

3D Reconstruction using a Sparse Laser Scanner and a Single Camera for Outdoor Autonomous Vehicle

Honggu Lee, Soohwan Song and Sungho Jo

Abstract—This paper presents a 3D scene reconstruction method for autonomous vehicle driving in a wide range of outdoor environments. Autonomous vehicles, most of which currently employ laser and image sensors, are required to have systems for object detection, obstacle avoidance, navigation etc. The one of the most important pieces of information for these systems is an accurate dense 3D depth map. However, range data is much sparser than image data, thus the challenging problem is to reconstruct a dense depth map using sparse range and image data.

Here we propose a novel approach to fuse these different types of sensor data to reconstruct 3D scenes which maintains the shape of local objects. Our method features two main phases: the local range modeling phase and 3D depth map reconstruction phase. In the local range modeling phase we interpolate 3D points from the laser scanner using Gaussian Process regression. It estimates 3D measurements across the outdoor environment and accommodates for defective sensor information. In the reconstruction phase, we fuse an image and interpolated points to build a 3D depth map and optimize based on a Markov Random Field. It provides a depth map corresponding to all image pixels. Qualitative and time complexity results show that our approach is robust and fast enough to demonstrate in real-time for an autonomous vehicle in complex urban scenes.

I. INTRODUCTION

3D range data is one of the most important pieces of information for an autonomous vehicle driving in outdoor environments [1] [2]. In the past few years, laser range finders such as Light Detection and Ranging (LiDAR) systems are used for acquiring a highly accurate 3D representation of the world [3] [4]. It can be used for many applications in autonomous vehicle systems. However, although LiDAR systems give us high quality 3D measurements such as 3D point clouds, it is not dense enough to represent outdoor environments in detail. In order to overcome this problem, it can be useful to accumulate 3D point cloud data over time but it is still a problem to process to data in real-time. In this paper we present a real-time approach for reconstructing a 3D depth map using sparse 3D points which are acquired by a Velodyne HDL-64E and RGB data. Many notable attempts have been made to enhance sparse depth data into a more accurate 3D depth map. Diebel and Thrun integrated a low-resolution depth map and high-resolution image for depth map up-sampling using a Markov Random Field (MRF) [5]. They first presented 3D depth map enhancement based

Honggu Lee and Soohwan Song are with the Department of Computer Science, KAIST, Daejeon 305-701, Republic of Korea (e-mail: nick-playa@kaist.ac.kr, dramanet30@kaist.ac.kr)

Sungho Jo is an associate professor with the Department of Computer Science, KAIST, Daejeon 305-701, Republic of Korea (corresponding author, e-mail: shjo@kaist.ac.kr)

