IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 8, 2018, accepted December 6, 2018, date of publication December 11, 2018, date of current version January 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886213

Generative Adversarial Network-Based Method for Transforming Single RGB Image Into 3D Point Cloud

PHUONG MINH CHU, YUNSICK SUNG, AND KYUNGEUN CHO, (Member, IEEE)
Department of Multimedia Engineering, Dongguk University, Seoul 04620, South Korea

Corresponding author: Kyungeun Cho (cke@dongguk.edu)

This work was supported by the National Research Foundation of Korean through the Korea Government (MSIP) under Grant 2018R1A2B2007934.

ABSTRACT Three-dimensional (3D) point clouds are important for many applications, including object tracking and 3D scene reconstruction. Point clouds are usually obtained from laser scanners, but their high cost impedes the widespread adoption of this technology. We propose a method to generate the 3D point cloud corresponding to a single red–green–blue (RGB) image. The method retrieves high-quality 3D data from two-dimensional (2D) images captured by conventional cameras, which are generally less expensive. The proposed method comprises two stages. First, a generative adversarial network generates a depth image estimation from a single RGB image. Then, the 3D point cloud is calculated from the depth image. The estimation relies on the parameters of the depth camera employed to generate the training data. The experimental results verify that the proposed method provides high-quality 3D point clouds from single 2D images. Moreover, the method does not require a PC with outstanding computational resources, further reducing implementation costs, as only a moderate-capacity graphics processing unit can efficiently handle the calculations.

INDEX TERMS Artificial intelligence, image processing, sensors, machine learning, neural networks.

I. INTRODUCTION

Three-dimensional (3D) point clouds are widely used in a variety of applications such as object tracking [1]–[4] and 3D scene reconstruction of both indoor [5], [6] and outdoor [7]–[14] environments. In the latter, 3D point clouds along with two-dimensional (2D) textures allow to reconstruct photorealistic 3D scenes. Point clouds are usually obtained from laser scanners, but their cost is sometimes prohibitive. Therefore, we aim to obtain the 3D point cloud from a single red–green–blue (RGB) image captured from an inexpensive 2D camera. Several methods are available to estimate 3D data from 2D images [15]–[18]. Nevertheless, these methods present disadvantages, as discussed in section 2.

Generative adversarial networks (GANs) [19]–[24], inspired by the Darwin's theory of evolution, are being extensively studied in artificial intelligence. A GAN model contains two networks, namely, generator and discriminator. The generator creates candidates that are evaluated by the discriminator, and these networks compete to increase the error rate of each other. After several training iterations, the competition outcome retrieves high-quality synthetic candidates. Based on GANs, we aim to develop a system that automatically generates a 3D point cloud from a single RGB image. Specifically, a GAN model generates a depth representation from the RGB image, and the estimated representation is used to create the corresponding 3D point cloud. We employed Kinect datasets for training and testing the proposed method.

The remainder of this paper is organized as follows. Section 2 provides an overview of the literature related to 3D point cloud generation, depth image generation, and some applications of GANs. Section 3 details the proposed point-cloud generation method. In Section 4, we provide experimental results and evaluations. Finally, we draw conclusions and propose directions of future work in section 5.

II. RELATED WORK

Galabov *et al.* [15] and Swarna Priya *et al.* [16] present methods for generating a 3D scene from multiple 2D images. In these methods, the extracted features across several 2D images captured from different views of a scene are used to estimate the corresponding 3D data, retrieving accurate

results. However, these methods cannot be applied to a single 2D image. Lee *et al.* [17] proposed a 3D scene generator from a single RGB image by extracting a list of line segments from an indoor image, and then generating a scene interpretation from the line segments. However, this method is not accurate if the environment has many small objects with various shapes. Abdulqawi and Mansor [18] presented a similar study to the one we propose by estimating a dense point cloud from a single 2D image. Nevertheless, their method is suitable only for single objects, thus impeding estimation of 3D scenes, which generally contain multiple objects. The method we propose aims to overcome these disadvantages by employing a new approach for research on the transformation of 2D into 3D data.

GANs were introduced by Goodfellow *et al.* [19] in 2014. The GAN model comprises two competing neural networks, and after training, both networks become ''experts'' in generating and discriminating images. This method retrieves synthetic images that look, at least superficially, authentic to human observers. This ability has turned GANs into a hot research topic in artificial intelligence, with thousands of publications available worldwide[20]–[24]. For instance, Wu *et al.* [20] present a GAN method to generate 3D objects from vectors without requiring a reference image or object by using voxels for representation. However, the method can generate only individual objects with low resolution, thus being not suitable for reconstructing 3D scenes of indoor and outdoor environments.

The method we propose requires the transformation of data from an RGB image into depth information. In addition, as we used Kinect datasets, the transformation must handle images with resolution of 640 × 480 pixels. Zhu *et al.* [21], Isola *et al.* [22], Radford *et al.* [23], and Wang *et al.* [24] proposed several image-to-image conversion methods aiming to automatically map an image from one group into one from other groups and vice versa. Although these GAN-based models are similar to our method, the results in [21]–[23] are only suitable for low-resolution images (up to 256 × 256 pixels), as higher resolution retrieves repeated parts and blurry areas. Therefore, using these methods for 3D point cloud estimation from depth images would introduce excessive noise.

