An Investigation of Interpolation Techniques to Generate 2D Intensity Image From LIDAR Data

IMRAN ASHRAF, SOOJUNG HUR, AND YONGWAN PARK
Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38544, South Korea

Corresponding author: Yongwan Park (ywpark@yu.ac.kr)

ABSTRACT Light detection and ranging (LIDAR) has become a part and parcel of ongoing research in autonomous vehicles. LIDAR efficiently captures data during day and night alike; yet, data accuracy is affected in altered weather conditions. LIDAR data fusion with sensors, such as color camera, hyperspectral camera, and RADAR, proves to be a viable solution to improve the quality of data and add spectral information. LIDAR 3-D point cloud containing intensity data are transformed to 2-D intensity images for the said purpose. LIDAR produces large point cloud, but, while generating images for limited field of view, data sparsity results in poor quality images. Moreover, 3-D to 2-D data transformation also involves data reduction, which further deteriorates the quality of images. This paper focuses on generating intensity images from LIDAR data using interpolation techniques, including bi-linear, natural neighbor, bi-cubic, kriging, inverse distance, and weighted and nearest neighbor interpolation. The main focus is to test the suitability of interpolation methods for 2-D image generation, and analyze the quality of the generated 2-D image. Image similarity metrics, such as root mean square error, normalized least square error, peak signal-to-noise ratio, correlation, difference entropy, mutual information, and structural similarity index measurement, are utilized for camera and LIDAR image matching, and their ability to compare images from heterogeneous sensors is also analyzed. Generated images can further be used for data fusion purpose. Images generated using LIDAR points have a relevant distance matrix as well, which can be used to find the distance of any given pixel from the image. In addiiton, the accuracy of interpolated distance data is evaluated as well by comparing it with the original distance values of traffic cones placed in front of vehicle. Results show that the inverse distance weighted interpolation outperforms other selected methods in 2-D image quality, and images from nearest neighbor appear brighter subjectively.

INDEX TERMS Image generation, intelligent vehicles, interpolation, sensor fusion.

I. INTRODUCTION

Light Detection And Ranging (LIDAR) has become an essential part of ongoing research in autonomous vehicles. LIDAR is widely used in many applications including urban planning, telecommunication, and security services and most recently in intelligent vehicles for environment sensing. The capacity of LIDAR to capture data during all time of the day alike makes it very attractive solution to capture 3D surface information [1]. Moreover, LIDAR can work well during day and night and even shadows do not affect its performance. Modern advanced High Definition LIDAR (HDL) like Velodyne HDL-64E can produce exponentially more data intensive point cloud (1.3 million points per second) [2]. Besides it is equipped with a GPS (Global Positioning System) as well which is used for mapping and tracking purposes. However, this GPS faces hindrances in urban areas, in that case RFID (Radio Frequency Identification) technology [3], [4] can be used for tracking and positioning. In a similar fashion, use of IOT (Internet of Things) sensors is another potential approach [5] which is being tested by many researchers for similar purposes. LIDAR is a laser scanning technology which can provide very accurate distance and intensity information in a fast manner. However, this technology is not beyond limitations. In spite the fact that LIDAR is less affected by

change in weather; precipitation and rain scatters as well as absorbs its waves and captured data suffers from noise and disparity. Additionally LIDAR data do not contain color information. LIDAR data is often fused with other sensors like color camera, hyperspectral camera, RADAR etc. to add spectral information which achieves increased object detection accuracy and robustness [6]–[8]. Conventionally, LIDAR 3D data is transformed into 2D images for data fusion. Data transformation is not a trivial task for many reasons. Firstly, Velodyne LIDAR when mounted on vehicles is intended to generate points for 360° Field of View (FoV) while camera has small FoV in comparison. So, restricting LIDAR FoV reduces data points thus resulting in low resolution image when 3D to 2D transformation is performed. Secondly, LIDAR has orthographic projection which needs to be converted to perspective projection to match camera projection. This projection transformation is often not very ideal. Last but not least, to formulate 2D images from LIDAR scattered data points, interpolation is performed which may produce very different results depending upon the density of the data.

This paper focuses on transforming LIDAR 3D point cloud to 2D images using various interpolation techniques and analyzing the quality of the generated images. The quality of generated images is evaluated using image similarity as a quality parameter and various image similarity metrics are used. Moreover, generated 2D images also have associated distance matrix and its accuracy is checked by comparing it with actual distance of placed objects at manually measured distance.

This paper is organized in the following manner. Section 2 describes the related work. Section 3 narrates the interpolation techniques used for the experiment. Section 4 is about the experiment setup, methodology and data acquisition. Experiment results are discussed in section 5. In the end conclusion is drawn.

II. RELATED WORK

LIDAR is a remote sensing technology which has been used since 1960's for a variety of tasks including flood risk mapping, oil and gas exploration surveys, engineering & construction surveys, coastal area mapping, forestry and urban modelling etc. Now such technology has been deployed in transport planning, accident/crime scene generation, imaging and visualization and gaming as well. It has been very attractive solution for environment sensing in autonomous vehicles for the last decade especially after Defense Advanced Research Projects Agency (DARPA) challenge in 2005. Other than being used to create Adaptive Cruise Control (ACC) system for automobiles, it is widely used to create 3D maps of the environment for autonomous driving tasks. Google robotic car is just one of the examples which is equipped with this high definition LIDAR for pedestrian and obstacle detection and similar autonomous tasks.

Latest HDL e.g. Velodyne HDL-64E, can produce very dense points cloud by generating 1.0 million to 1.3 million points per second. In spite of massive data point cloud, LIDAR points are scattered and objects detectability becomes difficult as the distance between LIDAR and object increases. Moreover, we have blank spots (gaps) where point cloud is sparse and scattered. In order to cater this problem, often interpolation is applied to fill in the gaps and have a uniform distributed point cloud. Image inpainting can also be used to fill the gaps but often it also uses interpolation. LIDAR intensity data is used for interpolation and 2D images are generated which can be fused with other sensors for road detection [9], curb detection [10], obstacle detection [11], roadside parked vehicle detection [12], lane detection [13] and assisting drivers at intersections [14]. Various research works exist in literature which use interpolation on LIDAR data to generate Digital Elevation Models (DEM), Digital Terrain Model (DTM), data fusion and lane detection etc.

Although LIDAR has been widely used in Geographic Information Systems (GIS), yet quite a few papers are found which use interpolation techniques on LIDAR data with respect to autonomous vehicles. P. Steinemann, J. Klappstein, and J. Dickman propose an algorithm in [15] for automatic detection of vehicles by registering consecutive outline contours in LIDAR generated surface. A two dimensional bilinear interpolation is used to increase the number of nodes in contour surface as the initially generated surface from LIDAR is of low resolution. D. A. Thornton, K. Redmill, and B. Coifman present in [16] a research on surveying parallel parking in opposite direction of the travel. Their proposed algorithm involves the generation and smoothing of GPS data from LIDAR. Piecewise Cubic Hermite interpolating Polynomial algorithm is applied to 4 Hz positioning data to achieve higher frequency data. H. Guan, J. Li, Y. Yu, Z. Ji, and C. Wang investigate use of LIDAR in [17] to automatically extract road markings. Initially, LIDAR is used to extract 3D road surface points. Later, acquired surface is interpolated into georeferenced features by the use of Inverse Distance Weighted interpolation method.

