high reflectivity items can be removed by focusing the feature extraction process on the road surface; therefore, an important data processing step is to extract the subset of P_i that contains the road surface. The road surface is modeled as a flat and continuous two-dimensional surface along each ingress or egress portion of an intersection.

The high-level steps in the UCR MTLS roadway feature extraction process can be summarized as follows.

A. Preprocessing to extract the portions of the point cloud (P) and trajectory (T) relevant to each intersection. The portion of the point cloud and trajectory relevant to the i-th intersection will be denoted by P_i and T_i, respectively.
B. Extraction of road surface point cloud for each intersection. The portion of the point cloud P_i that corresponds to the road surface of the j-th section of the i-th intersection will be denoted by S_i^j. Additional extracted information from this step includes roadway inner (median) and outer (curb) edge data that may be extracted during the road surface extraction process. This is useful as these edges are often not marked with high-reflectance paint and therefore not detectable from the Lidar intensity data in later steps.
C. Conversion of the intersection road surface point cloud to an image. This step enables use of image processing tools that are well established and easily available.
D. Image-based roadway feature extraction.
E. Definition of the J2735 feature map for each intersection, including metadata concerning whether the j-th intersection is an ingress or egress, which is extracted from the trajectory subset T_i^j.

The subsections below provide detailed explanations for input data, output data, and process involved in each high-level step.

A. Preprocessing to extract Intersection Point Clouds

The accuracy of the geo-rectification process and simplicity of the MTLS data acquisition process are both enhanced by driving the vehicle through the entire sequence of intersections without turning off the sensor suite. Also, by leaving the sensor suite turned on while driving between intersections, the data may be useful for multiple purposes besides projects such as this that are mainly focused on intersections.

The data acquisition process, after geo-rectification provides the point cloud denoted by P and trajectory denoted by T. The point cloud P is too large to process conveniently and contains data not relevant to the intersections that are of interest. See Fig. 2a which shows the trajectory acquired for the Palo Alto corridor. The eleven intersections with centers as defined in Table 1 are marked with circles of radius R. The intersections are ordered with i=1 at the upper left and i=11 at the lower right. The fifth intersection is decomposed into two distinct parts with two centers due to its

unique shape. Fig. 2b shows a magnified portion of the trajectories in the vicinity of the sixth intersection.

To develop a computationally reasonable approach, the first step is to subdivide P and T into subsets relevant to each intersection. The center point of the i-th intersection will be denoted as C_i. These centers are user specified. For the Palto Alto dataset, the values listed in Table 1, are defined within the metadata of the dataset [1]. The user also specifies an intersection radius parameter R. Given these two parameters, we define for i = 1,…11:

$$P_i = \{\mathrm{x} \in \mathrm{P} \mid \|\mathrm{x} - \mathrm{C}_i\| < \mathrm{R}\} \text{ and } T_i = \{\mathrm{x} \in \mathrm{T} \mid \|\mathrm{x} - \mathrm{C}_i\| < \mathrm{R}\}. \quad (1)$$

The sets P_i and T_i will be processed to extract a J2735 feature map for the i-th intersection. That process will be repeated eleven times to extract the eleven intersection maps necessary for the Palo Alto corridor.

Once P_i and T_i are defined for all intersection then, the original data sets P and T can be decomposed as

$$P = \bigcup_{i=1}^{11} P_i + Q \text{ and } T = \bigcup_{i=1}^{11} T_i + U \quad (2)$$

where Q and U contain point cloud and trajectory data not relevant to the intersections, which will be ignored in subsequent processing. For the Palo Alto dataset, the quantity of points in each P_i and in Q is stated in Table 1.

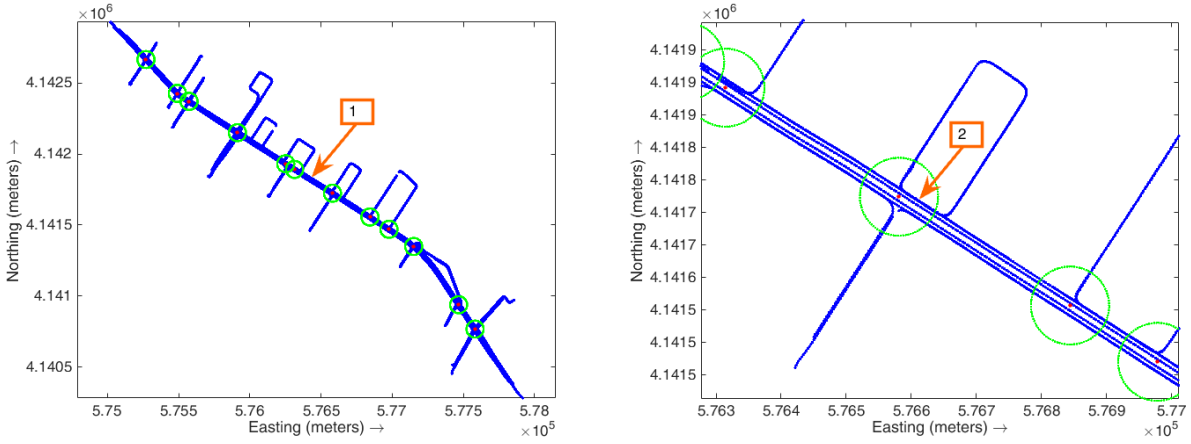


Figure 2: (a) The platform trajectory data for all 11 intersections. The orange box labeled 1 indicates the main thoroughfare. Each green circle encloses one intersection. (b) Magnified view of the trajectory through one intersection, clearly showing multiple traversals of the platform along the main thoroughfair.

To further automate the process in the future, the intersection centers C_i can be extracted automatically from the trajectory T.

B. Road Surface Extraction

The inputs for this processing step are the sets P_i and T_i defined in eqn. (1) of Section III.A. The goal of this step is to output a reduced point cloud that only contains those points on the roadway surface, thereby removing high reflectivity points from extraneous objects (e.g., buildings, vehicles, signs). Each extracted intersection is processed separately, so the process described in this section is repeated for i = 1, …, 11.

The roadway is modeled as a smooth 2-dimensional surface. This assumption is only locally valid within each well-defined ingress or egress section of the intersection; therefore, an important step is to process the intersection trajectory T_i to extract useful meta-data such as the number of intersection components and the direction of traffic flow (i.e., ingress or egress). This metadata will then be used to divide each P_i into regions where the 2-dimensional assumption is valid.

The road surface extraction process includes the following steps:

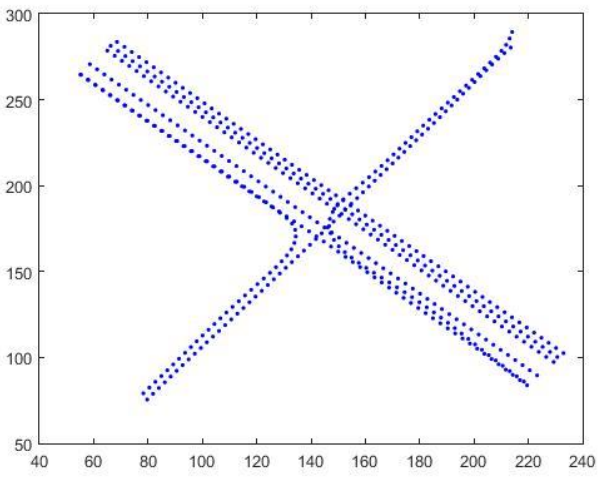
1. Definition of a set of polygons R_i^j each expected to contain the portion of one of the intersecting streets from the interior of the intersection to the boundary of P_i;
2. Process the portion of P_i within R_i^j to extract the road and median edges.
3. Extract point cloud subsets S_i^j for each road segment that only contain points on the road surface.

Each of these is discussed in the following subsections.

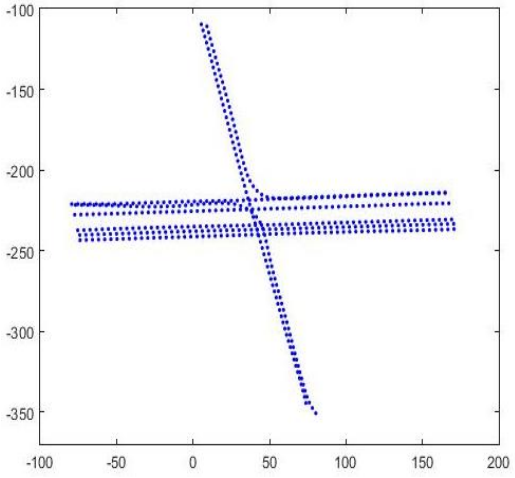
Definition of Intersection Regions

The point cloud P_i contains many points, only a small percentage of which are on the road surface. At the start of the processing the only information about the location of the road is the trajectory T_i and the center location C_i. The initial step uses T_i to define polygons R_i^j to decompose the point cloud P_i into overlapping regions that divide each cross street at the intersection and extend along that cross street to the boundary of P_i. For a T-intersection this would produce three polygons: j=1,…,3. For an X-shaped intersection, there would be four regions: j=1,…,4.

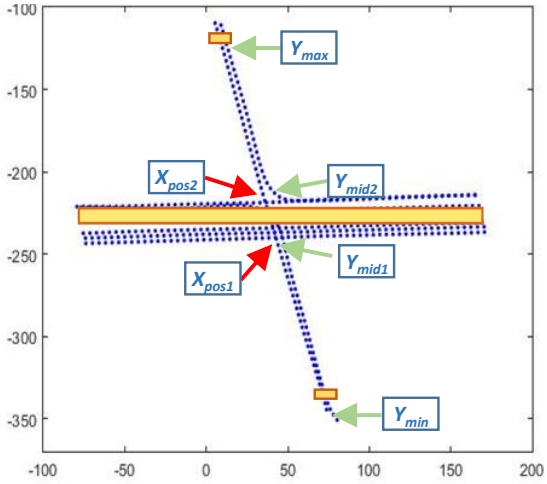
Fig. 3(a) shows the extracted trajectory data T_i for the sixth intersection. The trajectory data is first rotated to align the main thoroughfare with the horizontal axis, as illustrated in Fig. 3(b). Next, a sliding rectangular window, indicated by the yellow box in Fig. 3c, is moved vertically from the bottom to top of T_i (i.e., from Y_{min} to Y_{max} in Fig. 3(c)). The horizontal width of the trajectory within that vertical window is recorded as a function of the vertical position, as depicted in Fig. 3(d). As the sliding window passes through the main thoroughfare, the trajectory width greatly increases. The values of *y* at which this widening begins and ends are recorded as Y_{mid1} and Y_{mid2}. When the sliding window is in the position of Y_{mid1} (and Y_{mid2}), the mean of the horizontal position of the trajectory data within that sliding window is defined to be X_{pos1} (and X_{pos2}).