Wang *et al.* [24] overcame this issue by proposing high-resolution synthetic image generation, in a method called pix2pixHD, from label maps and instance maps. The method can handle an image resolution of 2048 × 1024 pixels, being suitable for our objective. In fact, as we need to generate highly accurate depth images for estimating point clouds, pix2pixHD would be a good choice. However, pix2pixHD requires a high-performance PC with at least a 12 GB of video random access memory (VRAM) of graphics processing unit (GPU) for implementing the standard model, whereas the full model demands a 24 GB of VRAM. Therefore, pix2pixHD cannot be directly implemented on a PC with regular capacity. Hence, we customized a method based on pix2pixHD to efficiently run on a PC having a 4 GB GPU, as detailed in section 3. Finally, to estimate the 3D point cloud from the depth image, we employed the camera model proposed by Zhang *et al.* [25].

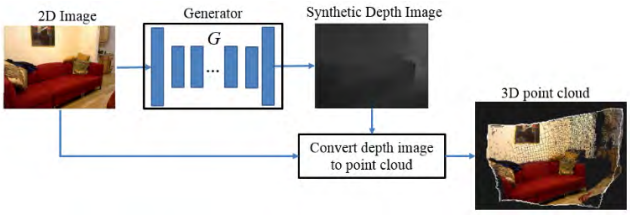


FIGURE 1. Diagram of proposed 3D point cloud generation.

In [26]–[31], a variety of deep neural networks to create depth images and 3D scenes from single RGB images are presented. We compared these methods to our proposed method to evaluate its performance on the NYU-Depth V2 dataset [32]. Moreover, we tested the proposed method on the TUM's RGB-D SLAM Dataset and Benchmark [33].

III. THREE-DIMENSIONAL POINT CLOUD RECONSTRUCTION

The proposed method generates a 3D point cloud from a single RGB image in two stages, as illustrated in Figure 1. First, a depth image is created via the generator of a GAN, whose input is a 2D digital image, obtained after training. Second, we estimate the corresponding 3D point cloud from the generated depth image. At this stage, we also consider the 2D image as input to gather color information for the point cloud. Each stage of the proposed method is detailed below.

A. GAN FOR TRANSFORMING RGB IMAGE INTO DEPTH IMAGE

To transform an RGB image into the corresponding depth image, we customized pix2pixHD proposed in [24]. In turn, pix2pixHD relies on the conditional GAN proposed by Isola *et al.* [22] to generate photorealistic images from semantic label maps and instance maps. As we aim to generate depth images, which is a completely different objective, we cannot apply pix2pixHD directly. Instead, we modified pix2pixHD to satisfy two goals, namely, create high-quality synthetic depth images from RGB ones and minimize the hardware requirements to efficiently run the proposed method on a limited GPU. The proposed model is illustrated in Figure 2 and contains two networks. Generator G learns to map 2D image X with random noise vector Z into depth image Y, $G : \{X, Z\} \rightarrow Y$. The generator is trained to produce depth images that cannot be distinguished from the ground truth using adversarial discriminator D, which is trained to detect the synthetic depth images created by generator G.

In our GAN model, we employed a single global generator for the pix2pixHD model. Generator G includes three components: one convolutional front-end, one set of residual blocks, and one transposed convolutional back-end. In addition, we

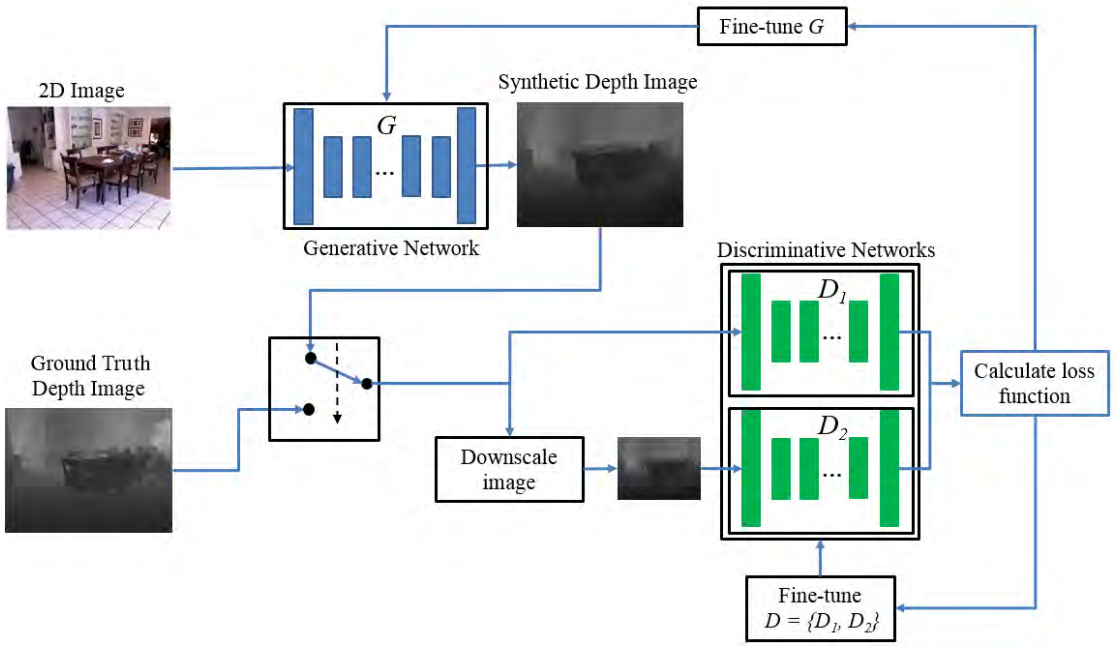


FIGURE 2. GAN model for generating depth image from 2D image.

decomposed discriminator D into two sub-discriminators, D_1 and D_2. Discriminator D_1 works with the full-resolution synthetic images retrieved by the generator, whereas D_2 works with half-scale synthetic images. Hence, discriminator D_1 provides a global view of the depth image to guide generator G to create globally consistent images, whereas discriminator D_2 directs generator G to create sharp and accurate images. Moreover, D_2 prevents G from retrieving repeated patterns on the synthetic depth images. Both discriminators have the same structure employed in [24].