L. Smadja, J. Ninot, and T. Gavrilovic work on automatic road extraction in [18] using LIDAR data along with color camera. In the first stage, LIDAR is used to detect road boundary candidates. Later, 3D NURUS (Non Uniform Rational Bezier Spline) parametric curve interpolation is used to approximate sparse and irregularly spaced LIDAR data into a regularly spaced grid. A. A. Matkan, M. Hajeb, and S. Sadeghian introduced a new method in [19] to extract road from LIDAR data in which classification is performed on LIDAR data using Support Vector Machines. Since, LIDAR data has holes so, spline interpolation is utilized to automatically locate and fill these gaps to get a smooth and higher resolution data grid. D. Gonzales-Augilera, P. Rodriguez-Gonzalvez, and J. Gomez-Lahoz in [20] investigated the registering of 3D range images from laser scanners with 2D high definition image from digital camera using collinearity condition. They apply bilinear interpolation to alleviate the influence of gaps in LIDAR data and get higher resolution image.

C. Axel in [21] employs Bi-linear interpolation to generate image from Sick LMS 151 LIDAR which is then fused with Complementary Metal Oxide Semiconductor (CMOS) Canon EOS color camera in indoor environment. J. Li in [22] uses interpolation for data fusion using 2D Hokuyo LIDAR with Nikon S3000 color camera. First camera projection is applied to project LIDAR 3D points into a plane and then an intensity image is acquired by interpolation of LIDAR 2D points. Nearest Neighbor interpolation technique is applied for this experiment. B. Yang, P. Sharma, and R. Nevatia in [23] make use of missing point interpolation to remove noise and enhance the quality of aerial LIDAR data for vehicle detection. C. Reinholtz, D. Hong, A. Wicks, and A. Bacha utilize spline interpolation in [24] for lane detection using Sick LMS on Ford Escape during DARPA challenge 2007. R. Rao, A. Konda, D. Optiz, and S. Blundell in [25] investigate Inverse Distance Weighted, Spline, Tinning and Natural Neighbor interpolation to generate the ground surface model using terrestrial LIDAR data on vehicle. The use of interpolation techniques result in high fidelity surface with good curb, road and obstacle features.

Above cited works have one or more limitations. First, interpolation is used either to generate a uniformly space grid or only one interpolation technique is applied; eventually, comparing interpolation performance is not possible. Second, interpolation is used to produce images using point clouds generated indoor. Since, indoor environments are often simple, narrow and constrained so point cloud is dense and almost evenly distributed thus producing good intensity image. In contrast, outdoor environments are complex where both LIDAR and other objects like vehicles, pedestrians etc. are moving at the same time leading to uneven point cloud. Moreover, in outdoor environment where we don't have tall obstacles LIDAR can sense through long distance (up to 100 meters) which causes thin and scattered point cloud. So, an independent research needs to be initiated to achieve the following objectives.

- To investigate the feasibility of using interpolation techniques to generate 2D intensity images from 3D point cloud obtained using Velodyne HDL LIDAR.
- Analyze the quality of interpolated images by using similarity measure between 2D interpolated image and a vision camera image.
- Generating a 2D image with associated distance matrix and evaluating the accuracy of distance data.
- Evaluating the suitability of similarity measurement techniques for images from heterogeneous sensors.

III. INTERPOLATION

Interpolation is a statistical technique capable of potentially generating the intermediate unknown points of independent variables for spatial data. All interpolation methods are based on Tobler's first law of geography which states that "Everything is related to everything else, but near things are more related than distant things" [26]. Since, LIDAR

point cloud is not equally distributed, interpolation is used to generate unknown points using the position and magnitude of the known points. A large number of interpolation techniques exist and their selection depends primarily on the nature of data. Albeit, many state-of-the-art interpolation techniques are devised including NEDI (New Edge Directed Interpolation), MEDI (Modified Edge-Directed Interpolation, ICBI (Iterative Curvature Based Interpolation), etc [27]–[29]. yet, these techniques are not intended for LIDAR data interpolation. These are adaptive interpolation techniques which are utilized mainly for image zooming and image resolution enhancement where we have low resolution camera images. LIDAR data is irregularly spaced which means that the points do not have any particular symmetry over the extent of the areas and we have missing data. So, interpolation techniques based on gridding methods are inevitable to generate a regularly spaced grid from LIDAR data. We select non-adaptive interpolation techniques to use in our experiment which are briefly described here.

A. BI-LINEAR INTERPOLATION
Linear interpolation is a numerical analysis technique which uses linear polynomials to derive a straight line between the given known points. Bi-linear interpolation extends this idea and performs interpolation on both directions. Equation used for bi-linear interpolation is:

$$y = y_0 + (y_1 - y_0)\frac{x - x_0}{x_1 - x_0} \tag{1}$$

Where y is the unknown value at x for the interval (x_0, x_1).

B. NATURAL NEIGHBOR INTERPOLATION
Natural neighbor interpolation invented by Robin Sibson, uses Voronoi and Delaunay diagrams of a discrete set of spatial points [30]. In order to interpolate a value, it applies weight to the closest points based on their proportionate areas. Equation used for natural neighbor interpolation is:

$$G(x, y) = \sum_{i=1}^{n} w_i f(x_i, y_i) \tag{2}$$

Where $G(x, y)$ is the estimated value of natural neighbor at (x, y), n is the number of nearest neighbors to be used for interpolation, $f(x_i, y_i)$ represents observed values at (x_i, y_i) and w_i is the associated weight. The weights are calculated while deciding how much of neighboring area needs to be stolen when making diagrams.

C. BI-CUBIC INTERPOLATION
Bi-cubic interpolation, an extension to cubic interpolation is used for two dimensional grid data. Contrary to bi-linear interpolation whose results has edges, bi-cubic interpolation produces surfaces with smooth edges. If the function values f and the derivatives f_x, f_y and f_{xy} are known for the four corner $(0, 0)$, $(1, 0)$, $(0, 1)$ and $(1, 1)$ then, interpolated surface is

given by:

$$p(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j \qquad (3)$$

D. INVERSE DISTANCE WEIGHTED INTERPOLATION

Inverse Distance weighted (IDW) interpolation is one of the most frequently used interpolation techniques for LIDAR data. The core idea of this technique is that nearby points are more alike than those which are far apart. Weights assigned to the points closer to the prediction location are greater than those far away. It is calculated using the following equation:

$$\hat{v} = \frac{\sum_{i=1}^{n} \frac{1}{d_i} v_i}{\sum_{i=1}^{n} \frac{1}{d_i}} \qquad (4)$$

Where v_i is the known value and $d_i, \ldots, d_n$ show the distance to the known points.

E. NEAREST NEIGHBOR INTERPOLATION

Nearest Neighbor (NN) interpolation is the simplest of interpolation techniques to implement. In lieu of calculating the weights of the neighboring points this method simply determines the value of the nearest neighbor and uses its intensity value for the unknown point.

F. KRIGING

Kriging is one of the famous spatial interpolation techniques in which the surrounding values are given weights to determine a predicted value for an unmeasured value. Kriging thus takes into account both the distance and the degree of variation of known points to estimate unknown points. It also provides error estimation for each interpolated point which is very helpful to determine the confidence of the modeled sample. Kriging is performed using:

$$\hat{z} = \sum_{i=1}^{n} w_i z_i \qquad (5)$$

Where z_i is the sample value at location i, w_i is a weight and n is the number of samples.

IV. MATERIALS AND METHODS

This section provides the details of the experiment setup, sensors used for the experiment and methodology adopted to conduct the experiment.