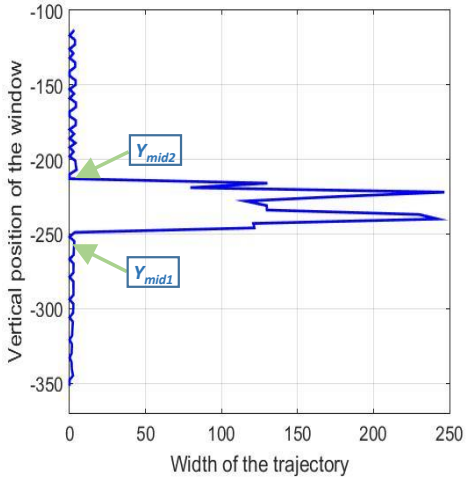
(a)



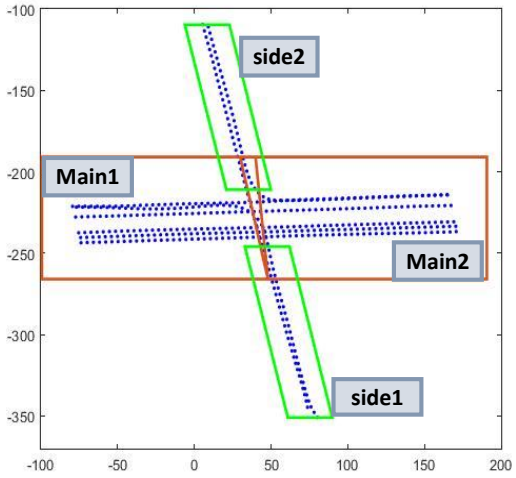
(b)



(c)



(d)



(e)

Figure 3: (a) Trajectory data T_i for i=6. (b) Rotated trajectory data. (c) Process for road segmentation. (d) Trajectory width as a function of vertical position. (e) Final output of the road segmentation process.

The first polygon for the main thoroughfare is defined to contain the corners defined by (X_{min}, Y_{mid1}), (X_{min}, Y_{mid2}), (X_{pos1}, Y_{mid1}) and (X_{pos2}, Y_{mid2}). The second polygon for the main thoroughfare is defined to contain the corners defined by (X_{max}, Y_{mid1}), (X_{max}, Y_{mid2}), (X_{pos1}, Y_{mid1}) and (X_{pos2}, Y_{mid2}). The first polygon for the cross street extends vertically between Y_{min} and Y_{mid1} with the x-values of the corners defined by the horizontal width of the trajectory at those y-values. The second polygon for the cross street extends vertically between Y_{mid2} and Y_{max} with the x-values of the corners defined by the horizontal width of the trajectory at those y-values. Each of these corners is expanded horizontally and vertically by a buffer value B=30 meters, because the road surface will be wider that the width defined based on the trajectory T_i. Fig. 3(e) shows the bounding polygons defined for intersection six.

Finally, these polygons are rotated back to the world frame and used to define

$$P_i^j = \left\{(x \in P_i) \bigcap \left(x \in R_i^j\right)\right\}$$

where $x \in R_i^j$ is interpreted as the horizontal components of *x* being in the interior of the polygon. This segmented intersection can be represented as

$$P_i = \bigcup_{j=1}^{J} P_i^j + G$$

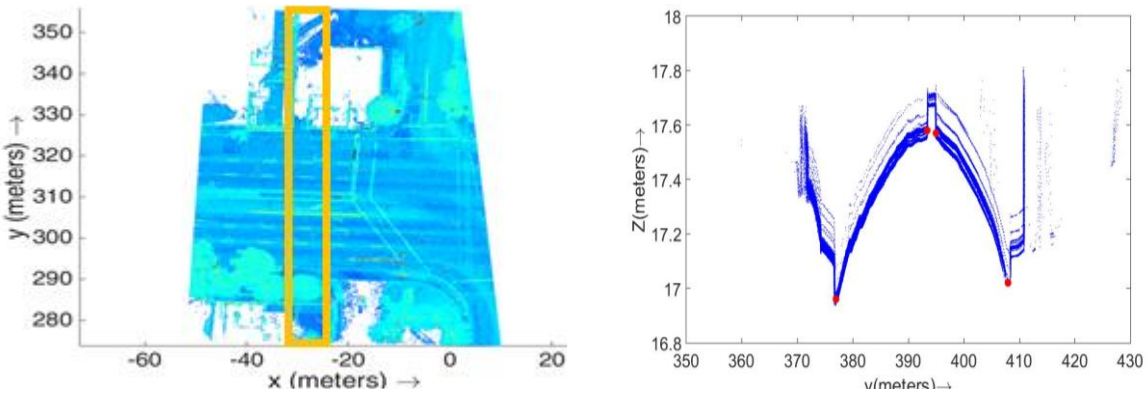
where *G* contains the elements of the point cloud that are not in any of the J polygons. The points in *G* will be ignored after this point. The value of *J* can be 3 or 4 depending on whether the intersection is "T" or "X" shaped.

Extraction of Road Edges per Section

Given the point cloud subsets P_i^j each containing a segment of an intersecting street, the next step is to extract the road and median edges for each segment. If any segment does not contain medians, only road edges are extracted.

For each segment, a one meter wide rectangular window of point cloud data is extracted and processed. See Fig. 4a. When this rectangle processing is complete, the rectangle slides along the road segment one meter and the process repeats. Within this narrow window, the road surface is assumed to be two-dimensional. This assumption is used to detect the road and median edges within the window. Once these edges are determined, discarding point cloud elements outside the road edges will automatically remove reflections from roadside entities such as buildings and trees.

Fig. 4b shows a sample of the point cloud elements within P_i^j and the sliding rectangle, with elevation denoted by *z* plotted versus cross-sectional position denoted by *y*. The processing records the mode of the *z*-value as a function of *y*. On the road surface, the mode is continuous. At the road and median edges this function changes abruptly. These abrupt changes are detected by monitoring both the derivative and inflection points of the curve. The detected points are indicated by red dots in Fig. 4b.



(a) (b)

Figure 4: (a) Intersection 6 road segment P_i^j. The yellow box depicts the rectangular window that is slid along the roadway (x- axis). (b) Elevation versus cross-sectional position for points within the sliding rectangle. Detected road and meridian edge points are marked with red dots.

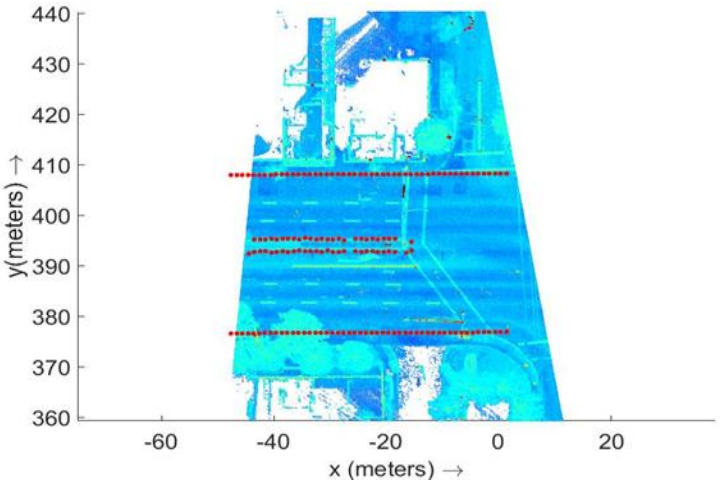


Figure 5: The estimated inner and outer edge points of a road segment after discarding the outliers.

After the road and median edges are extracted for each rectangle, each road and median edge is processed to remove outliers and insert estimate for missing data items. Results for one portion of intersection 6 are shown as red dots in Fig. 5. A curve is then defined for each road and median edge.

After the road edge extraction, points outside the road edges are discarded and the reduced point cloud

$$\bar{P}_i^j = \{x_m \in P_i^j \,|\, x_m \text{ is within the outer edges of the segment}\}$$

and the road and median curves are passed to the surface extraction process.