The objective function of our conditional GAN model is based on that from pix2pixHD. We propose to model the conditional distribution of depth images given the input RGB images via the following minimax game:

$$\min_{G}\left(\left(\max_{D_1D_2}\sum_{k=1,2}\mathcal{L}_1\left(G,D_k\right)\right)+\mu\sum_{k=1,2}\mathcal{L}_2\left(G,D_k\right)\right) \tag{1}$$

where $\mathcal{L}_1$ and $\mathcal{L}_2$ are loss functions defined in formulas (2) and (3), respectively, with $\mathcal{L}_1$ corresponding to the objective function of the conditional GAN in [22] and $\mathcal{L}_2$ defining a feature matching loss function, μ weighs feature matching loss, and D_k denotes the sub-discriminator. In formula (3), $D_k^{(i)}$ is the i^{th}-layer feature extractor of discriminator D_k, and T and N_i represent the number of layers in discriminator D_k and number of elements per layer, respectively. In [22] and [24], the L_1 distance is preferred over the L_2 distance as it reduces blurring, and therefore we selected the L_1 distance in formula (3). Overall, G tries to minimize the objective against adversarial D that tries to maximize it.

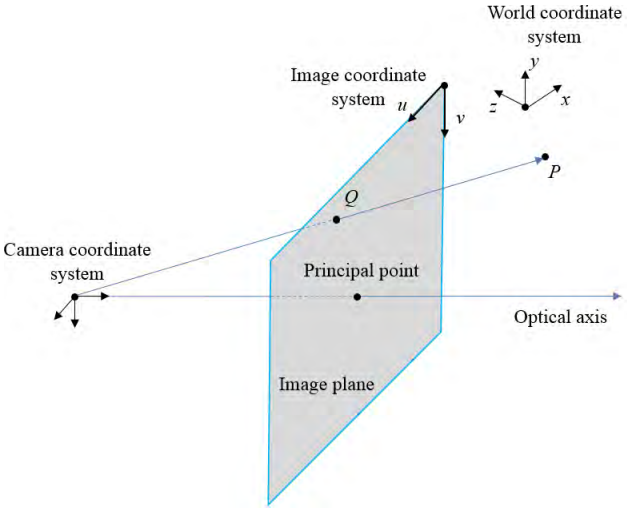


FIGURE 3. Coordinate systems and characteristics of camera model.

$$\mathcal{L}_1\left(G,D_k\right)=E_{X,Y}[\log D_k\left(X,Y\right)]+E_{X,Z}[\log(1-D_k(X,G(X,Z)))] \tag{2}$$

$$\mathcal{L}_2\left(G,D_k\right)=E_{X,Y}\sum_{i=1..T}\frac{1}{N_i}\times\left(\left\|D_k^{(i)}\left(X,Y\right)-D_k^{(i)}\left(X,G\left(X,Z\right)\right)\right\|_1\right) \tag{3}$$

After each iteration, we fine-tune networks G and $D = \{D_1, D_2\}$ and train the model until both G and D become experts. After training, we use generator G to retrieve depth