A. VELODYNE HDL-64E

It is a high definition LIDAR from Velodyne, equipped with 64 laser fixed on upper and lower laser blocks. Both laser blocks are rotated as a single unit instead of single laser firing through mirror rotation. Point cloud obtained from Velodyne HDL is exponentially denser due to its physical design. The sensor provides a 360° horizontal FoV and a 26.8 vertical FoV while angular resolution is 0.09° [2]. For the experiment 15 Hz spin rate is used. Since, laser scanner has errors in distance and intensity data so, its calibration is also necessary.
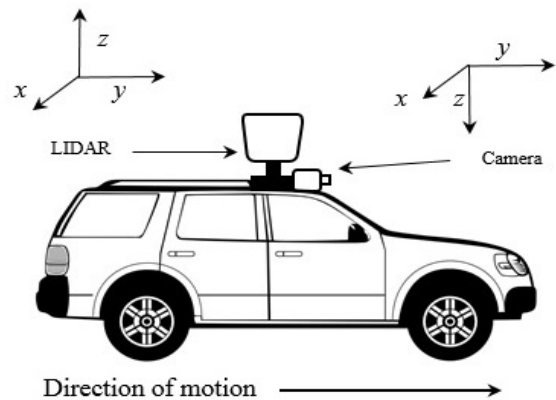

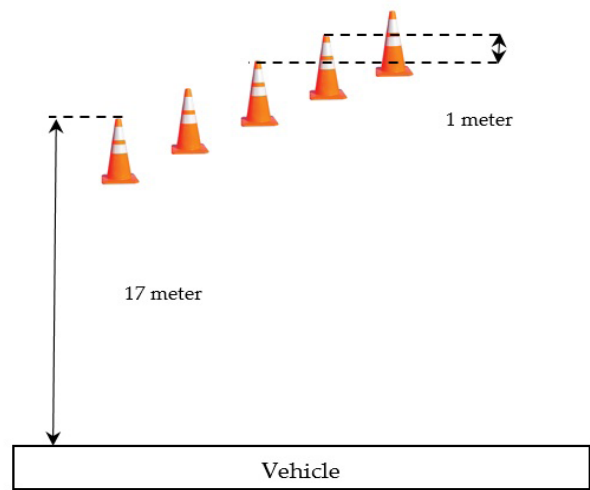**FIGURE 1.** Sensors' position on the vehicle.

FIGURE 2. Placement of traffic cones for distance measurement.

For our experiment Matlab routines provided in Velodyne HDL-64E S2 manual are used for distance and intensity data correction.

B. GO PRO HERO 4 BLACK

It is a 12.0 Mega Pixel (MP) CMOS vision camera with 94.4 vertical FoV and 122.6 horizontal FoV which can provide a screen resolution of up to 3840x2160. It's Frames per Second (fps) ranges from 24 to 120 depending upon the chosen video resolution. A video resolution of 1920x1080 with 30 fps is selected for the experiment.

C. EXPERIMENT SETUP

For data collection Velodyne HDL-64E and Go Pro camera mounted on the vehicle are used in the urban area of Gyeongsan, Republic of Korea. Camera and LIDAR are mounted on vehicle roof at the same position. Fig. 1 shows the sensor placement and their corresponding coordinate systems. Data is collected between 2:00 and 4:00 pm for clear weather conditions in the month of September, 2016. Temperature is 26°C, visibility 10 km, wind 11 km/h, humidity 62%,
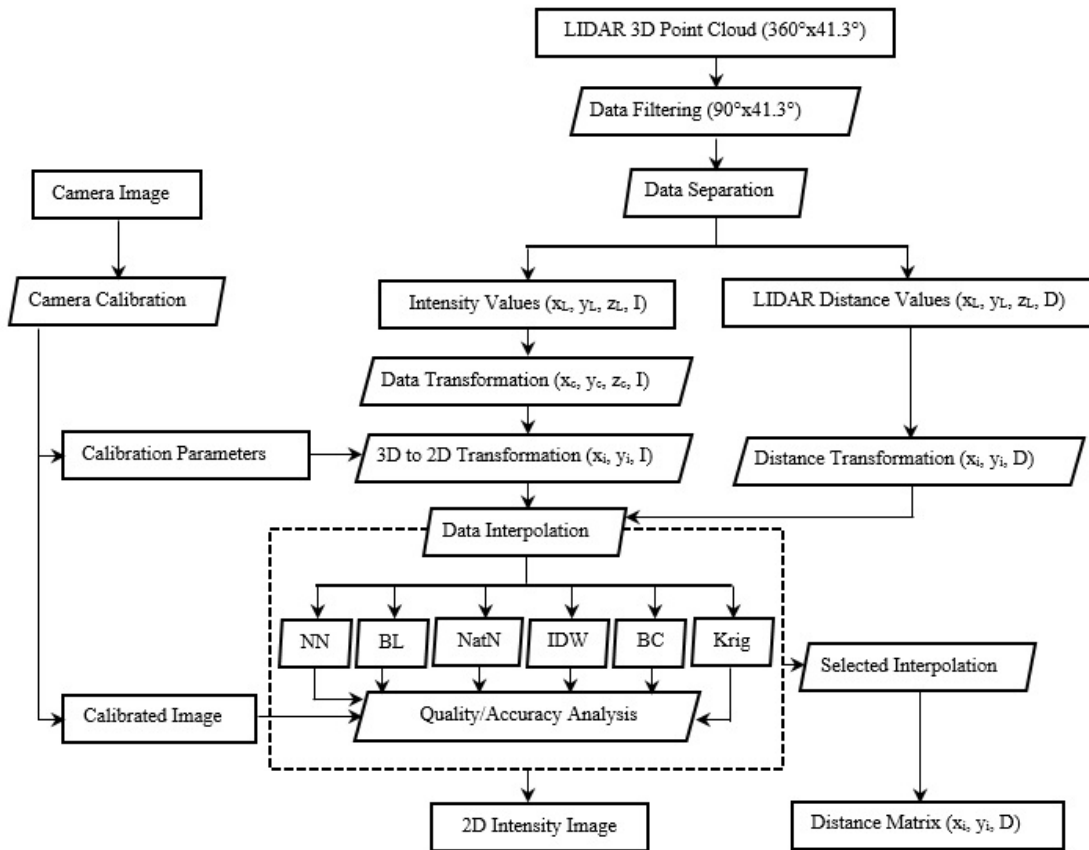
FIGURE 3. Data flow chart for adopted methodology.

dew point 17° and barometer is 1012.00 mb. Data filtering and transformation is performed using Matlab R2015b by taking a single frame. One frame implies one 360° scan of LIDAR unless otherwise specified. Intel core i-3 is used for data transformation and interpolation, with 4GB RAM on Windows 8.1 operating system.

The purpose of the experiment is to generate 2D images with distance data as well. Similarly, in addition to checking the quality of interpolated images, the accuracy of interpolated distance data needs to be evaluated. Since, camera image is used as a reference image for quality analysis, so in a similar fashion we need reference distance data as well. We use traffic cones at specific distance to get the reference distance data. The setting of traffic cones is given in Fig. 2. We place 5 traffic cones in front of vehicle, each with a distance of 1 meter from other. The first traffic cone is placed at exactly 17 meters from the LIDAR. The distance measurement is done manually using measurement tape. Later, these traffic cones are detected in the interpolated image and their corresponding interpolated distance is compared with original distance to check the accuracy of the interpolated distance data.