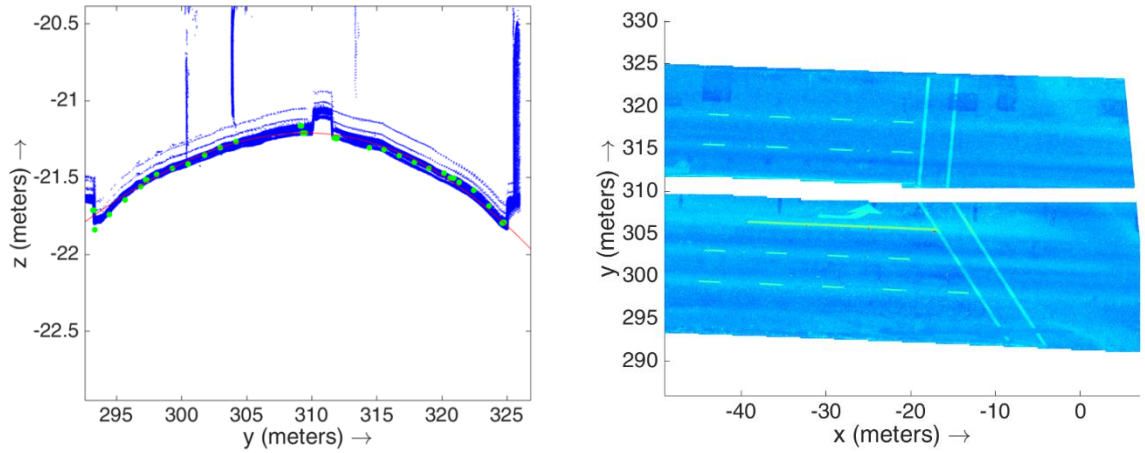
Extraction of the Road Surface per Section

The reduced point cloud $\bar{P}_i^j$ found in the previous step has removed points beyond the road edges, but will yet contain reflections from vehicles, pedestrians, signs or other objects that are within the roadway edges. The objective of this processing step is to extract a further reduced point cloud

$$S_i^j = \{x \in \bar{P}_i^j \mid x \text{ is point on the road surface}\}.$$

The set S_i^j is built up by considering small rectangular cross-sections of $\bar{P}_i^j$ in a manner similar to that illustrated in Fig. 4a. A sample elevation versus cross-sectional position distribution of points is shown in Fig. 6a. The road surface is within the wide horizontal band of points along the bottom. The vertical strips of points represent reflections from entities on top of the roadway (e.g., vehicles).

For each such rectangular slice, the platform trajectory points are first projected down onto the road using the known platform calibration parameters. These points are represented by green dots superimposed on the distribution in Fig 6a. Additional points are estimated between these projected points using the mode of the elevation distribute as a function of cross-sectional position *y*. Finally, a 6-degree polynomial is fit to this set of green dots. Point cloud elements within 20 cm of this curve-fit are extracted as the road surface points (S_i^j). An example surface is shown as a top-down view in Fig. 6b.



(a) (b)

Figure 6: (a) Blue dots are the elements of the point cloud $\bar{P}_i^j$. The green dots a points extracted either by projections of the sensor platform trajectory or the point cloud elevation distribution. (b) The extracted road surface S_i^j.

C. Map 3D points to 2D image

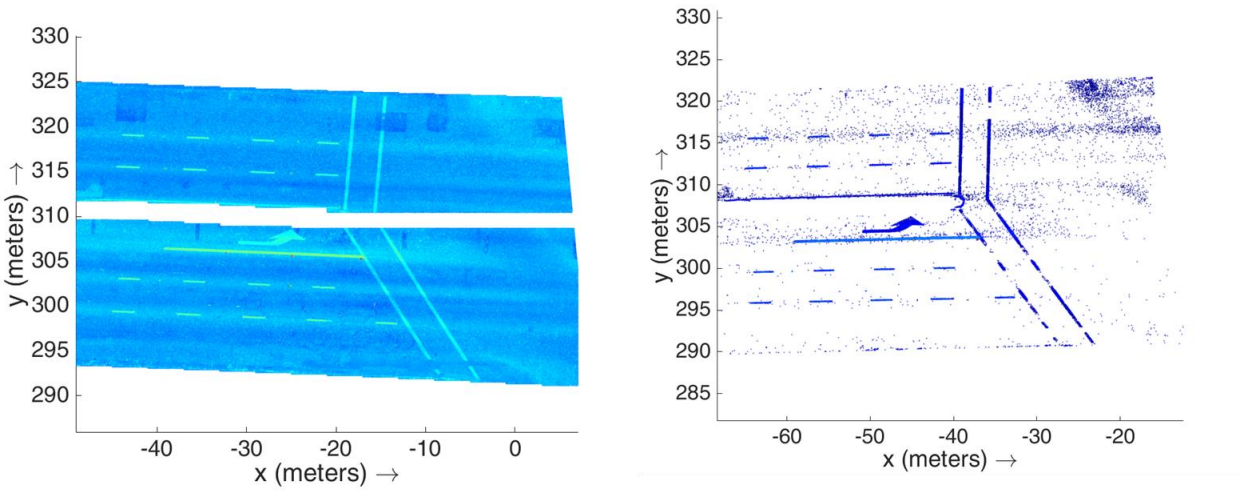
The purpose of this step is to convert each road surface point cloud S_i^j into either one or two raster images, one for each ingress or egress roadway section separated by a median. Many image processing tools are available to extract features from such images. This conversion is completed in the following steps.

First, for roadway segments where Step B detects a median, the median curve is used to divide S_i^j into two separate branches S_i^{jk} for k=1, 2. If a road segment does not have a median, then the segment is treated as one branch with k=1.

Second, for each road surface segment branch S_i^{jk}, points with low reflectivity are removed from S_i^{jk} to generate a new point cloud subset

$$\bar{S}_i^{jk} = \{x \in S_i^{jk} \mid I(\mathrm{x}) > \tau\}$$

where τ is a user defined intensity threshold. A segment before (i.e., S_i^{jk}) and after (i.e., $\bar{S}_i^{jk}$) intensity thresholding is shown in Figs. 7a and 7b.



(a) (b)

Figure 7: (a) Two branches S_i^{jk} for k=1, 2 of an extracted road surface segment. (b) Two branches $\bar{S}_i^{jk}$ for k=1, 2 of an extracted road surface segment after intensity thresholding.

A 2D raster image $\bar{R}_i^{jk}$ is generated from each point cloud $\bar{S}_i^{jk}$. The pixel size of the raster image in meters is a trade-off between computational complexity and feature position estimation accuracy. For the feature extraction approach described herein, each raster pixel is $3\ cm \times 3\ cm$. Therefore, each raster pixel maps to a 3 cm square cell in the XY region. The value of the pixel for each cell depends on the elements of $\bar{S}_i^{jk}$ that map to the cell. When no elements of $\bar{S}_i^{jk}$ map to a cell, the pixel value is zero. Otherwise, the pixel value is the average value of the intensities of the elements of $\bar{S}_i^{jk}$ that map to the cell. The raster image corresponding to Fig. 7 is shown in Fig. 8.

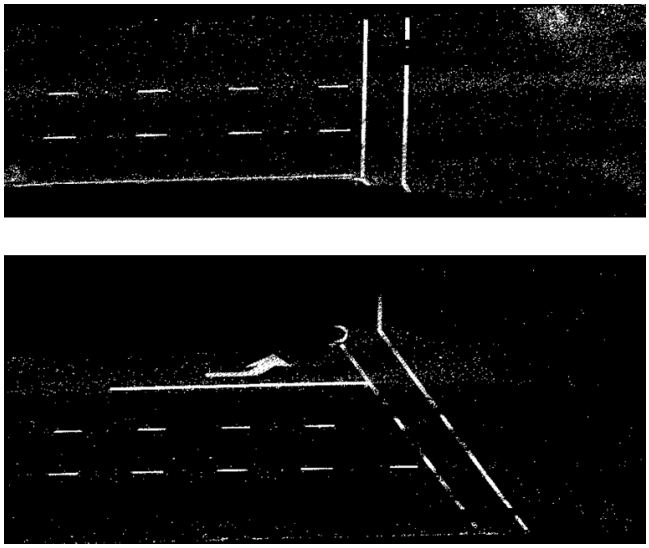


Figure 8: The two raster images generated from the intensity thresholded point cloud $\bar{S}_i^{jk}$

The world coordinates of the corner points of the raster image are saved as metadata so each pixel can be converted back to UTM coordinates later for feature mapping. This is referred to herein as a calibrated raster image.

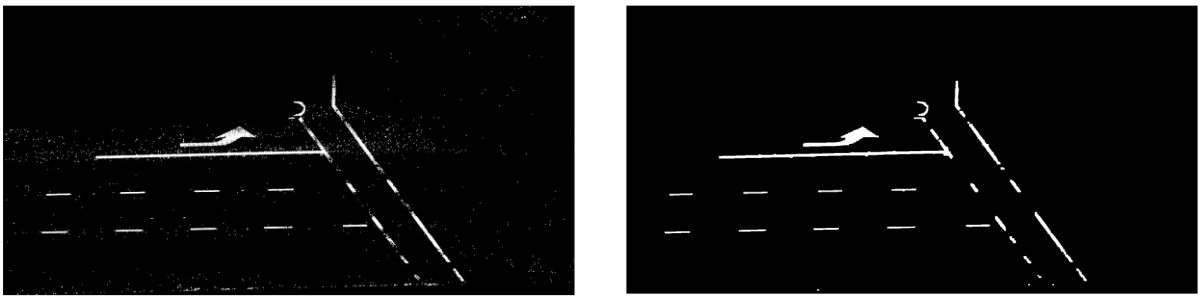
D. Roadway Feature Extraction

This section describes the methods used to extract the J2735 map message data from each calibrated raster image $\bar{R}_i^{jk}$. J2735 map message includes lane centerline nodes, lane widths and stop bar locations.

The process includes the following steps: image processing to improve the detectability of the desired features; detection of stop bars; detection of lane dividers; definition of lane centerlines and lane widths.

Image enhancement

Noise and unwanted artifacts can degrade performance of the feature recognition algorithms. Lane edge and stop bar markings are each composed of rectangular elements with standard dimensions [5]. In this step, these known characteristics are used to design templates for image processing methods (erosion and dilation) that will preserve and clarify matching features while deemphasizing elements of the image that do not match the known characteristics. Fig. 9 shows an image example before and after the image enhancement process.



(a) (b)

Figure 9: Image enhancement process. a) Original image. b) Processed image. Artifacts smaller than a certain size are removed from the image, high intensity regions are closed, linear features are emphasized.

Stop Bar Detection

Stop bars are identified using the Hough transform to detect lines, keeping on those lines with sufficient length and orientation (approximately) orthogonal to the trajectory direction. Because more than one line may fit these criteria when a pedestrian crosswalk exists, the detected line farthest from the intersection is processed. Its line is estimated and passed to the next steps for

further processing. An example result of stop bar detection is shown in Fig. 10. The stop bar line is shown in red. Other detected lines of sufficient length that are oriented (approximately) orthogonal to the lane direction are shown in green.