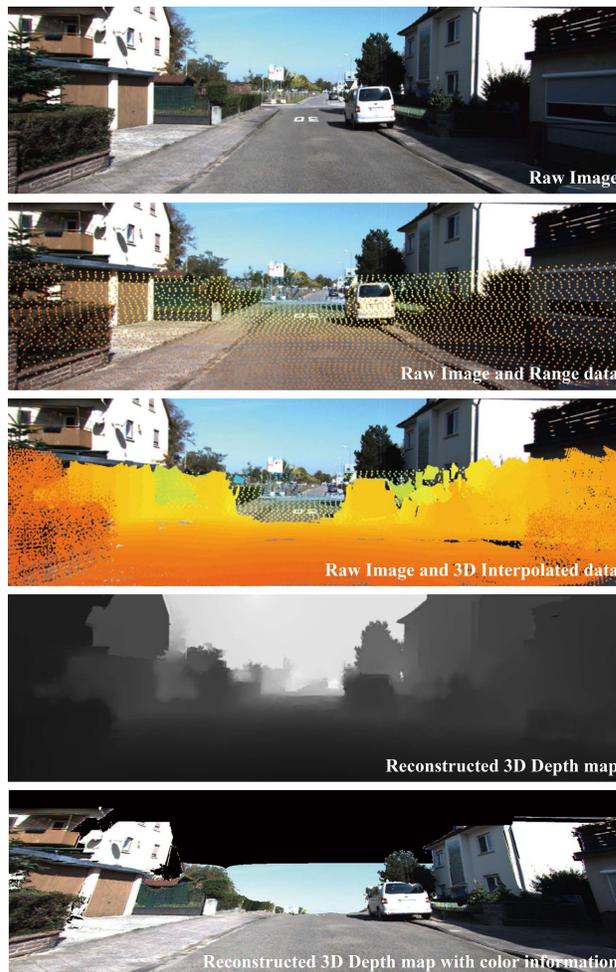


Fig. 1. Dense 3D depth map reconstruction from sparse 3D points and an image using our approach. The raw range data is not dense enough to directly combine the image with the depth map. However, with GP regression, we can generate a dense 3D point clouds. Using these points, we reconstruct a dense depth map as shown in the fourth and last pane. Although the half of scene does not have a depth information, we estimate it using color information based on MRF method.

on MRF. The MRF is solved with a conjugate gradient algorithm. Zhu et al. [6] extend MRF to the temporal domain which is composed by minimizing the energy function with respect to temporal coherence in dynamic scenes. Lu et al. [7] presented a MRF-based depth enhancement method which is solved with loopy belief propagation [8]. Park et al. [9] proposed a MRF optimization scheme with a nonlocal means method that preserved local structure. Yang et al. [10] proposed a new method for dense range image with an

iterative joint bilateral up-sampling (JBU). This approach modified a bilateral filter to up-sample a low resolution depth image.

These approaches focus on enhancing low quality depth maps into high resolution depth maps in indoor environments and use a time-of-flight (ToF) depth camera for acquiring depth information [11] [12]. Thus, these methods are not directly adaptable to outdoor environments. Because depth camera based systems are sensitive to ambient light, this leads to a big restriction for outdoor applications [13]. In contrast to work using a depth camera, there are a few existing methods for providing dense depth maps in outdoor environments using LiDAR. Harrison and Newman [14] presented dense scene reconstruction using a laser scanner. They used a second order smoothness term for planar and curved surfaces based on MRF formulation. Jonathan et al [15] present building a dense 3D range map in an urban environment. This work used [5] for up-sampling range maps. While this studies are more applicable in outdoor environments for autonomous vehicles, the reconstructed depth map's quality is still lower than indoor application methods.

The goal of this paper is to provide 3D reconstruction in real-time for autonomous vehicle driving using sparse 3D data and a single image. To accomplish our goal we split space into discrete voxels which contain local range points and find outlier points with a point histogram feature. We then eliminate the outlier points. Assuming the remaining points in the voxel are from the same object, we make a surface which is mathematically identical to a Gaussian distribution using Gaussian Process (GP) regression. Finally we build a 3D depth map with the interpolated 3D points and RGB image using a Markov Random Field.

The contribution of this paper is 1) locally model and represent outdoor environments using sparse 3D range data and 2) reconstruct a dense 3D scene by fusing interpolated 3D points and image data. Our approach is illustrated in Fig. 1. The rest of the paper is organized as follows. Section II describes the approach in detail, Gaussian Process formulation for learning and MRF for fusing two different types of sensor data. Experimental results are presented in Section III. The last section, section IV provides conclusions.

II. APPROACH

A. Voxelization and Outliers Detection

The outdoor information from a 3D laser scanner represents various objects such as roads, cars, trees, walls etc. We convert 3D point space into voxel space. Every voxel has an arbitrary number of points which belong to various objects. Thus we need to select points which represent a particular object within a voxel.