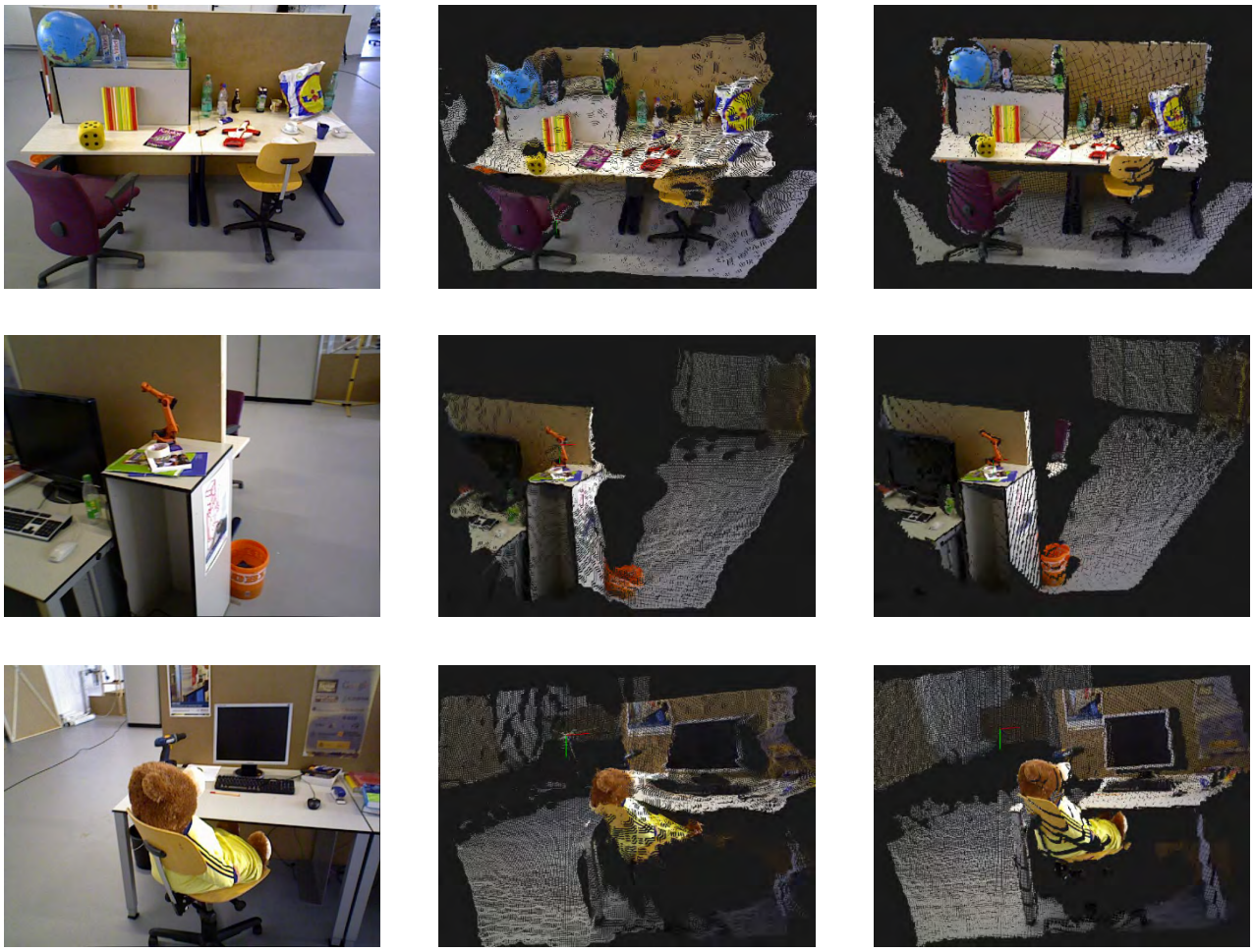


FIGURE 4. Outcomes from the proposed method on the TUM's RGB-D SLAM Dataset and Benchmark. (a) Input RGB images; (b) 3D point clouds generated by the proposed method; (c) 3D point clouds generated by ground-truth depth images.

images from single RGB images to finally create the 3D point clouds as detailed in the sequel.

B. CONVERSION OF DEPTH IMAGE INTO 3D POINT CLOUD

We estimate the 3D point cloud from a depth image based on the camera calibration technique proposed by Zhang *et al.* [25], who showed that each camera has intrinsic parameters that enable estimation. Figure 3 illustrates the coordinate systems of a camera model. Point Q on the plane can be easily obtained from point P in the space through a perspective projection. Conversely, we cannot exactly determine point P in space if only point Q on the plane is available, because $Q \rightarrow P$ is not a one-to-one mapping. However, if the intrinsic parameters of the capturing camera and depth at location Q are available, it is possible to exactly determine point P in space. Based on the camera calibration technique and the characteristics of Kinect sensor, we were able to convert depth images into the corresponding 3D point clouds. The intrinsic parameters of a depth camera are f_u, f_v, c_u, and c_v, where f_u and f_v are the focal lengths of the camera along the u and v axes, respectively, and c_u and c_v represent the principal point. In Figure 3, (c_u, c_v) is the center of the image plane. In the experiments, these parameters should be selected according to the employed depth camera which we use for training phase in section 3.1.

The relationship between point $P(p_x, p_y, p_z)$ in 3D space and point $Q(q_u, q_v)$ on the corresponding 2D depth image is given by formula (4). The inverse transformation results in formula (5) to convert a pixel at location (q_u, q_v) in the depth image into 3D point (p_x, p_y, p_z). In formula (5), parameter d is the depth of location (q_u, q_v) with respect to the camera and s is a scale factor. For determining the coordinates of point P in space, we need to first calculate the z-axis value of point P, p_z, by multiplying depth d by scale factor s, and then calculate p_x and p_y using their corresponding equations in formula (5).

$$\begin{pmatrix} q_u \\ q_v \\ 1 \end{pmatrix} = \begin{pmatrix} f_u & 0 & c_u & 0 \\ 0 & f_v & c_v & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