Fig. 3 shows the flow chart for the adopted methodology. In the following each step of the experiment is described separately in detail.

Step 1: LIDAR scan has data for 360° horizontal FoV however, we need the data which shows only the front view from vehicle. So, data filtering is performed for 90° (LIDAR angle between 0° to 45° and 315° to 360°) horizontal FoV.

Step 2: Distance and intensity data is separated from LIDAR point cloud to generate intensity images. As a result we get two matrices: first contains the intensity data while second has distance data. Coordinates are represented using $L(X_L, Y_L, Z_L)$ for LIDAR, $C(X_C, Y_C, Z_C)$ for camera and $i(X_i, Y_i)$ for image displayed on monitor.

Step 3: Intensity data contains intensity values for a given x, y and z coordinates for LIDAR. Since LIDAR coordinate system is different than camera, we need to translate the LIDAR coordinates into camera coordinates. Equation 6 is used for this transformation:

$$X = PM$$
$$X = \begin{bmatrix} x \\ y \\ z \end{bmatrix}^c$$
$$M = \begin{bmatrix} x \\ y \\ z \end{bmatrix}^L$$
$$P = \begin{bmatrix} R_{3\times3}^T & T_{3\times1} \end{bmatrix} \tag{6}$$

Where X is 3×1 matrix for camera image coordinates,

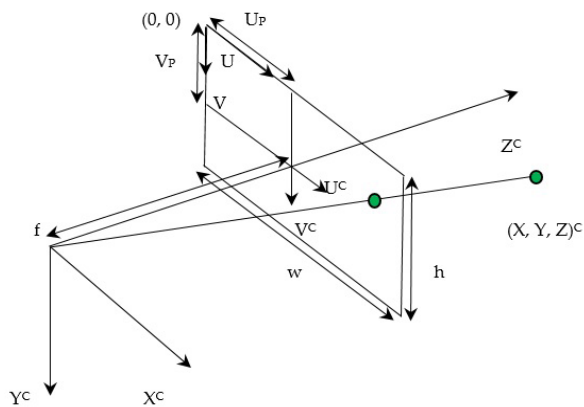**FIGURE 4.** Pinhole camera model.

P shows 3×4 camera extrinsic parameters containing 3×3 rotation and 3×1 translation from LIDAR to camera while *M* represents 3×1 3D LIDAR coordinates.

Step 4: Now 3D camera image to 2D image transformation is performed. Camera calibration is also performed at this stage on the images containing checkers board to get the intrinsic and extrinsic parameters of color camera. Camera 3D to 2D image is transformed using the principle of pinhole camera projection given in Fig. 4.

In Fig. 4, *Z* represents the optical axis while U^C and V^C are camera world coordinates. Then *u* and *v* are obtained using the triangulation rule as follows:

$$u = f\frac{x}{z} \quad v = f\frac{y}{z} \tag{7}$$

Where *u* and *v* are the points in 2D image coordinates and *f* is the focal length.

Step 5: 2D image generated from LIDAR point cloud transformation is of very poor quality. Data interpolation is applied with the objective to fill in the gaps and improve the resolution of the image. Selected interpolation techniques are applied to the low resolution image to get the higher resolution image.

Step 6: Image quality of interpolated images is evaluated using image similarity measure as a quality parameter; the more similar it is to the reference image, the higher quality it possesses. Image similarity methods (metrics) are classified into objective and subjective methods [31]. Objective methods are based on theoretical models and Root Mean Squared Error (RMSE), Signal to Noise Ratio (SNR) are its two examples. On the other hand, subjective methods are based on mathematical foundations like Structural Similarity Index Measurement (SSIM).

Literature is filled with many image similarity methods, we have selected a few which include both pixel based intensity and structural similarity methods. We have chosen seven image similarity methods including RMSE, Normalized Least Square Error (NLSE), Peak Signal to Noise Ratio (PSNR) and Correlation (CORR). RMSE, NLSE, PSNR and CORR are most commonly used approaches for image comparison. The definitions of these metrics are given

in Equations 8-11. In given equations *R(x, y)* and *I(x, y)* represents the reference and target image respectively while *M×N* shows the size of the image and *L* is the maximum pixel value.

$$RMSE = \sqrt{\frac{\sum_{m=1}^{M}\sum_{n=1}^{N}[R(m,n)-I(m,n)]^2}{MN}} \tag{8}$$

$$NLSE = \sqrt{\frac{\sum_{m=1}^{M}\sum_{n=1}^{N}[R(m,n)-I(m,n)]^2}{\sum_{m=1}^{M}\sum_{n=1}^{N}[R(m,n)]^2}} \tag{9}$$

$$PSNR = 10\log_{10}\left(\frac{L^2}{\frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}[R(m,n)-I(m,n)]^2}\right) \tag{10}$$

$$CORR = \frac{2\sum_{m=1}^{M}\sum_{n=1}^{N}R(m,n)I(m,n)}{\sum_{m=1}^{M}\sum_{n=1}^{N}R(m,n)^2 + \sum_{m=1}^{M}\sum_{n=1}^{N}I(m,n)^2} \tag{11}$$

Apart from above mentioned methods, three sophisticated methods Mutual Information (MI), Difference Entropy (DE) and SSIM [32], [33] are also selected. The SSIM proposed by Wang and Lohmann [34] uses the foundation of human perception about the scene. As human visual system is highly capable of getting the structural information so the loss of structural information is a good measure for the approximation of image distortion. The SSIM is measured as:

$$SSIM = \frac{(2\mu_R\mu_I + c_1) + (2\sigma_R\sigma_I + c_2)}{(\mu_R^2 + \mu_I^2 + c_1)(\sigma_R^2 + \sigma_I^2 + c_2)} \tag{12}$$

Where μ_R and μ_I represent the mean values while σ_R, σ_I and σ_{RI} show the corresponding variance values for reference and target images respectively. Two constants C_1 and C_2 are defined to tackle the situations when denominators get close to zero and are calculated using subjectively selected K_1, K_2 and dynamic range of the pixel values with $C_1 = (K_1L)^2$ and $C_2 = (K_2L)^2$. The entropy difference is about the difference between the average amount of information contained in two images and is calculated using:

$$DE = \left|\sum_{g=0}^{L-1}P_R(g)\log_2 P_R(g) - \sum_{g=0}^{L-1}P_I(g)\log_2 P_I(g)\right| \tag{13}$$

Where $P_R(g)$ and $P_I(g)$ represent the pixel values for the reference and target image respectively. The *MI* metric considers the normalized joint gray level histogram and normalized marginal histogram of the two images. The *MI* is calculated using Equation 14, where $h_{RI}(i, j)$ is the normalized joint gray level histogram of two images $h_R(i)$ and $h_I(j)$.

$$MI = \sum_{i=1}^{L}\sum_{j=1}^{L}h_{RI}(i,j)\log_2\frac{h_{RI}(i,j)}{h_R(i)h_I(j)} \tag{14}$$