In our previous work we proposed a point histogram [16] for point by point similarity. We use a point histogram feature to remove points which do not have a high similarity to neighboring points. A point histogram describes every point's geometric distribution with its neighbor points using angle and distance variance along each axis. A point histogram

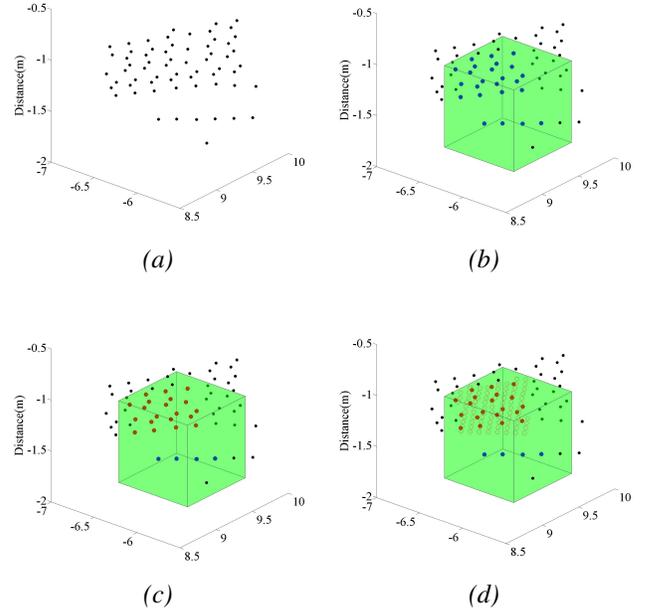


Fig. 2. Voxelization and Outliers Detection (a) 3D points in a local range (b) A fixed size voxel is represented in green and the points inside the voxel are blue (c) Selected points after outlier detection using a point histogram feature in red (d) Interpolated points (red dot) by GP regression in the voxel.

has four different features which represent the geometrical shape of the region around a selected point and distance from the sensor. Let the selected point be $p_m = \{x, y, z\}$ and its neighbor points are the set $\mathbf{P} = \{p_1, \dots, p_M\}$ then each feature is defined as,

$$f_m^k = 1 - \frac{1}{\pi} \arccos \frac{e_m \cdot (p_j - \mu_P)}{\|e_m\| \|p_j - \mu_P\|}, k = \{1, 2, 3\} \quad (1)$$

$$f_m^k = \left(\frac{\lambda_1 \lambda_2 \lambda_3 \times \|\mu_P\|}{V \times D} \right)^2, k = \{4\} \quad (2)$$

where μ_P is the center of the point set, and e_k is the eigenvector, where $\lambda_1, \lambda_2, \lambda_3$ are the eigenvalues corresponding to the eigenvectors e_1, e_2, e_3 , D is the maximum distance in the scene and V is the volume of the voxel. These four features have been normalized to the range [0, 1]. The geometry histogram using these features is then defined as,

$$h_b = \frac{1}{4M} \sum_{j=1}^M \sum_{k=1}^4 \mathbf{I}(f_j^k \in \text{bin}(b)) \quad (3)$$

where h is the b index of the histogram and \mathbf{I} is the indicator function. Using this point histogram we can calculate each point's similarity by χ^2 distance function. Fig. 2 shows the result of voxelization and elimination of points which do not have a similar geometric distribution in the voxel.

B. Local range modeling with GP regression

Since each voxel has an arbitrary number of points which have a similar geometric distribution, we apply GP regression [17] to the local range model. GP is a powerful and efficient nonparametric learning model. GP provides a

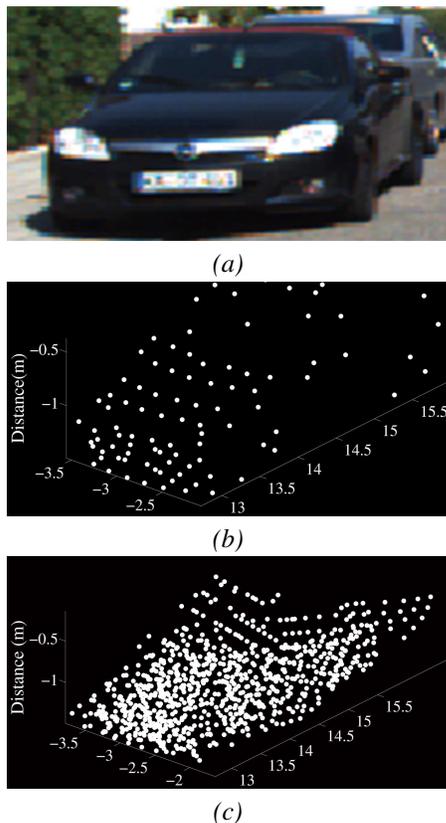


Fig. 3. GP regression result in local range around $1.5\text{m} \times 3\text{m} \times 1.5\text{m}$ size. (a) Raw image data (b) Raw range data (c) Interpolated 3D points by local range modeling.