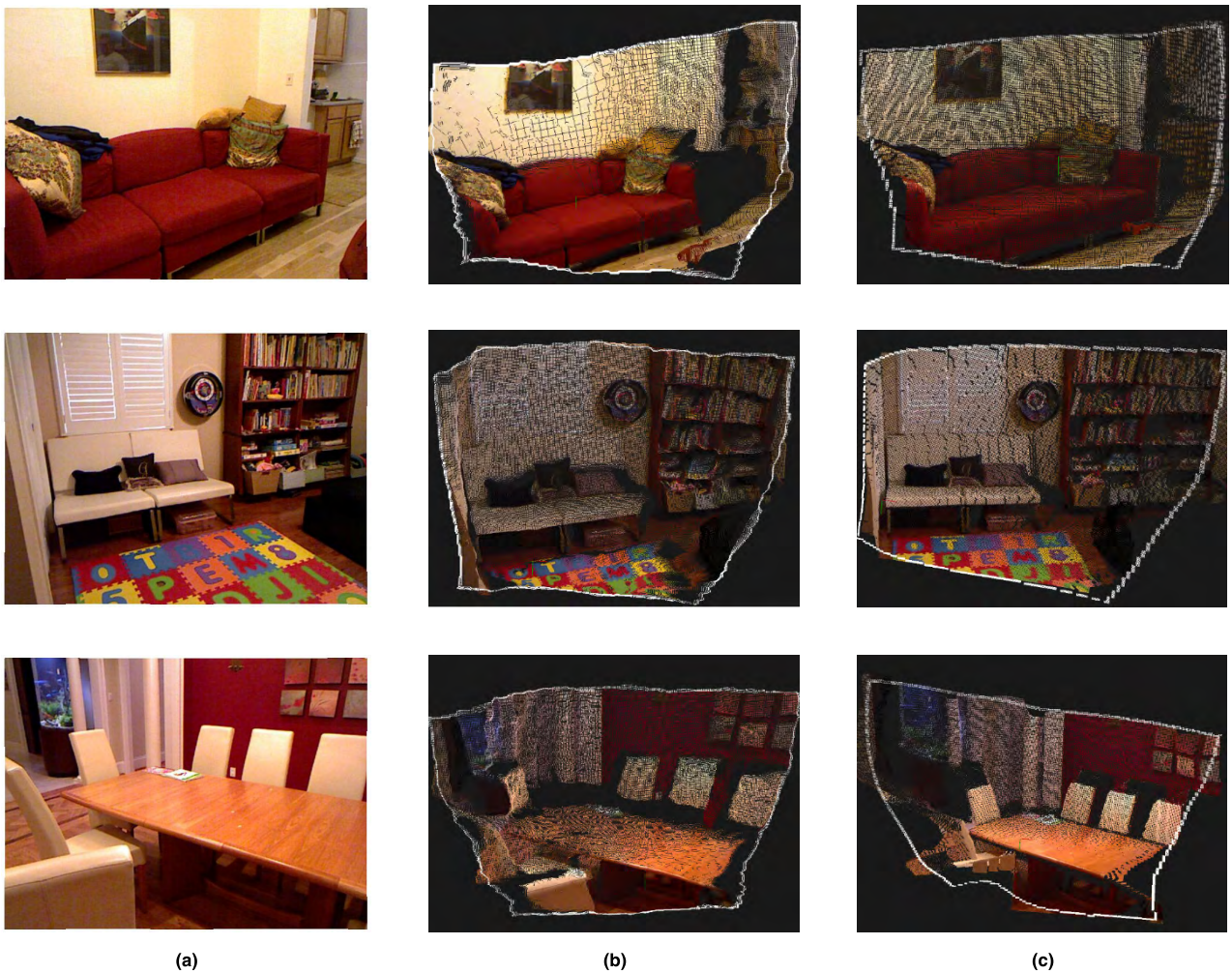


FIGURE 5. Outcomes from the proposed method on the NYU-Depth V2 dataset. (a) Input RGB images; (b) 3D point clouds generated by the proposed method; (c) 3D point clouds generated by ground-truth depth images.

$$\begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} = \begin{pmatrix} f_u p_x + c_u p_z \\ f_v p_y + c_v p_z \\ p_z \end{pmatrix} = \begin{pmatrix} \frac{f_u p_x}{p_z} + c_u \\ \frac{f_v p_y}{p_z} + c_v \\ 1 \end{pmatrix} \tag{4}$$

$$\begin{cases} p_x = \dfrac{p_z(q_u - c_u)}{f_u} \\ p_y = \dfrac{p_z(q_v - c_v)}{f_v} \\ p_z = d \times s \end{cases} \tag{5}$$

IV. EXPERIMENTS AND RESULTS

We implemented experiments to verify the proposed method both in qualitative and quantitative terms. Moreover, several state-of-the-art methods were used for comparison to demonstrate the effectiveness and high performance of our approach.

A. EXPERIMENTAL SETUP

For the experiments, we employed datasets obtained from Kinect sensors. Specifically, we used the NYU-Depth V2 dataset [32] and the TUM's RGB-D SLAM Dataset and Benchmark [33]. Both datasets are popular and were employed in many research to compare own performance. The NYU dataset includes 1449 pairs of RGB and depth images, from which 795 pairs were used for training and 654 pairs for testing. The TUM dataset includes 2500 pairs of RGB-D images, from which 2300 were used for training and 200 for testing. In our method, the quality of the resulting 3D point clouds depends on the depth estimation results. Therefore, we constructed a set of ground-truth depth images to quantitatively evaluate the method. In addition, we compared the performance of our method against others on the NYU dataset. To evaluate quality, we employed four commonly used performance measures, namely, average relative error (REL), root-mean-square error (RMSE), average $\log_{10}$ error, and accuracy with thresholds, given by formulas (6) to (9), respectively.

$$REL = \frac{1}{N} \sum\nolimits_{P} \frac{|d_P^{GT} - d_P|}{d_P^{GT}} \tag{6}$$

FIGURE 6. Depth images generated by the proposed and comparison methods on the NYU-Depth V2 dataset. (a) Input RGB images; (b) depth images using the method in [26]; (c) depth images using the method in [31]; (d) depth images using the proposed method; (e) ground-truth depth images.