Selected image similarity measurement methods are utilized to measure quality of interpolated images.
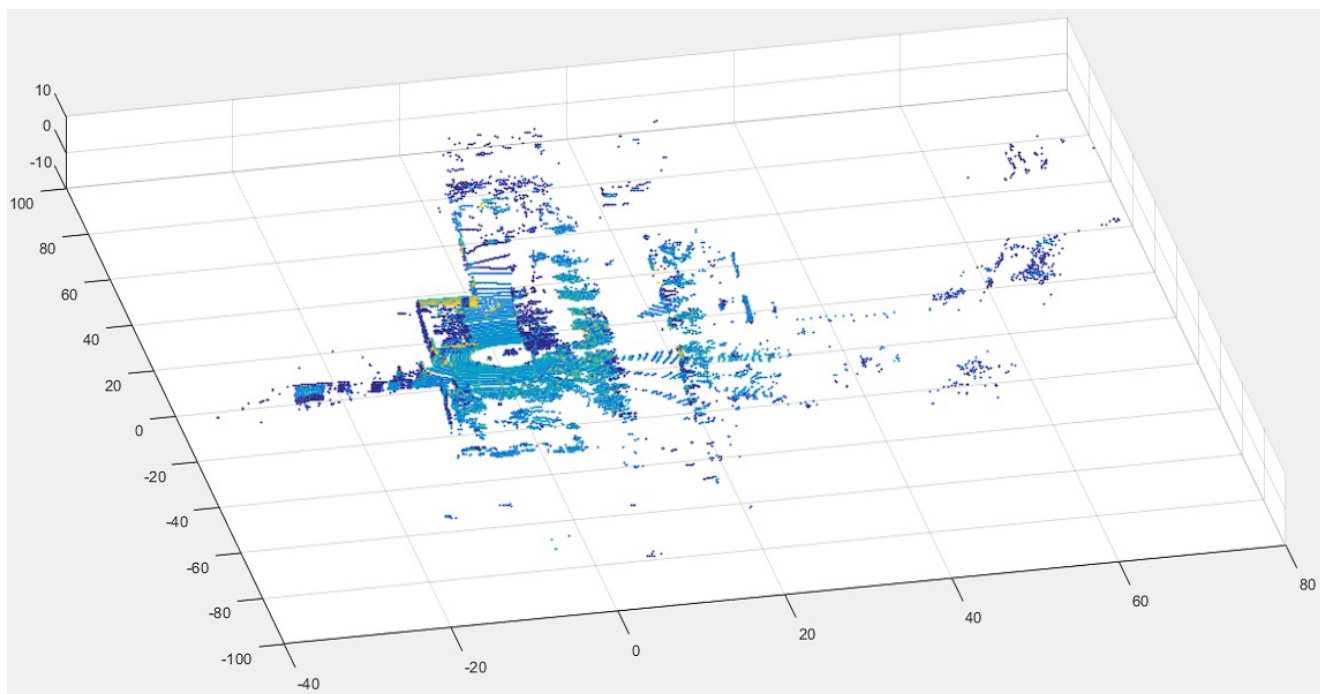
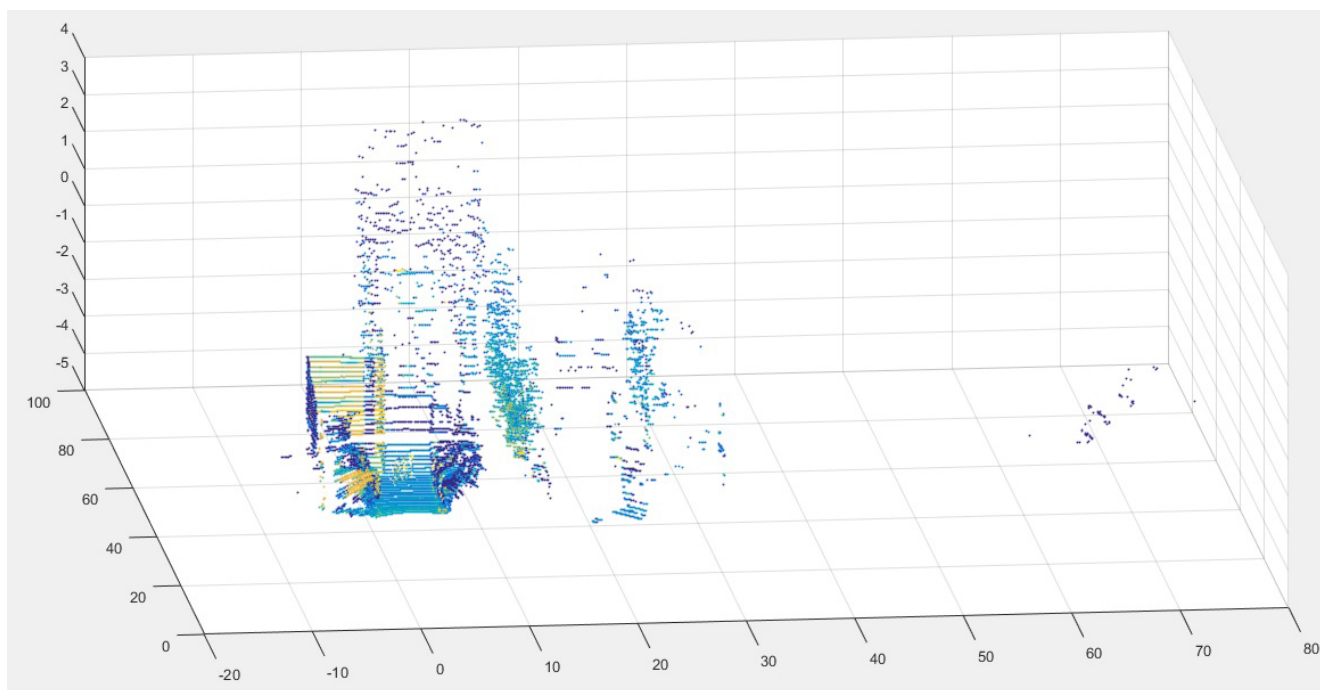**FIGURE 5.** LIDAR point cloud for scan of 360° horizontal FoV.

FIGURE 6. LIDAR filtered point cloud for scan of 90° horizontal FoV.

V. RESULTS AND DISCUSSIONS

LIDAR single frame is taken and filtering is done to get the data for desired FoV. Fig. 5 shows the result for complete single scan while Fig. 6 shows the filtered point cloud. After data filtering, LIDAR data separation into distance and intensity is performed. LIDAR intensity data is to be used to generate 2D intensity image while distance data is used to generate distance matrix. The generated distance matrix contains the distance value for each pixel of 2D intensity image. Camera intrinsic
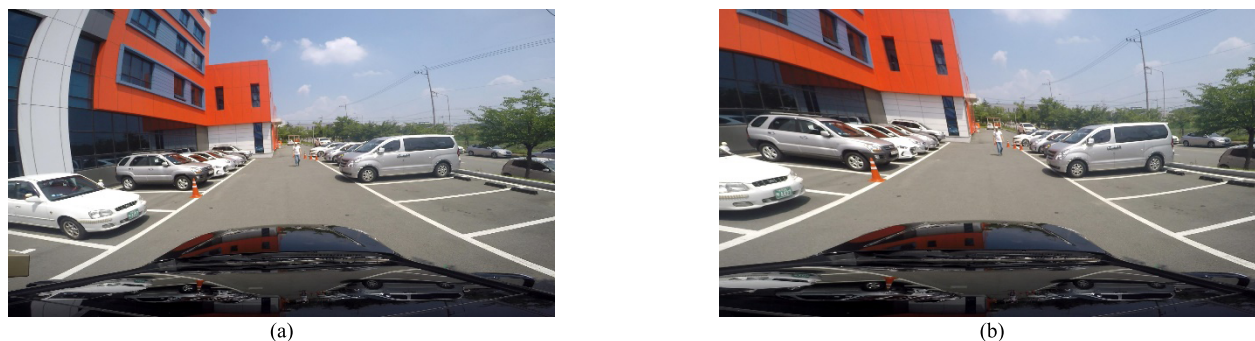
(a)   (b)

FIGURE 7. (a). Camera image before calibration. (b). Camera image after calibration and correction.