probabilistic estimate of a 3D depth map and incorporates more precise predictions in the empty space between sparse data points. Let $\mathbf{x} = \{x_1, \dots, x_n\}$ as input data and $\mathbf{z} = \{z_1, \dots, z_n\}$ is target data in voxel $k = \{1, \dots, K\}$. In our case, input data \mathbf{x} is a specific location coordinates at $y-z$ plane and target data is a perpendicular distance from the $y-z$ plane. The plane coordinate system is obtained by KITTI sensor setup [18]. Then we can model the relationship between \mathbf{x} and \mathbf{z} using the underlying function $f(x)$ as a Gaussian noise model. It is written as,

$$z_i = f(x_i) + \mathcal{N}(0, \sigma_k^2) \quad (4)$$

where $f(x)$ is a GP with a mean and covariance function written as $f(x) \sim \text{GP}(m(x), k(x, x'))$ and σ_k^2 is Gaussian variance representing the precision of the noise. Often unnecessary, it is assumed that the mean of the GP is zero. The covariance function $k(x, x')$ provides the corresponding relationship between the given data. The most popular choice is the squared exponential,

$$k(x, x') = \sigma_1^2 \exp\left[\frac{-(x-x')^2}{2l^2}\right] + \sigma_2^2 \delta(x, x') \quad (5)$$

where the first term in (5) is the original 'squared exponential' kernel and the second term is the Kronecker delta function for assuming that observed data has noise with Gaussian zero-mean. The parameter set $\theta = \{l, \sigma_1, \sigma_2\}$ are the hyperparameters of the kernel. In order to use the GP

model, we need to calculate the covariance function among all possible combinations of points. This is performed by maximum marginal likelihood estimation and optimization. By definition of the log marginal likelihood, it is given by using the parameter set,

$$\log p(\mathbf{z}|\mathbf{x}, \theta) = -\frac{1}{2}\mathbf{z}^T \mathbf{C}^{-1} \mathbf{z} - \frac{1}{2}\log|\mathbf{C}| - \frac{n}{2}\log 2\pi \quad (6)$$

where \mathbf{C} is the covariance matrix for all target output \mathbf{z} . To estimate the parameters we use a conjugate gradients algorithm [17].

As with the GP model for regression, we have a joint distribution with \mathbf{z} and estimated output \mathbf{z}_* which is represented by,

$$\begin{bmatrix} \mathbf{z} \\ \mathbf{z}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \mathbf{C} & \mathbf{C}_*^T \\ \mathbf{C}_* & \mathbf{C}_{**} \end{bmatrix}\right) \quad (7)$$

where \mathbf{C} is a n by n covariance matrix which is formulated as $\mathbf{C}(i, j) = k(x_i, x_j) + \sigma_k^2 \delta_{ij}$, \mathbf{C}_* is a n by n_* matrix of evaluated at all pairs of training and new points \mathbf{x}_* and \mathbf{C}_{**} is a n_* by n_* matrix. Then using these covariance matrices, we estimate \mathbf{z}_* corresponding to the new point \mathbf{x}_* and the function of estimation \mathbf{z}_* has mean and covariance functions given by,

$$\mathbf{m}(\mathbf{z}_*) = \mathbf{C}_* \mathbf{C}^{-1} \mathbf{z} \quad (8)$$

$$\text{cov}(\mathbf{z}_*) = \mathbf{C}_{**} - \mathbf{C}_* \mathbf{C}^{-1} \mathbf{C}_*^T \quad (9)$$

Therefore our last equation for data \mathbf{z}_* is represented by

$$p(\mathbf{z}_*|\mathbf{x}_*) = \mathcal{N}(\mathbf{m}(\mathbf{z}_*), \text{cov}(\mathbf{z}_*)) \quad (10)$$

From (7), (8) and (9) we can estimate empty space points elevation with selected points in a voxel. Fig. 3 shows the results local range modeling with GP regression. The results of interpolated points are dense enough and maintains the shape of its objects. In some case, it can be possible to multiply an weighted function with $\mathcal{N}(\mathbf{m}(\mathbf{z}_*), \text{cov}(\mathbf{z}_*))$ such as variogram [19] but in this work we use a constant weight.