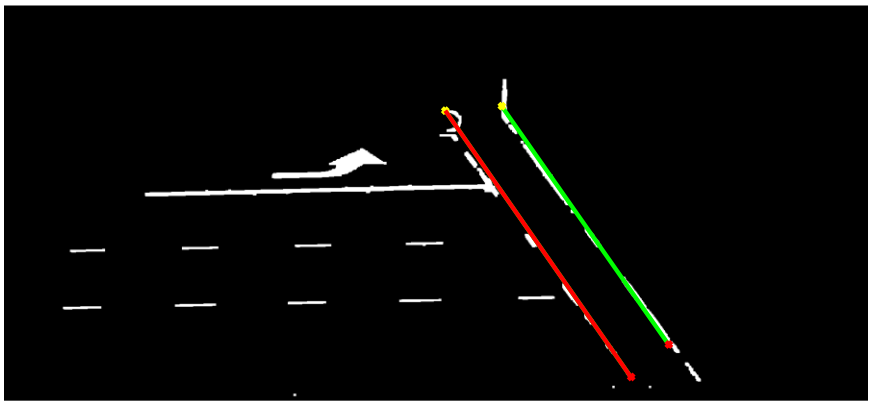


Figure 10: Example of detected stop bar candidates.

Lane Divider Detection

The goal of this step is to divide each branch into lanes. The inputs to this process are the calibrated raster image $\bar{R}_i^{jk}$ and the curves defining the road and median edges.

The Hough transform of $\bar{R}_i^{jk}$ from the previous step also returns lines parallel to the direction of traffic flow, which are candidates for lane dividers. Good candidates will have sufficient length, sufficient separation from other lane dividers, and match one of the templates defined in the subsequent sentences. Left turn lanes are frequently set-off by a solid line to their right. On multi-lane roads, lanes are separated by dashed lines where dashes are 2-3 m long and 4-6 m apart (approximately) [5].

The software uses a decision tree to process the lines returned by the Hough transform and road and median edge curves. Depending on the branch width as determined from the median, lane dividers, and road edge curves, the algorithm labels the branch as a single lane or multi-lane. The algorithm first checks for solid lines, as used to demarcate turning lanes. The criteria for accepting solid lines as lane dividers include distance of the line from the branch edges, distance of this line from already accepted lane edges, line direction and length. After finding solid lines that demarcate lane edges, the remaining road width is computed, as the basis for a decision concerning the maximum number of remaining lanes. Given this estimate of the maximum number of remaining lanes, the algorithm begins searching through the remaining dashed line checking direction, dash length and spacing, and separation between previously accepted lane edges and median and road edges. The process concludes either when the remaining maximum number of remaining lanes is less than one or when there are no remaining suitable lines to serve as lane dividers. In some cases, for cross-streets, the branch is too narrow to support multiple lanes and has not lane markings. In this case, the lane edges are defined to be the road and or median edges. In some cases, such as cross-streets, the branch has two lanes with no or undetectable markings, yet has width sufficient for two lanes. In this case, the algorithm defines the curve half way between the branch edges as the lane separator and processes each lane as a single lane.

Once the lane edges are determined, the lane centerline is computed by dropping nodes midway between the two lane edges. Nodes are separated by a distance D = 6 m (200 pixels apart). The first node of the centerline is defined to be at the midpoint of the line segment between the intersections of the lane edges and the stop bar. Lane width is calculated as the inner product of the line connecting the two points defined by the two lane edges intersecting with the stop bar and the unit vector that is perpendicular to the lane edge.

Fig. 11 illustrates the output of the feature extraction algorithm. The input image $\bar{R}_i^{jk}$ is the background. The input road edges are shown as green dotted lines. The extracted stop bar by a solid yellow line and extracted nodes by blue circles.

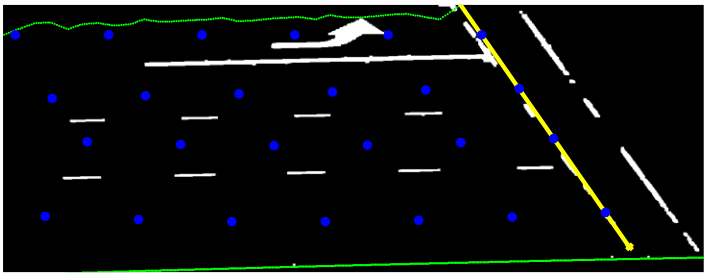


Figure 11: Output of the feature extraction algorithm.

Challenges

When the road markings are severely faded, the road has no markings, or the input image is very noisy, the algorithm cannot succeed in its automated processing. In such cases, it generates a warning and marks the segment for human oversight or intervention. Another issue arises when only one stop bar is detected, in which case the algorithm proceeds with the detected stop bar and generates a warning for further human inspection.

Further examples of when such warnings occur are presented in Section IV.

E. J2735 Intersection Feature Map Estimation

After the lane centerline nodes and lane widths are estimated, the image coordinates of these nodes are converted back to UTM coordinates using the metadata saved during raster generation. The intensity thresholded road surface segments are first rotated back to their original orientation and then joined together to get an intensity thresholded point cloud $\bar{S}_i$ for the whole intersection. These nodes are then superimposed R_i for analytic comparison.

Top-down view of an intensity thresholded point cloud $\bar{S}_i$ with the extracted J2735 nodes superimposed is shown in Fig. 12. According to the specification of the J2735, the first node of each lane starts from the position of the stop bar. In addition, the nodes coordinates are converted from UTM to lat-long to superimpose on Google Earth to make a second comparison. The Google Earth image is shown in Fig. 13.

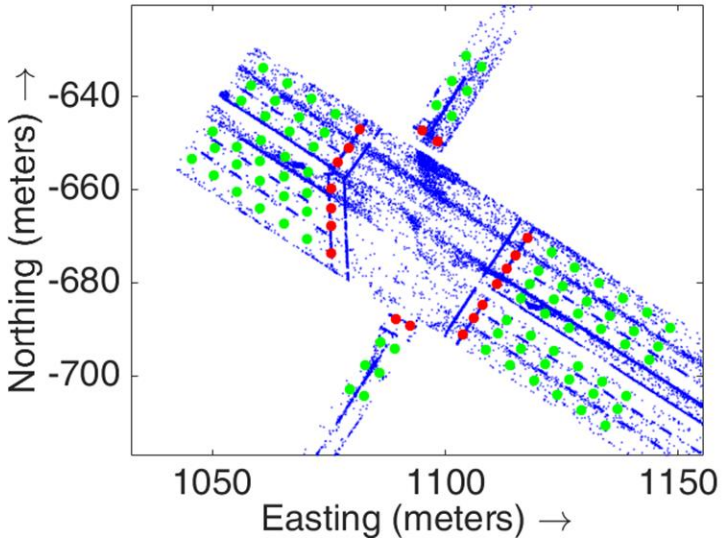


Figure 12: Extracted nodes superimposed on the point cloud.

Figure 13: Extracted nodes overlaid on google earth.

IV. EXPERIMENTAL RESULTS:

This section presents the results of applying the method of Section 3 to all eleven intersections 11 intersections along Palo Alto testbed. Each intersection map is output in the SAE J2735 format.

The J2735 map message includes:

- coordinates for the intersection center
- the number of lanes
- a data structure for each lane including
    - a lane identifier
    - a sequence of nodes defining the lane centerline
    - the lane width
    - a point indicating the start of the centerline at the stop bar
    - lane attributes such as whether the lane is for ingress or egress.

This project has focused on standard intersections, where essentially orthogonal and straight streets intersect. The MTLS based approach assumes that the lane markings are clearly painted.

The automated mapping results are illustrated by drawing the J2735 map centerline nodes overlaid on both the point cloud and a Google Earth image of the intersection. The point cloud data acquisition process includes surveyed control points such that the point cloud has a known accuracy at the centimeter level [1]. The J2735 overlaid on the point cloud allows evaluation of the absolute accuracy of the J2735 intersection map. The J2735 overlaid on the Google Earth image gives a visually interpretable result. The Google Earth imagery is only accurate to 2-5 meters [2, 3, 4]. Therefore, in some instances the entire J2735 overlay is offset relative to the Google Earth image while aligning well with the point cloud. Such images clearly show the lesser accuracy of the Google Earth imagery relative to the point cloud. The visual overlay of the J2735 onto the Google Earth imagery is useful for detecting relative errors between portions of the J2735, for example a stop bar at the wrong location.

Most of the intersections process completely with correct outputs. In cases where the road markings are severely faded, the road has no markings, or the input image is very noisy the algorithm generates a warning and marks the segment for human processing. Another issue arises when only one stop bar is detected, in which case the algorithm proceeds with the detected stop bar and generates a warning for further human inspection. In case no stop bar is detected and the algorithm generates a warning. For these cases, in our results, a stop bar is added manually so the algorithm can perform the rest of the tasks. The discussion will point out any known processing issues or errors.