TABLE 1. Results for image similarity metrics.

| Interpolation	RMSE	NLSE	PSNR	CORR	DE	MI	SSIM
Original image	0	0	Infinity	1	0	6334.4	1
Bi-linear	11.241	1.0434	5.2535	1	1.0751	7679.4	0.2479
Bi-cubic	11.439	1.0617	5.2541	1	1.9303	8739.5	0.3037
Natural Neighbor	11.038	1.0245	5.2536	1	1.2088	7948.7	0.2581
Kriging	9.2045	0.8543	5.2606	1	2.1329	8961.1	0.3402
IDW	11.366	1.055	5.2537	1	1.5449	9342.3	0.3333
Nearest Neighbor	10.901	1.0118	5.2536	0.9957	0.9701	7525.1	0.2134
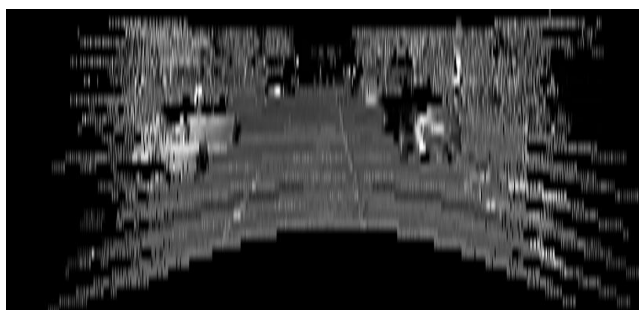
FIGURE 8. 2D image generated from LIDAR point cloud

and extrinsic parameters are achieved using camera calibration.

Moreover, due to wide FoV of Go Pro camera, image is distorted and its correction is necessary as camera image serves as a reference image to check the quality of interpolated image. Camera calibration is performed using Matlab calibration toolbox. Fig. 7 shows camera images before (a) and after (b) calibration.

Now camera 3D to 2D image transformation is carried out using the principle of pinhole camera. Fig. 8 shows that initially generated 2D image is of poor quality. Image quality is affected primarily due to translation of 3D camera coordinates to image coordinates which lead to overlapping of many pixels. We can see in Fig. 8 that there are blank spaces in the image, additionally the image is of low resolution. Image is concentrated in the center while on the sides data is very sparse. Since, data gathered from LIDAR is dense at center while it becomes scattered as we move away from the center

of the LIDAR so, objects have more visibility in the center of the image as compared to at the corners.

Interpolation techniques are applied to fill the gaps and get higher quality image. Selected interpolation techniques are applied and results are given in Fig. 9 (a-f).

To evaluate the quality of interpolated images, image similarity measuring metrics described in section 4.3 are used. Table 1 shows the results of selected image similarity metrics using reference image given in Fig. 10 and images generated from interpolation techniques.

Results indicate that RMSE and NLSE have almost identical behavior to discriminate the distortions caused by interpolation except for the image generated using kriging. Values of PSNR and CORR for selected interpolations are identical for all interpolated images, an indication of their inability to check image similarity from heterogeneous sensors.

For DE, zero means a perfect match between two images. DE value shows that Nearest Neighbor interpolated image is closer to the reference image. With MI, higher value shows strong relationship between the images; the higher the MI value, the stronger the similarity between the compared images is. MI values is higher for IDW, kriging and Bi-cubic interpolation indicative of higher similarity between interpolated and reference image. With SSIM, a value 1 demonstrates a perfect match of the given images. Since, SSIM is more sensitive to degradations caused by blur, compressions and objects displacements so, the values are lower. SSIM and MI are in agreement for image similarity results.

Since the purpose of this research is to generate 2D images with related distance data, so to check the accuracy of
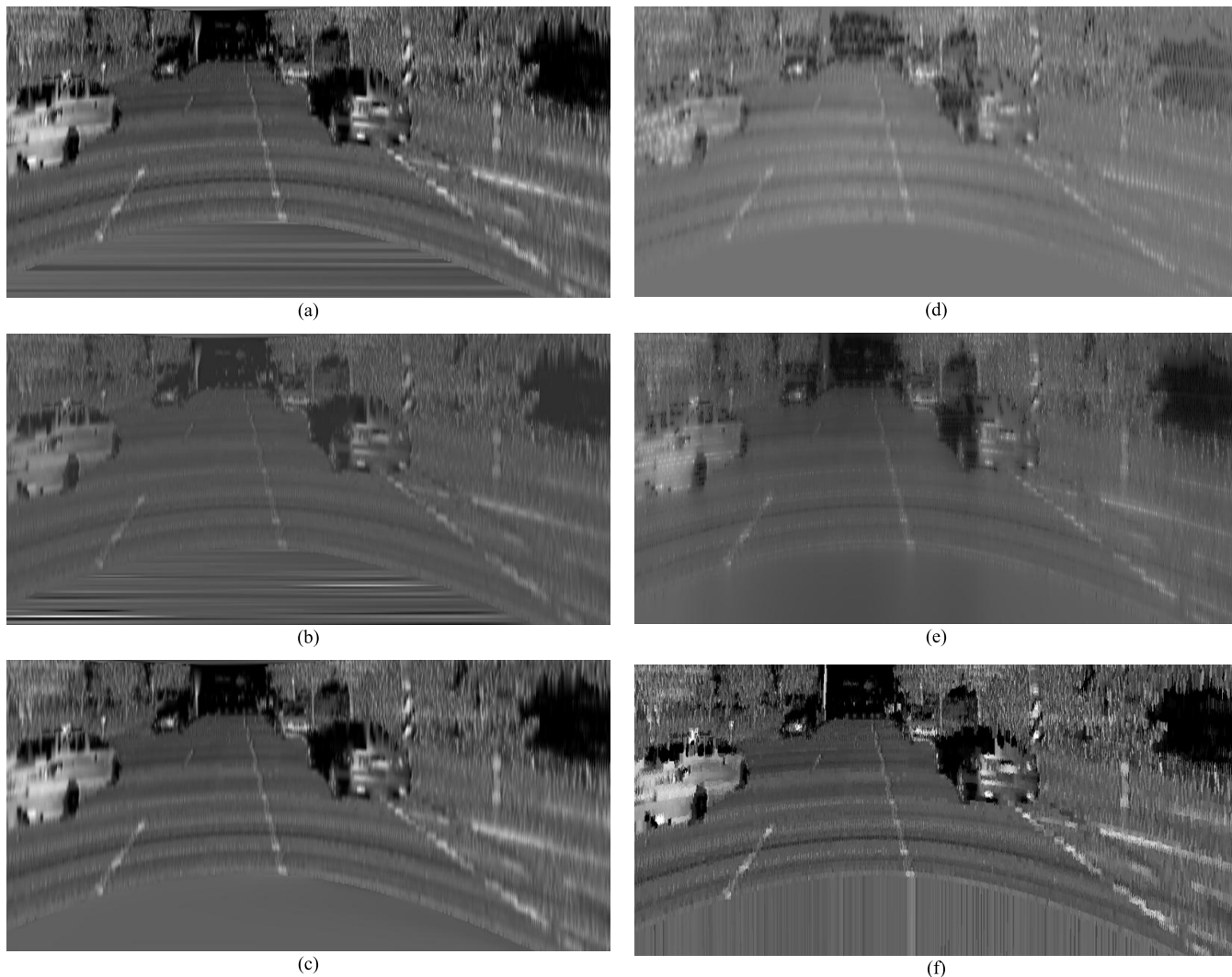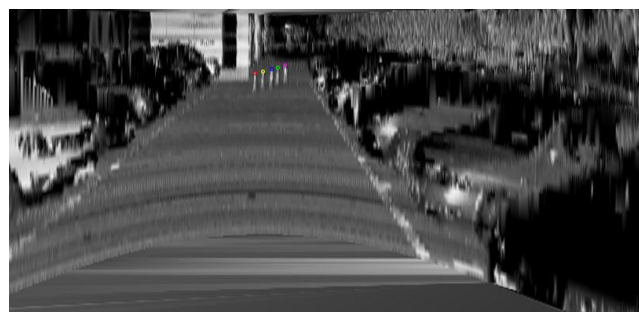
FIGURE 9. (a). 2D image generated from Bi-linear interpolation. (b). 2D image generated from Bi-cubic interpolation. (c). 2D image generated from Natural Neighbor interpolation. (d). 2D image generated from kriging interpolation. (e). 2D image generated from IDW interpolation. (f). 2D image generated from Nearest Neighbor interpolation.