$$RMSE = \sqrt{\frac{1}{N}\sum_{P}\left(d_P^{GT} - d_P\right)^2} \tag{7}$$

$$L10E = \frac{1}{N}\sum_{P}\left|log_{10}d_P^{GT} - log_{10}d_P\right| \tag{8}$$

$$ACC_\delta = \frac{N_\delta}{N} \tag{9}$$

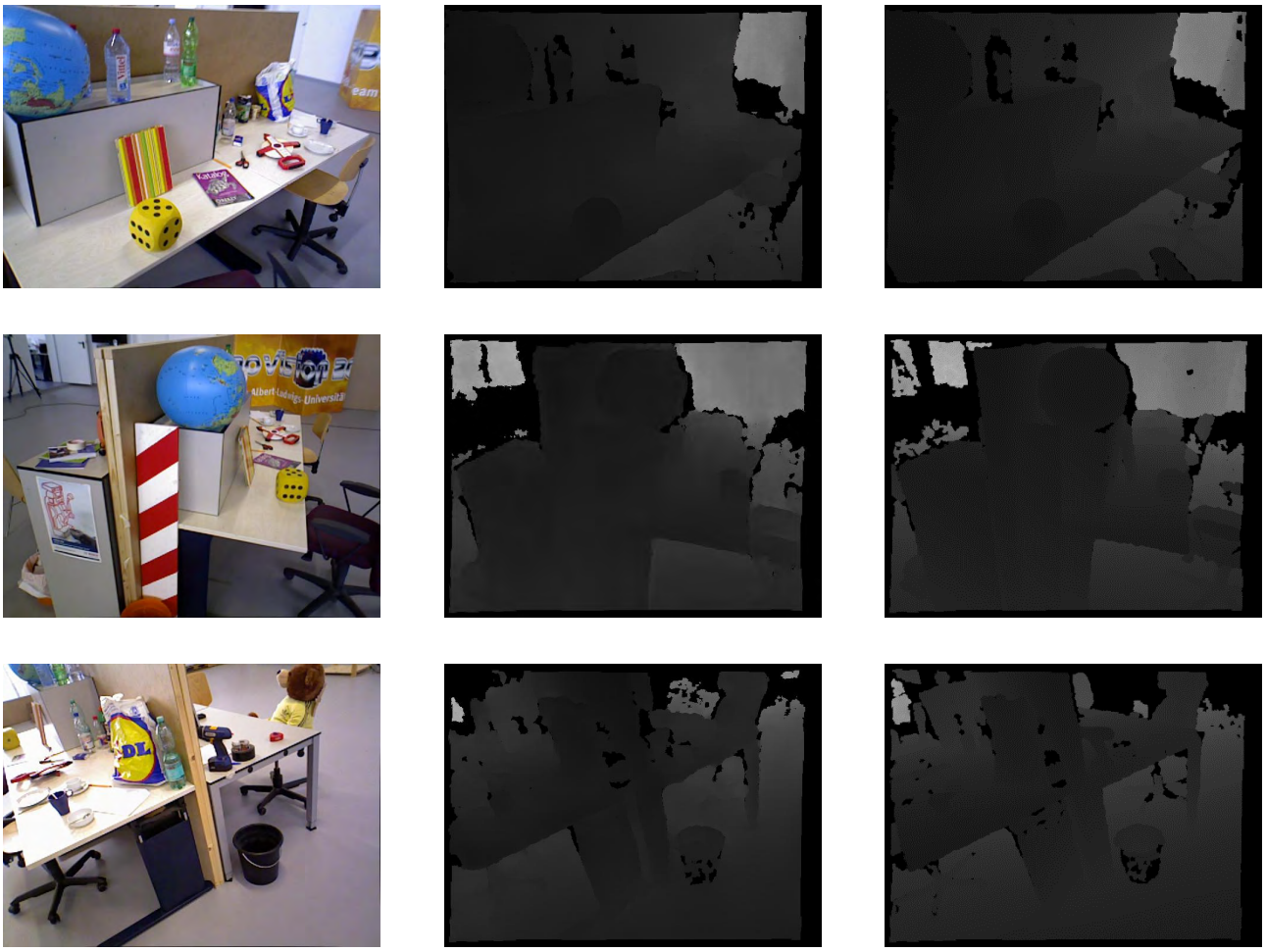
$$\max\left(\frac{d_P^{GT}}{d_P}, \frac{d_P}{d_P^{GT}}\right) < \delta \tag{10}$$

In the formulas above, d_P^{GT} is the ground-truth depth value, d_P is the estimated depth value at pixel P, N is number of pixels in the evaluated images, and N_δ is the number of pixels in the evaluated images satisfying condition (10).

The experiments were performed on a PC equipped with a Nvidia GTX 970 4 GB GPU. For running the GAN model, we used the Ubuntu 16.04 operating system, Cuda 8.0, and CuDNN 5.1. In addition, we employed PyTorch for deep learning library and the Python programming language. We also used minibatch stochastic gradient descent with learning rate 0.0002, momentum parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$, parameter $\mu = 10$ based on the model in [24], and 32 generator and discriminator filters in the first convolutional layer. The training times for the NYU-Depth V2 dataset and the TUM's RGB-D SLAM Dataset and Benchmark were approximately one and three days, respectively. The Kinect sensor parameters were $f_u = 535.4$, $f_v = 539.2$, $c_u = 320.1$, and $c_v = 247.6$. Implementing the method with other depth sensors can require different parameter values.