C. 3D Reconstruction with interpolated points and image

This section describes how to combine multi-sensor information from a single camera image and 3D interpolated range data for reconstructing a 3D depth map. Although we interpolate 3D points, there are still image pixels missing a corresponding 3D point. Thus we employ the MRF method [5] to obtain a depth value for each pixel in the image. This method enhances each depth value by applying a smoothness term which is weighted according to image intensity derivatives.

We suppose that discontinuities in target range value \mathbf{z}'_* and image intensity \mathbf{I} tend to co-align. Let Ψ and Φ be the depth measurement and depth smoothness potentials respectively. Moreover, since we have two types depth value one is observed and other one is interpolated, we modified depth measurement function. The modified MRF formulation is defined as,

$$p(z'_*|I, z, z_*) = \frac{1}{Z} \exp\left(\frac{1}{2}(\Psi + \Phi)\right) \quad (11)$$

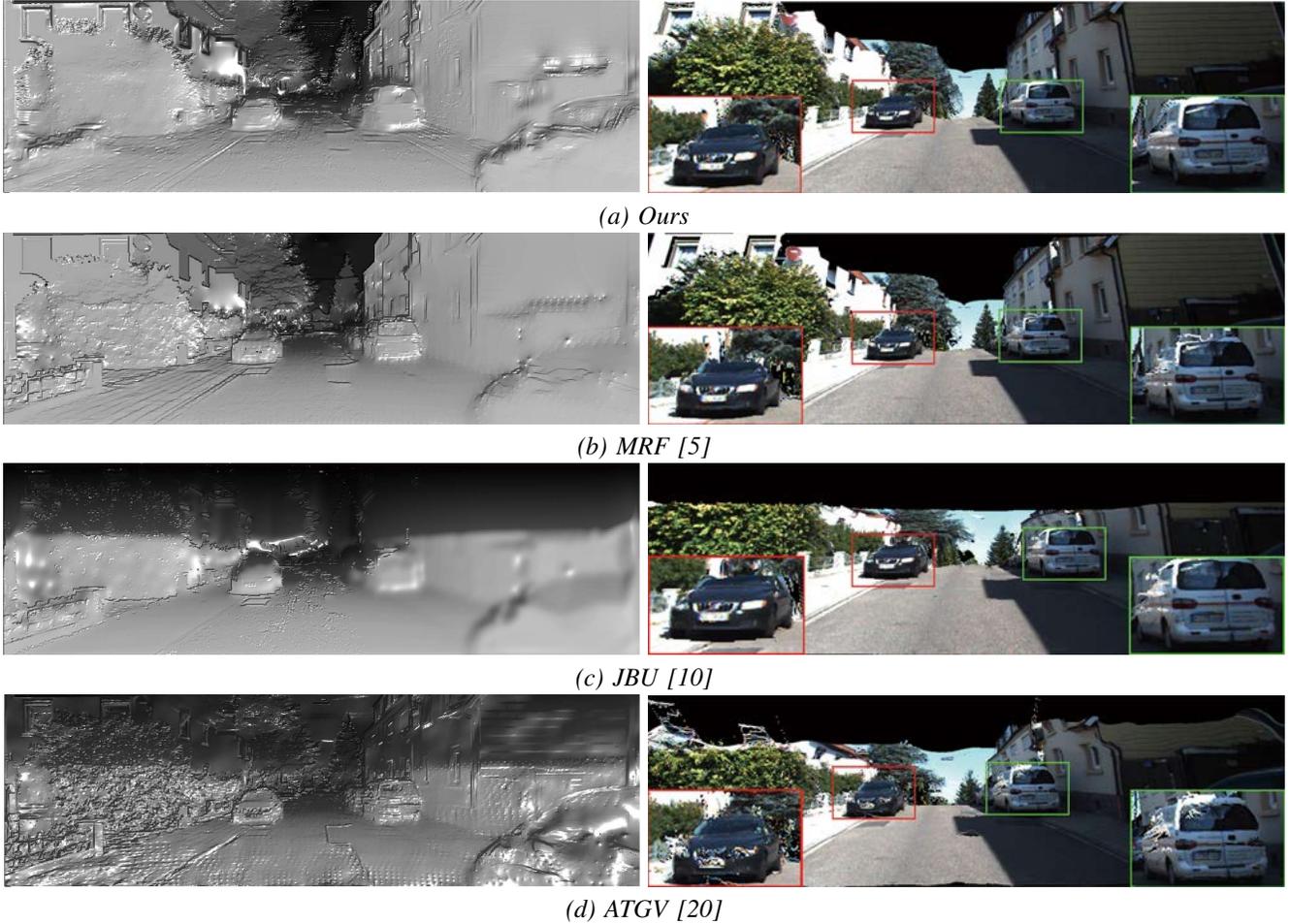


Fig. 4. Qualitative visual comparison on one frame of KITTI dataset for the different methods. Each figure on the left shows a volumetric surface reconstruction result in grayscale, and each figure on the right shows the surface reconstruction with the image data map on top.

$$\Psi = \sum_{i \in L} k_L (I_i - z_i)^2 + \sum_{i \in L_*} k_{L_*} (I_i - z_{*i})^2 \quad (12)$$

$$\Phi = \sum_i \sum_{j \in N(i)} w_{ij} (I_i - I_j)^2 \quad (13)$$

where z is the observed depth value, z_* is the interpolated depth from the local range modeling, k_L and k_{L_*} are the constant weights for depth value accuracy. L and L_* are a set of indices, Z is a normalizer, and $N(i)$ is the neighboring set of the i^{th} pixel. The weight w_{ij} of the smoothness term be defined as

$$w_{ij} = \exp(-c u_{ij}) \quad (14)$$

$$u_{ij} = \|I_i - I_j\|_2^2 \quad (15)$$