FIGURE 10. Camera reference image.

FIGURE 11. Detected traffic cones.

distance data is also very important. Table 2 shows the error in interpolated distance for five traffic cones.

Fig. 11 shows that five traffic cone heads are detected and their corresponding distance are found in the distance matrix.

Table 2 shows that collectively Bi-linear and Nearest Neighbor has lowest error for distance interpolation. Since, interpolated distance is subtracted from the original distance, negative sign indicates that the interpolated distance is greater

TABLE 2. Results for error (meters) in distance interpolation.

Interpolation	Cone 1	Error	Cone 2	Error	Cone 3	Error	Cone 4	Error	Cone 5	Error
Original distance	17		18		19		20		21	
Bi-linear	17.42	-0.42	19.29	-1.29	21.42	-2.42	21.16	-1.16	23.47	-2.47
Bi-cubic	18.68	-1.68	19.16	-1.16	22.68	-3.68	24.97	-4.97	26.59	-5.59
Natural Neighbor	18.06	-1.06	17.93	0.07	21.54	-2.54	21.77	-1.77	24.32	-3.32
Kriging	27.23	-10.23	26.54	-8.54	30.45	-11.45	33.28	<u>**-13.28**</u>	32.56	-11.56
IDW	19.41	-2.41	20.13	-2.13	25.23	-6.23	27.14	-7.14	28.44	-7.44
Nearest Neighbor	16.61	0.039	18.05	**0.05**	22.16	-3.16	22.75	-2.75	25.14	-3.15

than the original distance. Table 2 shows that Nearest Neighbor interpolation has the lowest (bold value in Table 2) error in distance while the highest (bold underlined value in Table 2) error in distance is when distance matrix is generated using kriging. Results indicate that the error in interpolated distance may vary between 5.0 cm to 13.28 meters.

In spite that image from Nearest Neighbor appears more bright and sharp, analysis of image similarity metrics indicates that IDW interpolation produces 2D images which are more similar to camera images with reference to both pixel intensity and structural composition. However, if we consider accuracy of distance interpolation Nearest Neighbor interpolation outperforms other interpolation techniques as, it has the smallest error of 5cm. With huge errors given in Table 2, use of interpolation techniques especially kriging for distance data is questionable, however we believe further experiments may reveal the cause of such errors and improve the accuracy.

VI. CONCLUSIONS

In this paper raw LIDAR data is processed using Matlab to generate 2D intensity images matching to a specified camera parameters. First LIDAR data is filtered and separated into intensity and distance data. Later data is transformed into 2D image using pinhole camera model and camera intrinsic and extrinsic parameters acquired using camera calibration. Since, 2D image is of very poor quality and low resolution, after wards six interpolation techniques Bi-linear, Natural Neighbor, Bi-cubic, Kriging, Inverse Distance Weighting and Nearest Neighbor are utilized to enhance its quality and resolution. Image similarity is used as a quality parameter to evaluate interpolated images. Image similarity metrics including RMSE, NLSE, CORR, PSNR, DE, MI and SSIM are used to analyze the quality of interpolated images. To check the accuracy of the distance interpolation, interpolated distance is compared with the manually measured distance.

Our experiment reveals that interpolation techniques are suitable enough to generate good images even when the data is scattered. Results show that IDW interpolation is accurate with respect to objective image quality while Nearest Neighbor interpolated images appear brighter. Additionally, image similarity metrics used in the experiment show that images generated by IDW interpolation are more similar to camera in terms of both pixel intensity and structural information.

Nearest Neighbor interpolation is more accurate regarding distance data with a lowest error of 5 cm. It is also observed that RMSE, NLSE, CORR and PSNR are not good measures when comparing images from heterogeneous sensors. Although, interpolation techniques do not seem quite appropriate for distance data interpolation as the error reaches up to 13.28 meters with kriging, yet, further experimentation may uncover the causes of such huge error in distance data interpolation. With the current experiment we may assume that IDW and NN interpolation are suitable enough to interpolate LIDAR intensity images. Future work is to study the impact of LIDAR data density on quality of generated images. An effort to devise a new interpolation technique is also under study to interpolate distance data with less error. Current experiment is performed using Velodyne HDL-64E under clear weather at noon time and use of other LIDARs or different weather conditions may lead to deviation in results.

REFERENCES

[1] J. Shan and S. Aparajithan, "Urban DEM generation from raw LiDAR data," *Photogramm. Eng. Remote Sens.*, vol. 71, no. 2, pp. 217–226, 2005.
[2] *Velodyne LIDAR HDL-64E Manual*, accessed on Oct. 4, 2016. [Online]. Available: http://velodynelidar.com/docs/manuals/63-HDL64 ES2h%20HDL-64E%20S2%20CD%20Users%20Manual.pdf
[3] Z. Wang, N. Ye, R. Malekian, F. Xiao, and R. Wang, "TrackT: Accurate tracking of RFID tags with mm-level accuracy using first-order Taylor series approximation," *Ad Hoc Netw.*, vol. 53, pp. 132–144, Dec. 2016.
[4] R. Malekian, A. F. Kavishe, B. T. Maharaj, P. K. Gupta, G. Singh, and H. Waschefort, "Smart vehicle navigation system using hidden Markov model and RFID technology," *Wireless Pers. Commun.*, vol. 90, no. 4, pp. 1717–1742, 2016.
[5] J. Prinsloo and R. Malekian, "Accurate vehicle location system using RFID, an Internet of Things approach," *Sensors*, vol. 16, no. 6, p. 825, 2016.
[6] E. G. Parmehr, C. S. Fraser, C. Zhang, and J. Leach, "Automatic registration of optical imagery with 3D LiDAR data using statistical similarity," *ISPRS J. Photogram. Remote Sens.*, vol. 88, pp. 28–40, Feb. 2014.
[7] E. Haber and J. Modersitzki, "Intensity gradient based registration and fusion of multi-modal images," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, 2006, pp. 726–733.
[8] B. Zitová and J. Flusser, "Image registration methods: A survey," *Image Vis. Comput.*, vol. 21, pp. 977–1000, Oct. 2003.
[9] L. Xiao, B. Dai, D. Liu, T. Hu, and T. Wu, "CRF based road detection with multi-sensor fusion," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2015, pp. 192–198.
[10] T. Chen, B. Dai, D. Liu, J. Song, and Z. Liu, "Velodyne-based curb detection up to 50 meters away," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2015, pp. 241–248.
[11] C. Creusot and A. Munawar, "Real-time small obstacle detection on highways using compressive RBM road reconstruction," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2015, pp. 162–167.