B. EXPERIMENTAL RESULTS

All experiments produced suitable results on both datasets. Figures 4 and 5 show some reconstructed images from the TUM and NYU datasets, respectively. On both datasets, the depth images were accurately estimated from the corresponding RGB images. The colored 3D point clouds were obtained from the depth and RGB images. In both figures,



(a) (b) (c)

FIGURE 7. Depth images on the TUM's RGB-D SLAM Dataset and Benchmark. (a) Input RGB images; (b) synthesized depth images by the proposed method; (c) ground-truth depth images.

the left, middle, and right columns show the input RGB images, the colored 3D point clouds estimated using the proposed method, and the colored 3D point clouds obtained from ground-truth depth images, respectively. We considered the point clouds on the right columns as ground truth for quantitative evaluation. The resulting 3D point clouds suggest that the proposed method can generate data similar to the ground truth.

To quantitatively evaluate the proposed method, we analyzed both the generated 3D point clouds and depth images. The quality of the depth images is the most important for output data, as we did not have ground-truth 3D data available. Still, for a comprehensive evaluation, we first analyzed the depth images and then the corresponding 3D point clouds by considering the point clouds generated from the ground-truth depth images.

To determine the estimation accuracy of depth images, we compared our results with the depth ground-truth data by calculating both the error and accuracy measures. The evaluation results on the NYU-Depth V2 dataset are listed in Table 1 and include those for the comparison methods. The accuracy was determined at thresholds of 1.25, 1.25^2 and 1.25^3. Table 2 lists the evaluation results for the proposed method on the TUM's RGB-D SLAM Dataset and Benchmark. Overall, the quality of the proposed method is better on the TUM than on the NYU dataset, especially for accuracy at threshold of 1.25, where the TUM retrieves 19% more accurate results than the NYU dataset. This improved accuracy may derive from the larger training data of the TUM dataset and the more varied scenes available on the NYU dataset. Some examples to compare the proposed and other evaluated methods are shown in Figure 6. Comparing the figures and measures in Table 1 shows that the proposed method provides a better reconstruction quality than the other evaluated methods. Figure 7 shows depth images synthesized by the proposed method on the TUM dataset. These images (middle column of the figure) are very similar to the corresponding ground-truth depth images (right column of the figure).

Figure 8 shows performance results of the synthetic 3D point clouds obtained from the proposed method. The blue and red lines show the rate of the test images with RMSE

TABLE 1. Evaluation of the proposed and comparison methods on the NYU-depth V2 dataset.

Method	Error (lower is better)			Accuracy (higher is better)		
	REL	*RMSE*	*L10E*	$ACC_{1.25}$	$ACC_{1.25^2}$	$ACC_{1.25^3}$
Make3D [27]	0.349	1.214	–	0.447	0.745	0.897
Discrete–continuous CRF [29]	0.335	1.06	0.127	–	–	–
DepthTransfer [30]	0.35	1.2	0.131	–	–	–
Ladicky et al. [28]	–	–	–	0.542	0.829	0.941
Eigen et al. [26]	0.215	0.907	–	0.611	0.887	0.971
Liu et al. [31]	0.23	0.824	0.095	0.614	0.883	0.971
Ours	**0.168**	**0.538**	**0.069**	**0.752**	**0.956**	**0.994**

TABLE 2. Evaluation of the proposed method on the TUM'S RGB-D slam dataset and benchmark.

Method	Error (lower is better)			Accuracy (higher is better)		
	REL	*RMSE*	*L10E*	$ACC_{1.25}$	$ACC_{1.25^2}$	$ACC_{1.25^3}$
Ours	0.107	0.548	0.047	0.942	0.984	0.990

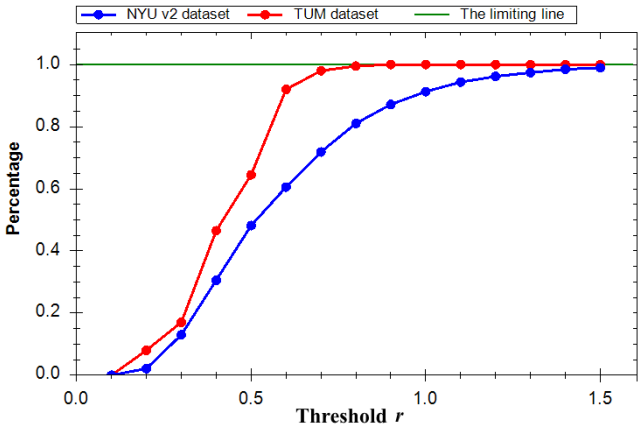


FIGURE 8. Rate of test images with RMSE < r on NYU-Depth V2 Dataset (blue line) and TUM's RGB-D SLAM Dataset and Benchmark (red line).

below threshold r on the NYU and TUM datasets, respectively. The line of the TUM dataset converges to 1 faster than that of the NYU dataset. Overall, the average RMSE on the NYU dataset is 0.585 and that on the TUM dataset is 0.420. Consequently, the proposed method applied on the TUM dataset provides better accuracy than applying it on the NYU dataset. These results are consistent with the quality of the synthetic depth images.