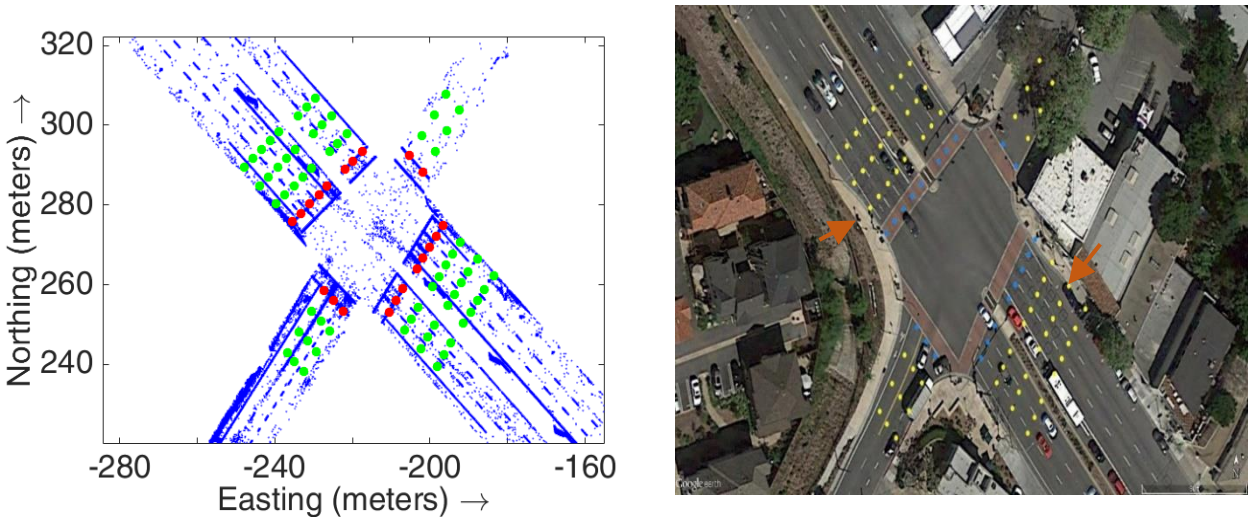
Intersection 01:

Intersection 01 is a standard cross shaped intersection and the stop bars have formed regular rectangular shape.

Fig. 14a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has three green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 14b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- All lanes were extracted. No lanes were missed.
- From the point cloud overlay, all lane centerlines and stop bar locations are correct.
- The Google Earth image shows that the image and J2735 are shifted relative to each other, clearly showing the relative inaccuracy of the Google Earth image relative to the more precise point cloud [2, 3, 4].
- Fig. 14b contains two orange arrows. These markers indicate two non-standard lanes that have been detected and mapped as traffic lanes. The point cloud image shows that these two lanes satisfy the current definitions of lanes, so they are included in the J2735 output. In future work, either their narrowness, narrowing as they approach the intersection, or position near the road edge can be used as triggers either to eliminate them from the J2735 output or to request human oversight.



(a) (b)

Figure 14: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 1. (b) Extracted J2735 nodes overlaid on Google earth image.

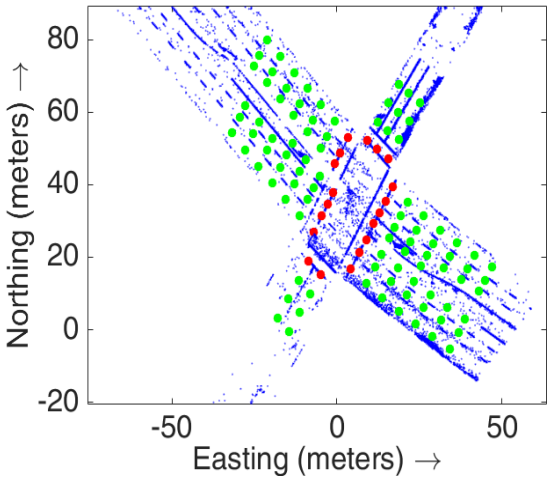
Intersection 02:

This is an almost regular shaped intersection. The main thoroughfare is slightly curved and the intersecting streets not quite orthogonal, yet the approach worked well.

Fig. 15a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 15b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- All lanes were extracted. No lanes were missed.
- From the point cloud overlay, all lane centerline locations are correct.
- The Google Earth image shows that the image and J2735 are shifted relative to each other, such that the J2735 centerlines sit on the paint stripes. This clearly showing the relative inaccuracy of the Google Earth image relative to the point cloud [2, 3, 4].
- Fig. 15b has two orange arrows. These markers point to two J2735 stop bar locations which have been detected and mapped at the wrong line of the pedestrian cross walk. For the arrow at the lower left, the error is due to worn lane markings. In future work, this issue could be detected by the fact that stop bar location discontinuously changes from the ingress and egress portions of that road segment.



(a)

(b)

Figure 15: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 2. (b) Extracted J2735 nodes overlaid on Google earth image.

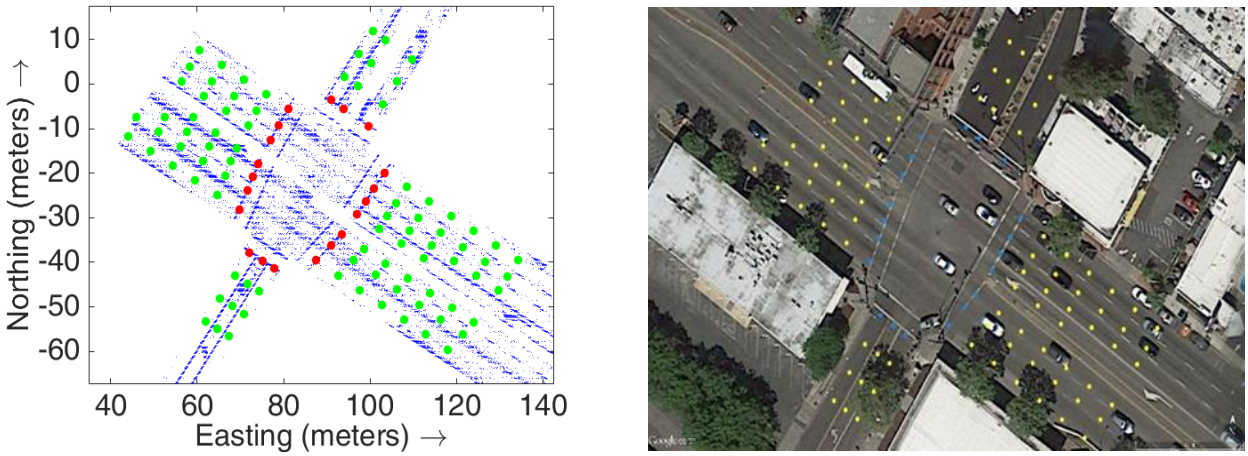
Intersection 03:

Intersection 03 is a regular rectangular shaped intersection.

Fig. 16a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 16b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- All lanes were extracted. No lanes were missed.
- From the point cloud overlay, all lane centerline locations are correct.
- The Google Earth image and J2735 are well aligned.
- Note in the Google Earth image that the upper half of the cross street appears to be newly paved without any lane striping. The point cloud image indicates high intensity lines, so we suspect that the point cloud was acquired prior to the street being repaved. The mapping process worked well. This bullet clarifies why no lane striping is visible on this road segment.



(a) (b)

Figure 16: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 3. (b) Extracted J2735 nodes overlaid on Google earth image.

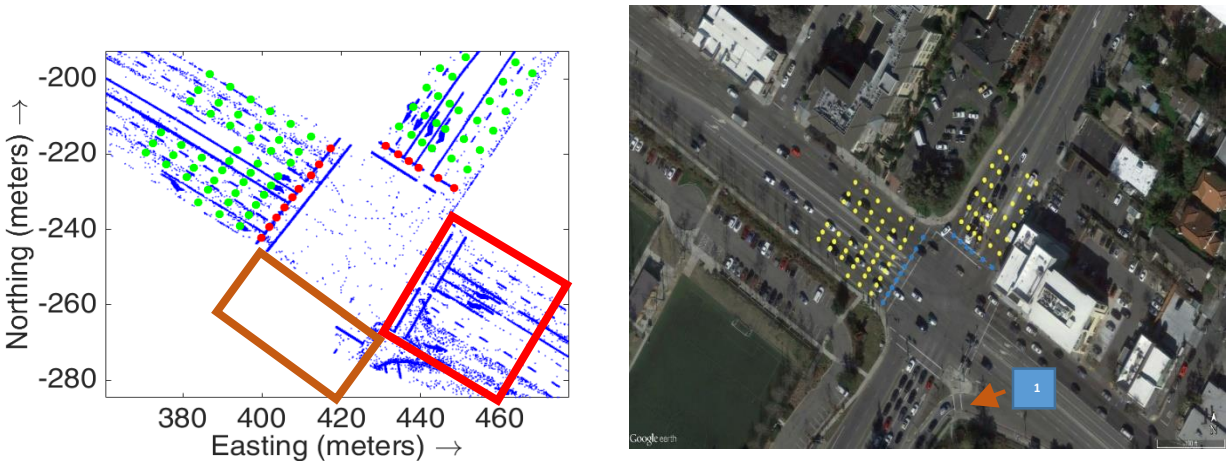
Intersection 04:

Intersection 04 is an irregularly shaped intersection that has one side road with a dedicated right-turn lane with an extra stop bar (see box 1 in Fig. 17b), wide side roads with medians, and faded paintings.

Fig. 17a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 17b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details the following items:

- The region indicated by the brown and red boxes could not be processed due to the non-standard road geometry and faded lane striping.
- For other road segments the detection of centerline nodes of every lane in all the ingress and egress branches and the position of the stop bars are estimated correctly.



(a) (b)

Figure 17: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 4. (b) Extracted J2735 nodes overlaid on Google earth image.

Intersection 05:

Intersection 05 is a cross-shaped intersection with special roadway features. See Figs. 18b and 19b. There are two T-shaped intersections in close proximity. Due to this special structure we decomposed the intersection into intersections 5a and 5b, which are each much closer to being standard. Each is processed separately.

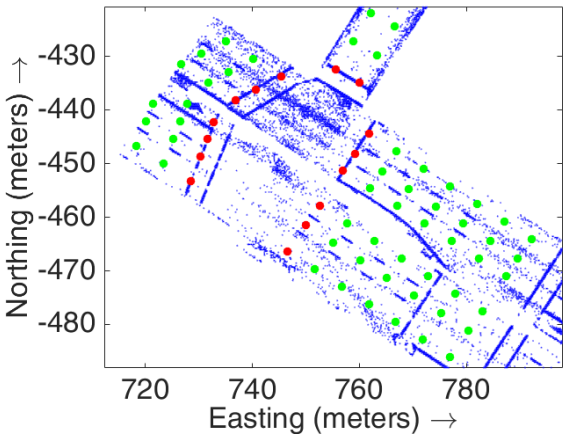
Intersection 5(a): It is a T-shaped intersection with three road segments.