[12] X. Mei, N. Nagasaka, B. Okumura, and D. Prokhorov, "Detection and motion planning for roadside parked vehicles at long distance," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2015, pp. 412–418.

[13] D. Kim, T. Chung, and K. Yi, "Lane map building and localization for automated driving using 2D laser rangefinder," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2015, pp. 680–685.

[14] J. M. Scanlon, K. D. Kusano, R. Sherony, and H. C. Gabler, "Potential of intersection driver assistance systems to mitigate straight crossing path crashes using U.S. nationally representative crash data," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2015, pp. 1207–1212.

[15] P. Steinemann, J. Klappstein, J. Dickmann, H.-J. Wünsche, and F. V. Hundelshausen, "3D outline contours of vehicles in 3D-LiDAR-measurements for tracking extended targets," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2012, pp. 432–437.

[16] D. A. Thornton, K. Redmill, and B. Coifman, "Automated parking surveys from a LiDAR equipped vehicle," *Transp. Res. C, Emerg. Technol.*, vol. 39, pp. 23–35, Feb. 2014.

[17] H. Guan, J. Li, Y. Yu, Z. Ji, and C. Wang, "Using mobile LiDAR data for rapidly updating road markings," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2457–2466, Oct. 2015.

[18] L. Smadja, J. Ninot, and T. Gavrilovic, "Road extraction and environment interpretation from LiDAR sensors," in *Proc. IAPRS*, vol. 38. 2010, pp. 281–286.

[19] A. A. Matkan, M. Hajeb, and S. Sadeghian, "Road extraction from LiDAR data using support vector machine classification," *Photogramm. Eng. Remote Sens.*, vol. 80, no. 5, pp. 409–422, 2014.

[20] D. González-Aguilera, P. Rodríguez-Gonzálvez, and J. Gómez-Lahoz, "An automatic procedure for co-registration of terrestrial laser scanners and digital cameras," *ISPRS J. Photogram. Remote Sens.*, vol. 64, no. 3, pp. 308–316, 2009.

[21] C. Axel, "Fusion of terrestrial LiDAR point clouds with color imagery," Rochester Inst. Technol., Rochester, NY, USA, Tech. Rep., May 2013.

[22] J. Li, "Fusion of LiDAR 3D points cloud with 2D digital camera image," Oakland Univ. Library, Rochester, MI, USA, Tech. Rep. TA 1560.L5, 2015.

[23] B. Yang, P. Sharma, and R. Nevatia, "Vehicle detection from low quality aerial LiDAR data," in *Proc. IEEE Workshop Appl. Comput. Vis. (WACV)*, Jan. 2011, pp. 541–548.

[24] C. Reinholtz *et al.*, "Odin: Team VictorTango's entry in the DARPA urban challenge," in *The DARPA Urban Challenge*. Berlin, Germany: Springer-Verlag, 2009, pp. 125–162.

[25] R. Rao, A. Konda, D. Opitz, and S. Blundell, "Ground surface extraction from side-scan (vehicular) LiDAR," in *Proc. MAPPS/ASPRS Fall Conf.*, San Antonio, TX, USA, 2006, pp. 1–8.

[26] W. R. Tobler, "A computer movie simulating urban growth in the Detroit region," *Econ. Geograph.*, vol. 46, pp. 234–240, Jun. 1970.

[27] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.

[28] W.-S. Tam, C.-W. Kok, and W.-C. Siu, "Modified edge-directed interpolation for images," *J. Electron. Imag.*, vol. 19, no. 1, p. 013011, 2010.

[29] A. Giachetti and N. Asuni, "Real-time artifact-free image upscaling," *IEEE Trans. Image Process.*, vol. 20, no. 10, pp. 2760–2768, Oct. 2011.

[30] R. Sibson, "A brief description of natural neighbour interpolation," in *Interpreting Multivariate Data*, vol. 21. New York, NY, USA: Wiley, 1981, pp. 21–36.

[31] D. Nister, "An efficient solution to the five-point relative pose problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–770, Jun. 2004.

[32] D. L. Wilson, A. J. Baddeley, and R. A. Owens, "A new metric for grey-scale image comparison," *Int. J. Comput. Vis.*, vol. 24, no. 1, pp. 5–17, 1997.

[33] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[34] Y. Wang and B. Lohmann, "Multisensor image fusion: Concept, method and applications," Univ. Bremen, Bremen, Germany, Tech. Rep., 2000.

IMRAN ASHRAF received the B.S. degree (Hons.) in computer science from Allama Iqbal Open University, Islamabad, Pakistan, in 2004, and the M.S. degree in computer science from the Blekinge Institute of Technology, Karlskrona, Sweden, in 2010. He is currently pursuing the Ph.D. degree at the Information and Communication Engineering Department, Yeungnam University, Gyeongsan, South Korea. He did his research project with Ericsson AB, Sweden, and received a Sparbanksstiftelsen Kronan Scholarship for his outstanding research project. His research areas include object detection using data fusion in altered weather conditions for autonomous vehicles and accident analysis and prevention. He was the recipient of the Gold Medal for the B.S. degree.

SOOJUNG HUR received the B.S. degree from Daegu University, Gyeongbuk, South Korea, in 2001, the M.S. degree in electrical engineering from San Diego State University, San Diego, CA, USA, in 2004, and the M.S. and Ph.D. degrees in information and communication engineering from Yeungnam University, Gyeongsan, South Korea, in 2007 and 2012, respectively. She is currently a Research Professor with the Mobile Communication Laboratory, Yeungnam University. Her current research interests include performance of mobile communication, indoor/outdoor location, and unnamed vehicles.

YONGWAN PARK received the B.E. and M.E. degrees in electrical engineering from Kyungpook University, Daegu, South Korea, in 1982 and 1984, respectively, and the M.S. and Ph.D. degrees in electrical engineering from the State University of New York at Buffalo, Buffalo, NY, USA, in 1989 and 1992, respectively. He was a Research Fellow with the California Institute of Technology, from 1992 to 1993. From 1994 to 1996, he served as a Chief Researcher for developing the IMT-2000 system with SK Telecom, South Korea. In 2000, he was an Invited Professor with the NTT DoCoMo Wireless Lab., Japan. He was a Visiting Professor with the University of California at Irvine, Irvine, CA, USA, in 2003. From 2008 to 2009, he served as a Director of the Technology Innovation Center for Wireless Multimedia by the Korean Government. From 2009 to 2017, he also served as the President of the Gyeongbuk Institute of IT Convergence Industry Technology, South Korea. Since 1996, he has been a Professor of information and communication engineering with Yeungnam University, Gyeongsan, South Korea. His current research areas of interest includes 5G systems in communication, OFDM, PAPR reduction, indoor location-based services in wireless communication, and smart sensors (LIDAR) for smart car. He is serving as the Chairman of the 5G Forum Convergence Service Committee in South Korea.

● ● ●