where c is a constant weight of image intensity derivatives. The target depth value z'_* can be obtained by determining the maximum a posteriori estimate of $p(z'_* | I, z, z_*)$.

III. EXPERIMENTAL RESULTS

In this section, the experimental results are presented to evaluate our approach quantitatively and qualitatively. The

results show a wide range real world depth map using our approach. Additionally we present computational performance results that show our approach is running in real-time.

A. Dataset and Parameters

We tested our approach using publicly available KITTI datasets [18] which are composed of color images and raw 3D point clouds for each frame from a moving platform while driving around various outdoor scenes. From the dataset, we use calibrated raw data for evaluation. The images of the datasets are extracted from a forward-facing camera (1392×512 pixels), and the raw 3D range data were scanned from a Velodyne HDL-64E sensor (around 60 000 3D points for the forward scene).

A ground truth frame was given for the KITTI datasets but it is sparse information in the scene described above, thus we constructed a ground truth frame by accumulating 20 consecutive frames. We extracted 20 ground truth frames.

In our experiments, we generated long distance 3D depth maps up to 100m. For voxelization, we set the voxel size to 0.4m, a point histogram have 16 bins. We set the MRF parameter k_L and k_{L_*} to 1 and 0.8 respectively and other comparison algorithm parameters follow on their papers.



Fig. 5. Qualitative visual comparison on local range. Each image shows that the reconstructed surface of local range objects such as car, wagon, and bush. The result of a car reconstruction, only our approach maintain side shape of the car.

The system configuration for experiments is a 3.5GHz CPU with 16GB RAM. We implemented our algorithm via Matlab using its built-in standard linear solver.