Fig. 18a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 18b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- There are no stop bar markings on the pavement at the egress section marked as item 1 in Fig. 18b. Given this situation, this system generates a warning to the human operator for inspection. To proceed, the human manually extends the stop bar line from the parallel ingress lane.
- With the manually inserted stop bar, all roadway feature nodes (i.e., centerline and stop bar nodes in every ingress and egress section) have been detected and mapped correctly.
- Note that the stop bar of the left segment of the main thoroughfare (i.e., item 2 Fig. 18b) is not a straight line; nonetheless, all the stop bar positions have been detected correctly.

In future work, the insertion of stop bars could be automated, for the cases where the pavement is unmarked. It is still recommended that the human be alerted, for verification, since this extrapolation of information may not be applicable in non-standard situations. The item 2 in Fig. 18b is an example of a non-standard situation where extending the stop bar between roadway sections would be inappropriate.



(a)

(b)

Figure 18: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 5a. (b) Extracted J2735 nodes overlaid on Google earth image.

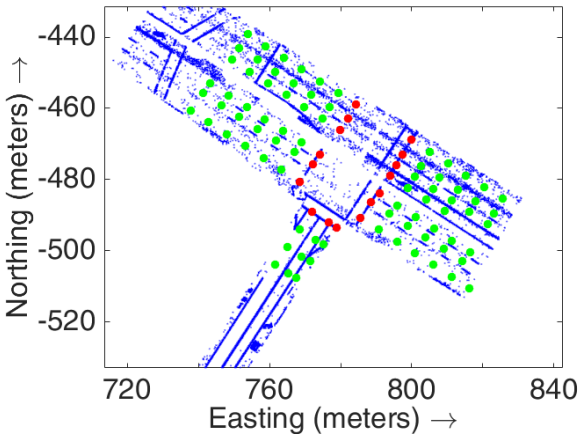
Intersection 5(b): It is a non-standard T-shaped intersection with three road segments. The side road has two dedicated lanes for ingress and egress with extra stop bars (marked in Fig. 19(b) as items 2 and 3). Along with that structural ambiguity there is an egress segment of the main thoroughfare without a painted stop bar (see item 1 in Fig. 19b).

Fig. 19a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 19b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

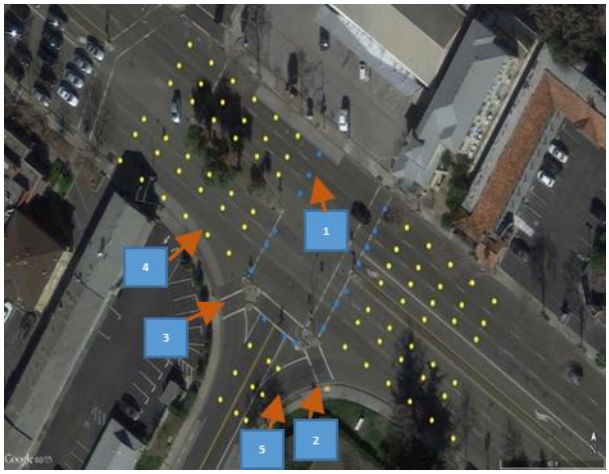
Visually inspecting the two images details several items:

- At the egress to the main thoroughfare indicated as item 1, there is no painted stop bar on the road surface. In this case, the program alerts the user, who manually extends the line from the parallel ingress lane so that the stop bar definition can be completed.
- Items 2-5 all relate to the non-standard structure of the intersection. The current software is not written to accommodate the curved lane marked by items 4-5 nor the extra stop bars indicated by items 2-3.
- Other than the exceptions note above, the stop bars and centerline nodes of all lanes of both ingress and egress have been identified correctly.

In future work, the insertion of stop bars could be automated, for the cases where the pavement is unmarked. It is still recommended that the human be alerted, for verification, since this extrapolation of information may not be applicable in non-standard situations. It is also expected that the curved right-hand turn lanes can be incorporated into the automated process.



(a)



(b)

Figure 19: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 5b. (b) Extracted J2735 nodes overlaid on Google earth image.

Intersection 06:

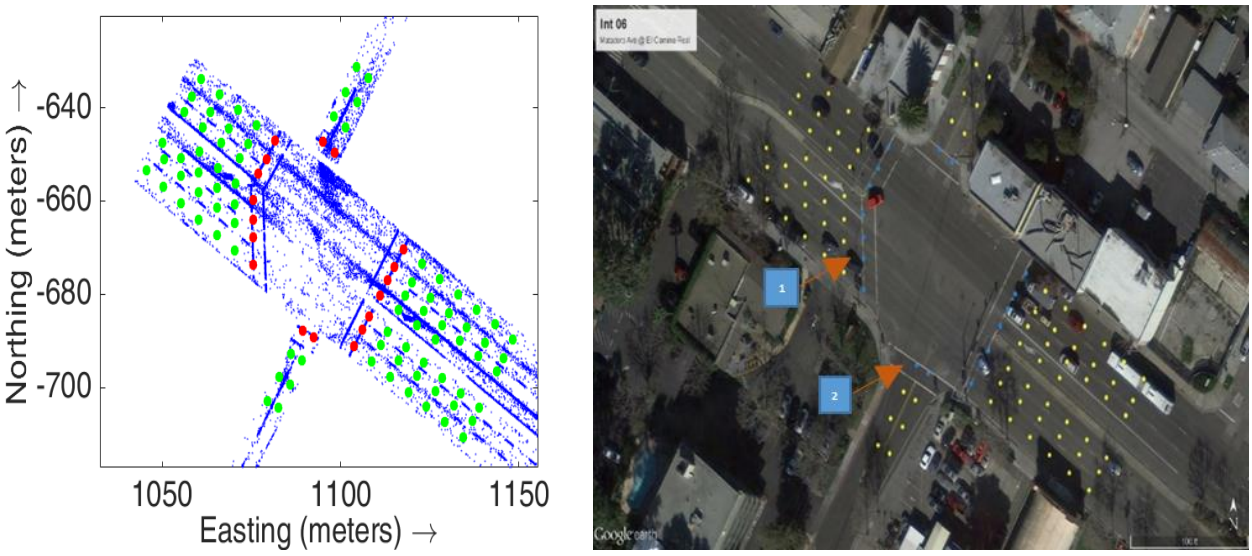
It is a non-standard cross shaped intersection. Note that the cross street sections are off-set laterally relative to each other and the stop bar indicated as item 1 is not orthogonal to the lane centerline and not a straight-line across the entire road.

Fig. 20a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 20b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- Despite of its nonstandard aspects all the lanes of every egress and ingress section have been correctly extracted.
- The lane striping is barely visible on the google image.
- All the centerline nodes of every lane have been identified correctly.
- All the stop bars are positioned correctly except for one side road segment part (indicated in Fig. 20b as item 2).

The misplaced stop bar is expected to be fixable in future efforts.



(a) (b)

Figure 20: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 6. (b) Extracted J2735 nodes overlaid on Google earth image.

Intersection 07:

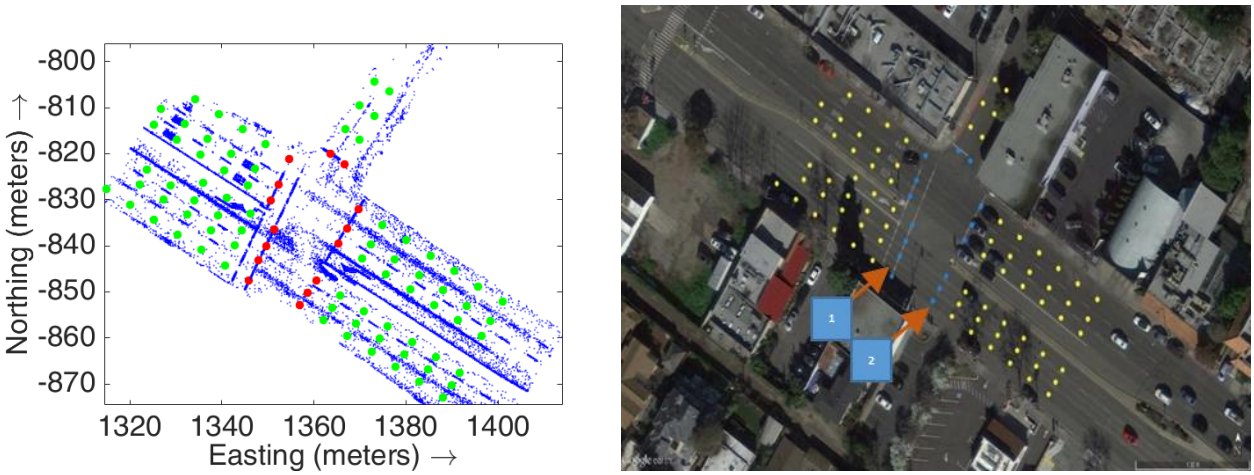
This is a standard T-shaped intersection with three road segments.

Fig. 21a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 21b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- All lanes were extracted. No lanes were missed.
- All centerlines are correct.
- Box 1 shows misplacement of stop bar at an ingress branch of the main road.
- Box 2 indicates a main thoroughfare eggress where there is no painted stop bar. In this case, the program alerts the user, who manually extends the line from the parallel ingress lane so that the stop bar definition can be completed.

The issues pointed to by boxes 1 and 2 could both be addressed in future work.



(a) (b)

Figure 21: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 7. (b) Extracted J2735 nodes overlaid on Google earth image.