B. Evaluations and Quantitative Comparisons

To evaluate the performance of our proposed approach, we compared our algorithm against state of the art approaches: Joint Bilateral Upsampling (JBU) [10], Anisotropic Total Generalized Variation (ATGV) [20], and the original MRF model [5]. The JBU uses the joint bilateral filter which is an edge-preserving and noise-reducing smoothing filter. When JBU up-samples a low resolution depth map, it should be initially rescaled to the high resolution image size by using a common interpolation method (e.g., nearest or bilinear). In this experiment, we linearly interpolated empty depth pixels and used the joint bilateral filter. For the depth up-sampling, the ATGV solves a global energy optimization problem using Total Generalized Variation (TGV) regularization. We obtained the results by using the ATGV software provided by the author. The MRF is the same as our algorithm when the interpolated depth value weight $k_{L^*} = 0$.

TABLE I
QUANTITATIVE COMPARISON ON KITTI DATASET

Algorithms	16 Layers	32 Layers	64 Layers	Avg.Time [s]
JBU	3.39 (± 0.31)	2.79 (± 0.44)	2.44 (± 0.28)	23.11
ATGV	4.24 (± 0.53)	3.16 (± 0.35)	2.47 (± 0.32)	342.37
MRF	3.04 (± 0.59)	2.65 (± 0.37)	2.38 (± 0.28)	1.66
Ours	2.87 (± 0.53)	2.45 (± 0.31)	2.29 (± 0.31)	1.92

The numerical results for this experiment in terms of the root mean squared error (RMSE) and computation time are

shown in Table. 1. Each result is the average of processing 20 scenes with different range data density. The results show that our algorithm provides better performance than the other algorithms. In particular, RMSE of our algorithm is lower than any other previous research. Compare to original MRF method, the RMSE indicates that local range modeling with the GP regression task can improve the performance of MRF by spending a little additional computation time. In our experiments, we implement our algorithm via Matlab. Since Matlab is a script language, we believe that implementation in compiled language such as C++ for real-world application can be performed in near real-time.

To determine the robustness of the proposed method, we performed it using much sparse range data which remove some layer data from 64 layers. Following the Table. 1, the RMSE in 21 layers and 32 layers also is lower than any other method.

Since range points of the datasets are very sparse, for reasonable depth map results, we had to set the iteration parameter of ATGV to least 2000. Though ATGV shows good performance, it is not suitable for real-time applications.

C. Qualitative Visual Comparisons

Fig. 4 and Fig. 5 shows a visual comparison for the different methods. In Fig. 4, each image on the left shows a volumetric surface reconstruction which was generated from a depth map. As can be seen in these figures, the results of MRF and our approach have smoother surfaces and cleaner edges than the other algorithms. They also show good reconstruction results in the upper image region where every pixel does not contain range information. ATGV has a relatively rough surface and many protrusive errors. On the other hand, JBU reconstructed very flat and unsophisticated surfaces.

Each image on the right is a color version of the left image, and were rotated to look like a real world 3D scene. In the

upper region of Fig. 4(c) and Fig. 4(d), the reconstruction results of JBU and ATGV are very distorted and inaccurate because their method is unable to estimate depth value if its neighbor depth value does not exist. Though the result of MRF seems to be similar to our approach, the result of reconstruction of the two objects in red and green boxes are different. The top and side of reconstructed objects in Fig. 4(c) are crushed since point clouds in that region are very sparse. However, our approach completely reconstructed that region. The results clearly indicate that our approach can obtain a good result even in regions of sparse range data.

Since the dataset is from urban environments, it includes cars and small bushes. The results of car and small bushes reconstruction is in Fig. 5. The reconstructed local range objects from our approach are less than ideal, but still maintains its surface and captures the characteristic of objects such as flat and scatter.

IV. CONCLUSIONS

We have presented a 3D reconstruction approach using sparse range data and a single image. Our method effectively selects 3D points corresponding to an object and eliminates non-similar points in a voxel using a point histogram feature. Then, using local range modeling with GP regression, we interpolate 3D points which are accurate compared to ground truth and maintain an object's shape. Local range modeling does not require off-line training and provides incorporation of range data with uncertainty. Moreover, 3D reconstruction based on MRF show that our results are superior compared to previous work in terms of both RMSE and running-time.

Through the study of 3D reconstruction we hope that our proposed method could be more improved. Since we interpolated 3D points in a fixed voxel region, sometime surfaces are overlapped and it causes edge distortion. We are interested in handling this problem. Moreover, LiDAR intensity information [21] could help the reconstruction of non-rigid objects such as bushes, tree canopies etc. In the future work we would like to reconstruct more complex outdoor environments that include foliage, bushes and vegetation.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2016R1D1A1B01013573)

REFERENCES

- [1] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, *et al.*, "Stanley: The robot that won the darpa grand challenge," *Journal of field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.
- [2] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, *et al.*, "Towards fully autonomous driving: Systems and algorithms," in *Intelligent Vehicles Symposium (IV)*, 2011 IEEE. IEEE, 2011, pp. 163–168.
- [3] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA)*, 2011 IEEE International Conference on. IEEE, 2011, pp. 1–4.
- [4] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3d-ndt," *Journal of Field Robotics*, vol. 24, no. 10, pp. 803–827, 2007.
- [5] J. Diebel and S. Thrun, "An application of markov random fields to range sensing," in *Advances in neural information processing systems*, 2005, pp. 291–298.
- [6] J. Zhu, L. Wang, J. Gao, and R. Yang, "Spatial-temporal fusion for high accuracy depth maps using dynamic mrfs," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 5, pp. 899–909, 2010.
- [7] J. Lu, D. Min, R. S. Pahwa, and M. N. Do, "A revisit to mrf-based depth map super-resolution and enhancement," in *Acoustics, Speech and Signal Processing (ICASSP)*, 2011 IEEE International Conference on. IEEE, 2011, pp. 985–988.
- [8] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1999, pp. 467–475.
- [9] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon, "High quality depth map upsampling for 3d-tof cameras," in *Computer Vision (ICCV)*, 2011 IEEE International Conference on. IEEE, 2011, pp. 1623–1630.
- [10] Q. Yang, R. Yang, J. Davis, and D. Nistér, "Spatial-depth super resolution for range images," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.
- [11] D. Chan, H. Buisman, C. Theobalt, S. Thrun, *et al.*, "A noise-aware filter for real-time depth upsampling," in *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications-M2SFA2* 2008, 2008.
- [12] J. Dolson, J. Baek, C. Plagemann, and S. Thrun, "Upsampling range data in dynamic environments," in *Computer Vision and Pattern Recognition (CVPR)*, 2010 IEEE Conference on. IEEE, 2010, pp. 1141–1148.
- [13] S. Foix, G. Alenya, and C. Torras, "Lock-in time-of-flight (tof) cameras: a survey," *Sensors Journal, IEEE*, vol. 11, no. 9, pp. 1917–1926, 2011.
- [14] A. Harrison and P. Newman, "Image and sparse laser fusion for dense scene reconstruction," in *Field and Service Robotics*. Springer, 2010, pp. 219–228.
- [15] J. R. Schoenberg, A. Nathan, and M. Campbell, "Segmentation of dense range information in complex urban scenes," in *Intelligent Robots and Systems (IROS)*, 2010 IEEE/RSJ International Conference on. IEEE, 2010, pp. 2033–2038.
- [16] H. Lee, K. Kwak, and S. Jo, "Multi-sensor-based online positive learning for drivable region detection," *Electronics Letters*, vol. 50, no. 20, pp. 1440–1442, 2014.
- [17] C. E. Rasmussen, "Gaussian processes for machine learning," 2006.
- [18] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR)*, 2012 IEEE Conference on. IEEE, 2012, pp. 3354–3361.
- [19] E. Gringarten and C. V. Deutsch, "Teacher's aide variogram interpretation and modeling," *Mathematical Geology*, vol. 33, no. 4, pp. 507–534, 2001.
- [20] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rütther, and H. Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *Computer Vision (ICCV)*, 2013 IEEE International Conference on. IEEE, 2013, pp. 993–1000.
- [21] M. W. Lang and G. W. McCarty, "Lidar intensity for improved detection of inundation below the forest canopy," *Wetlands*, vol. 29, no. 4, pp. 1166–1178, 2009.