Intersection 08:

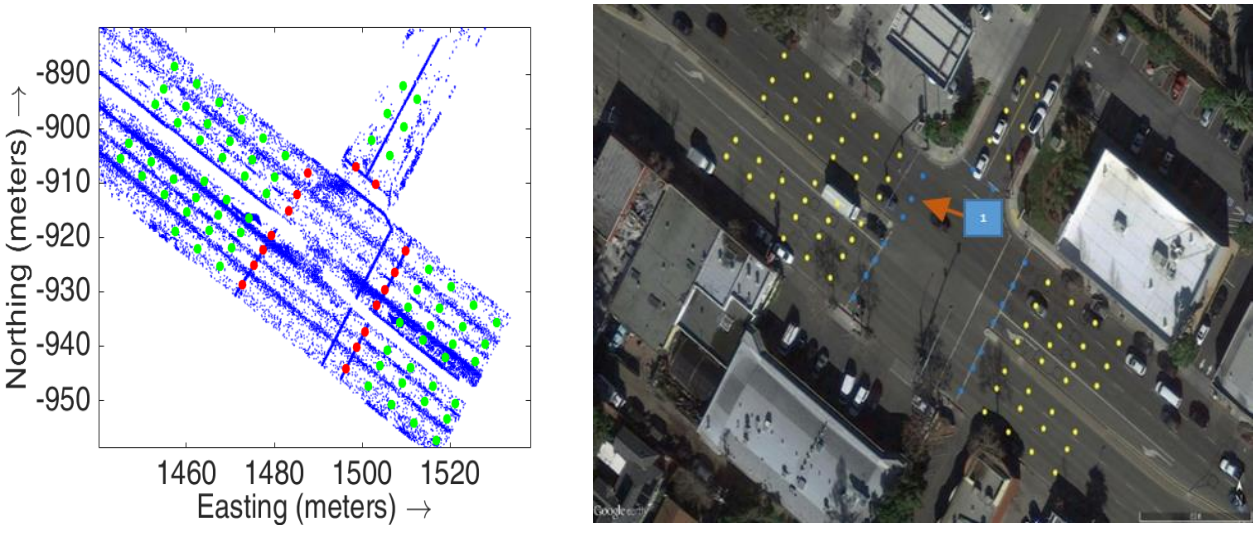
This is a standard T-shaped intersection.

Fig. 22a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 22b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- The number of lanes, lane centerlines, and lane widths in every ingress and egress branch are extracted correctly. No lanes were missed.
- Box 1 points to a main thoroughfare egress section for which there is no stop bar painted on the road surface. This is clear in the point cloud image. The Google Earth image appears to show a stop bar; however, more careful observation shows that this is an overhead traffic signal and its shadow on the road surface. The absence of the painted stop bar on the road surface causes the automatic processing to end with an alert to the user. The user then manually extends the line of the stop bar from the parallel ingress road section to allow definition of the J2735 stop bar for each egress lane.
- With human interaction for the stop bar at the single egress section, correct extraction and placement of all stop bars and lane centerline nodes has been achieved.

In future work, the insertion of stop bars could be automated, for the cases where the pavement is unmarked. It is still recommended that the human be alerted, for verification, since this extrapolation of information may not be applicable in non-standard situations.



(a) (b)

Figure 22: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 8. (b) Extracted J2735 nodes overlaid on Google earth image.

Intersection 09:

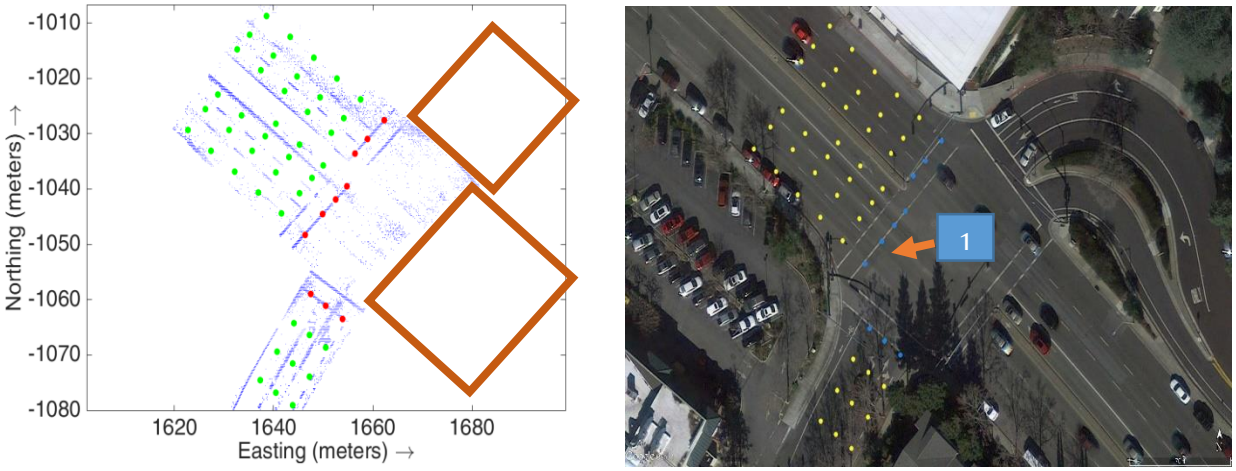
Intersection 09 is an irregularly shaped intersection with one curved and two merging road segment.

Fig. 23a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 23b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- The two brown boxes mark two road segments that cannot be processed due to irregular geometric structure and light paint markings. The irregular structure prevents segmentation, therefore, the remainder of the feature extraction process cannot complete for those segments of the intersection.
- The experiment operated successfully on all the regular shaped road segments with all centerlines mapped identified correctly.
- The orange arrow marked box 1 indicates the misplacement of the stop bar of the ingress branch of the main road segment.

Improvement of the segmentation process for non-standard intersections is a topic for future work.



(a) (b)

Figure 23: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i =9. (b) Extracted J2735 nodes overlaid on Google earth image.

Intersection 10:

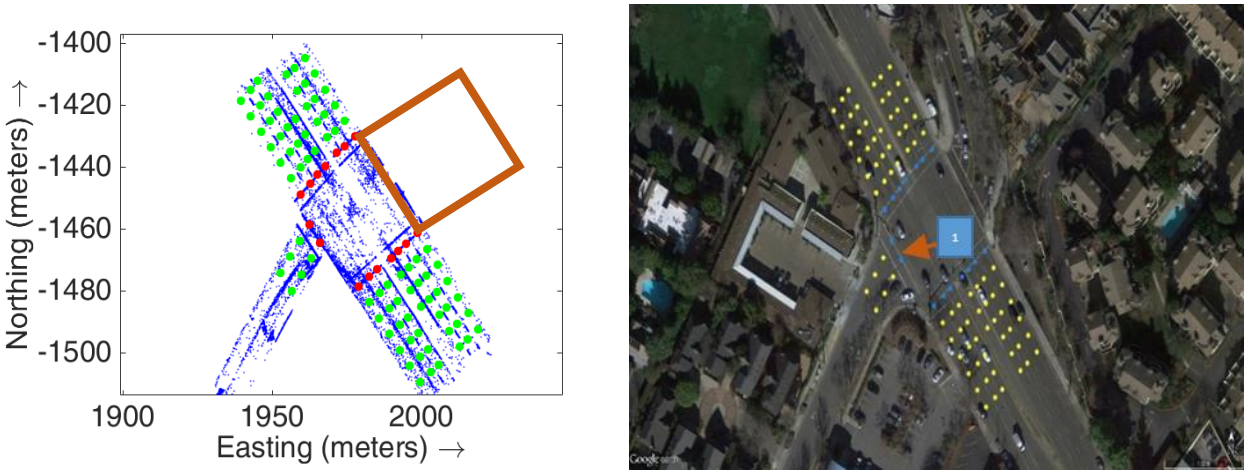
Intersection 10 is a cross intersection with an irregularly shaped side road segment. The non-standard side road is identified with a brown box of Fig. 24a.

Fig. 24a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 24b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details several items:

- All lanes along the main thoroughfare were extracted correctly. No lanes were missed.
- The lanes of the lower portion of the cross street are extracted correctly, but the wrong stop bar is extracted. This is marked by the orange arrow.
- For the top portion of the cross street, the J2735 map cannot be extracted due to failure at the segmentation step, caused by the non-standard intersection geometry.

Improvement of the segmentation process for non-standard intersections is a topic for future work.



(a) (b)

Figure 24: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 10. (b) Extracted J2735 nodes overlaid on Google earth image.

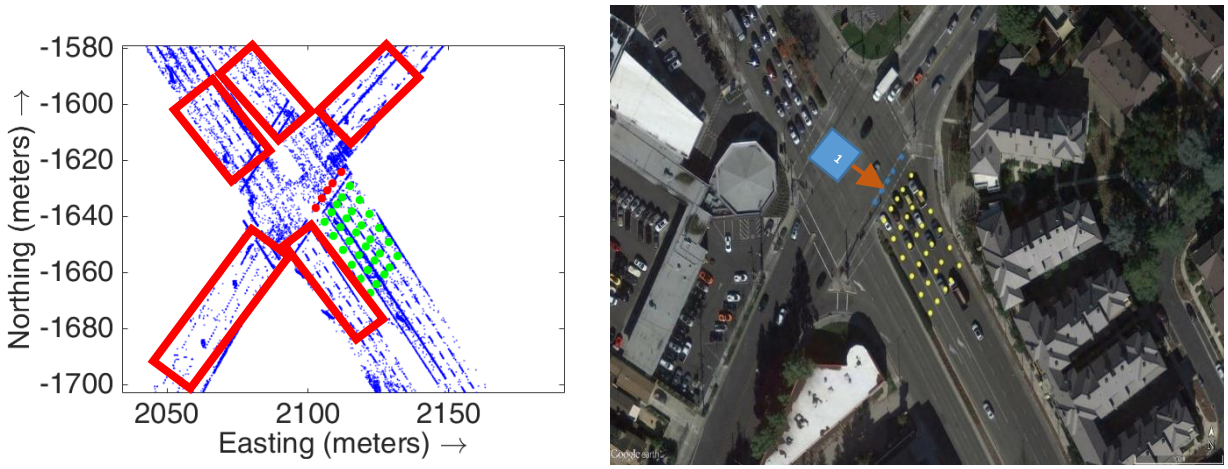
Intersection 11:

Intersection 11 is an almost standard cross shaped intersection.

Fig. 25a shows the extracted J2735 overlaid on the point cloud. Each lane centerline starts from a red dot and then has several green dots. The red dot should be on the stop bar. The sequence of green dots should fall on the lane centerline. Fig. 25b shows the same J2735 overlaid on the Google Earth image. In this case, each centerline has a blue dot marking the stop bar location and a sequence of yellow dots marking the lane centerline.

Visually inspecting the two images details that:

- Both egress and one ingress portion of the thoroughfare and both sections of the cross street could not be processed due to faded lane stripes. This is indicated by the red rectangles in Fig. 25a.
- One ingress portion of the main thoroughfare has all lanes extracted correctly, but marked the wrong stop bar as indicated by the orange arrow in Fig. 25b.



(a) (b)

Figure 25: (a) Extracted J2735 nodes superimposed on the intensity thresholded point cloud $\bar{S}_i$ for i = 11. (b) Extracted J2735 nodes overlaid on Google earth image.

BEST PRACTICES FOR MTLS BASED AUTOMATED EXTRACTION OF ROADWAY RELEVANT FEATURES

Equipment

- **LIDAR**: Active imaging sensor for roadway feature mapping. It should have a field-of-view sufficient to simultaneously capture inner and outer roadway edges when vehicle is driven in the center lane.
- **High-resolution 360 degree camera**: Enables photogrammetry to assist visual interpretation
- **Two-frequency carrier phase GNSS receiver**: Enables high-precision (centimeter accuracy) georectification (estimation of position and attitude) of the LIDAR and camera data.
- **Inertial Measurement Unit (IMU)**: Enables georectification to be implemented at the high LIDAR (15-20 Hz) and camera rates (30 Hz), even though the GNSS makes precise, independent position and attitude measurements at only at much lower rates.
- **Differential GPS corrections:** Necessary to achieve the desired centimeter to decimeter map position accuracy. One means of implementation is by a proximal (< 10 km) base station. Another is by network computation of corrections. In either case, corrections may be communicated by RF means in real-time or combined with GPS in post-processing.
- **Power**: Sufficient support power to run all equipment.
- **Data-storage**: Sufficient to save the raw, time-stamped, sensor data over the time required to complete the data collection exercise.

Data collection procedure: The goal is to ensure a high-density of LIDAR reflections from roadway relevant features, to facilitate reliable automated extraction and mapping of those features.

- **Means to obtain a high-density of reflections:**
    - **The mapping vehicle should traverse each entry and exit point for each intersection a few times.** These traverses should travel along different lanes. It is not necessary to traverse each lane, but one traverse of the inner-most and outer-most lanes is useful, due to the different perspectives that they provide. The tradeoff is that more traverses gives a higher density of LIDAR reflections, different perspectives, and different obstructions; however, each traverse adds cost due to time, fuel, and data storage.
    - **Mapping vehicle should travel at reduced speeds (20-30mph)**, when appropriate. The tradeoff is slower speeds provide gives a higher density of LIDAR reflections; however, adds cost due to time and data storage requirements.
    - **Including multiple LIDAR's**, directly improves the density of reflections, per mile driven.
    - **Ensure that the LIDAR's are mounted such that they illuminate the regions expected to contain the roadway features of interest.**

- **Means to facilitate automatic extraction of features:**
    - In post-processing, the GNSS and IMU data will be used to estimate the platform trajectory and to geo-rectify the LIDAR and camera data. **Both this geo-rectified data and the platform trajectory should be saved for the feature extraction processing**. The platform trajectory provides useful data: number of intersection entry and exit points and direction of travel along each.
- **Means to avoid occlusion of roadway features:**
    - Fore and aft rolling closure and/or traffic break minimizes artifacts from vehicle intrusions.
    - Performing multiple traverses provides multiple opportunities to receive reflections from each roadway feature. The traverses occurring at different time improves the chances that the occluding object is no longer occluding the same features.
- **Means to confirm accuracy:**
    - Occasional control points (known locations) consisting of highly reflective material allows both instrument calibration and confirmation of the accuracy of mapped features.

Data post processing: After determining the platform trajectory and georectified LIDAR point cloud and camera photolog. The basic issues are:

- Retain the raw sensor data as long as possible.
- Retain the point cloud data for the entire roadway region of interest.
- Utilize control points to assist with calibration, validation, and merging of data sets.
- For georectification and feature mapping use a consistent Earth Center Earth Fixed (ECEF) reference frame in addition to other formats (state plane; latitude, longitude, elevation)

In the mapping of intersections, the following have also proved useful:

- Define a reduced point cloud for each intersection, for example all points within a radius R for the defined origin of the intersection.
- Within this region, process the sensor platform trajectory to determine the number of approaches and exits to the intersection.
- From the intersection point cloud, extract a subset that represents one approach or exit.
- Filter and remove data from interfering objects (e.g., vehicles, trees, people). This can be accomplished for example by extracting from the point cloud only those points that form a smooth two-dimensional surface, discarding any points too far from that surface.
- Perform feature extraction as necessary for the application of interest.
- Store the extracted lane markings, road edges, and other features of interest with precision (perhaps 10 times) greater than the desired accuracy of the map.

CONCLUSIONS

Connected Vehicle applications require roadway feature digital maps stored and communicated using uniform methods for representation and reference. These maps must conform to well understood accuracy specifications to ensure the safety of life and infrastructure. Due to the scope of the mapping requirements, when considered at the national and global scales, Automated and data driven mapping approaches must be considered.

Numerous methods exist to acquire and record data indicative of geospatial objects and terrain. The preferred existing acquisition methods for roadway mapping utilize GPS integrated LIDAR sensing to generate a three dimensional georectified point cloud data set. At present, the state of the art approaches to feature extraction are human intensive. When data acquisition is conducted with multiple integrated sensors at a high frequency and national (or global) scale, the resulting dataset is sufficiently large to require automated and application specific data processing. This task report has presented an automated feature extraction approach and demonstrated its use on eleven intersections. The resulting processed dataset presented herein provides a meaningful representation of roadway features of interest for intersections, while excluding irrelevant data.

The roadway feature extraction process presented detailed the following primary steps:

- Preprocessing to extract the georectified point cloud and associated trajectory portions of interest for given intersections.
- Identification and extraction of the road surface point cloud, road edge curves, and median edge curves.
- Conversion of the intersection road surface point cloud to an image to enable additional image processing.
- Image-based roadway feature extraction.
- Translation to a J2735 feature map for intersections of interest.

This methodology for feature extraction and map representation has been presented with a detailed explanation of data processing and integration. This detailed approach allows for future feature extraction of relevant roadway features in a connected vehicle environment.

The analysis of the algorithm performance shows that it works very well for “standard” intersections. However, additional work is required to achieve functionality for non-standard intersections. Also, on the continuum between manual and fully automated, current algorithm is somewhere in the middle, human-assisted. When the algorithm detects an anomalous situation, it stops to alert the human, this is considered good practice as the generated maps have implications for human safety. Also, the algorithm includes parameters such as intensity thresholds that currently require human adjustment.

FUTURE WORK

The details of the feature extraction methodology and its evaluation on eleven intersections demonstrate the complexity of representing relevant roadway features in a standard usable format within a connected vehicle environment. The algorithm worked well in spite of these complexities,

making a significant advance from manual feature extraction toward automated data processing for roadway relevant feature extraction. Following are a few avenues fruitful for future efforts:

- Automated feature extraction advancements are developed best within an application testbed environment. UCR is interested in collaborating with government agencies, academic groups, or commercial entities in testbed applications and demonstration projects that require roadway relevant maps and allow the opportunity for advancements within the automated mapping field.
- The accuracy of roadway feature maps is built upon the foundation of the platform trajectory estimation step in the georectification process. Methods to enhance the reliability and integrity of high accuracy, especially in GNSS challenged environments is necessary.
- The approach presented herein was limited in scope to intersections and specific features. Broadening this is important.

Evolution of geospatial data acquisition will yield ever increasing accurate and precise topology data for the roadway environment. The necessity to identify and extract relevant roadway features will become significantly more valuable. The methodology presented herein focused on roadway attributes associated with intersection relevant road surface features. It should be viewed as a glimpse of what is possible in the future as the technology develops. Present and future sensing and computing resources are not the limiting factors. This field is in its infancy. Feature extraction algorithms, mapping standards, commercial approaches, and potential connected vehicle applications are still in their formative and demonstration phases.

BIBLIOGRAPHY

1. Kin Yen, Bahram Ravani and Ty Lasky, **"Mobile Terrestrial Laser Scanning Workflow Development, Technical Support and Evaluation,"** *Advanced Highway Maintenance and Construction Technology Research Center (AHMCT), University of California at Davis, Report Number: CA14-2517, June 14, 2014*

2. Nagi Mohammed, Ahmed Ghazi and Hussam Eldin Mustafa**, "Positional Accuracy Testing of Google Earth,"** *International Journal of Multidisciplinary Sciences And Engineering, Vol. 4, No. 6, July 2013.*

3. Cutberto Paredes-Hernandez, Wilver Salinas-Castillo, Francisco Guevara-Cortina and Xicotencatl Martinez-Becerra, **"Horizontal Positional Accuracy of Google Earth's Imagery Over Rural Areas: A Study Case in Tamaulipas, Mexico,"** *BCG - Boletim de Ciências Geodésicas - On-Line version, ISSN 1982-2170.*

4. Ashraf Farah and Dafer Algarni, **"Positional Accuracy Assessment of Google Earth in Riyadh,"** *Artificial Satellites, Vol. 49, No. 2 – 2014.*

5. California State Transportation Agency, "**California Manual on Uniform Traffic Control Devices,**" p. 653, 2014, URL: http://www.dot.ca.gov/hq/traffops/engineering/mutcd/pdf/camutcd2014/CAMUTCD2014.pdf.