V. DISCUSSION

The experimental results demonstrated that we could generate a high-quality point cloud from a single RGB image. The proposed method can be applied to the diverse kinds of applications such as autonomous robots and remote-controlled systems. For examples, object tracking and 3D scene reconstruction from the point clouds can be performed. In the applications, an expensive laser sensor can be replaced by a low-cost 2D camera, hence the overheads are significantly decreased. In future work, we will modify our GAN model to improve depth estimation and consequently the quality of the resulting 3D point clouds.

VI. CONCLUSIONS

We propose a method to transform a single RGB image of a scene into the corresponding 3D point cloud. As GANs are powerful and flexible tools for generating synthetic data, we implemented a two-stage GAN-based method to generate depth images that are then used to reconstruct point clouds. Experimental results indicate that the proposed method can effectively create a high-quality point cloud from a single RGB image. The comparison of the proposed method to state-of-the-art reconstruction methods shows that our method achieves the best performance by considering visual assessment and quantitative analyses. In addition, our method does not have high PC hardware requirements including high-end GPUs. In fact, we verified the efficient execution of the proposed method on a common PC with a 4 GB GPU. In future work, we will modify the GAN model to improve depth estimation and consequently the quality of the resulting 3D point clouds.

REFERENCES

[1] D. Brščić, T. Kanda, T. Ikeda, and T. Miyashita, "Person tracking in large public spaces using 3-D range sensors," *IEEE Trans. Human-Mach. Syst.*, vol. 43, no. 6, pp. 522–534, Nov. 2013.

[2] J. Ćesić, I. Marković, S. Jurić-Kavelj, and I. Petrović, "Detection and tracking of dynamic objects using 3D laser range sensor on a mobile platform," in *Proc. 11th Int. Conf. Inform. Control, Automat. Robot. (ICINCO)*, Vienna, Austria, Sep. 2014, pp. 1–3.

[3] Y. Ye, L. Fu, and B. Li, "Object detection and tracking using multi-layer laser for autonomous urban driving," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Rio de Janeiro, Brazil, Nov. 2016, pp. 259–264.

[4] Z. Gao *et al.*, "Real-time visual tracking with compact shape and color feature," *Comput., Mater. Continua*, vol. 55, no. 3, pp. 509–521, 2018, doi: 10.3970/cmc.2018.02634.

[5] H. Macher, T. Landes, and P. Grussenmeyer, "From point clouds to building information models: 3D semi-automatic reconstruction of indoors of existing buildings," *Appl. Sci.*, vol. 7, no. 10, p. 1030, 2017.

[6] F. Tsai, T. S. Wu, I. C. Lee, H. Chang, and A. Y. S. Su, "Reconstruction of indoor models using point clouds generated from single-lens reflex cameras and depth images," in *Proc. Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, Tokyo, Japan, May 2015, pp. 99–102.

[7] D. Huber, H. Herman, A. Kelly, P. Rander, and J. Ziglar, "Real-time photo-realistic visualization of 3D environments for enhanced tele-operation of vehicles," in *Proc. IEEE 12th Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Kyoto, Japan, Sep./Oct. 2009, pp. 1518–1525.

[8] A. Kelly *et al.*, "Real-time photorealistic virtualized reality interface for remote mobile robot control," *Int. J. Robot. Res.*, vol. 30, no. 3, pp. 384–404, 2011, doi: 10.1177/0278364910383724.

[9] W. Song, S. Cho, K. Cho, K. Um, C. Won, and S. Sim, "Traversable ground surface segmentation and modeling for real-time mobile mapping," *Int. J. Distrib. Sens. Netw.*, vol. 10, no. 4, pp. 1–8, 2014, doi: 10.1155/2014/795851.

[10] W. Song and K. Cho, "Real-time terrain reconstruction using 3D flag map for point clouds," *Multimedia Tools Appl.*, vol. 74, no. 10, pp. 3459–3475, 2015, doi: 10.1007/s11042-013-1669-4.

[11] P. Chu, S. Cho, S. Fong, Y. Park, and K. Cho, "3D reconstruction framework for multiple remote Robots on cloud system," *Symmetry*, vol. 9, no. 4, p. 55, 2017, doi: 10.3390/sym9040055.

[12] A. Khatamian and H. R. Arabnia, "Survey on 3D surface reconstruction," *J. Inf. Process. Syst.*, vol. 12, no. 3, pp. 338–357, 2016, doi: 10.3745/JIPS.01.0010.

[13] W. Song, L. Liu, Y. Tian, G. Sun, S. Fong, and K. Cho, "A 3D localisation method in indoor environments for virtual reality applications," *Hum. Cent. Comput. Inf. Sci*, vol. 7, p. 391, Oct. 2017, doi: 10.1186/s13673-017-0120-7.

[14] P. M. Chu, S. Cho, S. Sim, K. Kwak, and K. Cho, "Multimedia system for real-time photorealistic nonground modeling of 3D dynamic environment for remote control system," *Symmetry*, vol. 10, no. 4, p. 83, 2018.

[15] M. N. Galabov, "A real time 2D to 3D image conversion techniques," *Int. J. Eng. Sci. Innov. Technol.*, vol. 4, no. 1, pp. 297–304, 2015.

[16] R. M. S. Priya, S. L. Aarthy, C. Gunavathi, P. Venkatesh, K. Srinivas, and G. Xiao-Zhi, "3D reconstruction of a scene from multiple 2D images," *Int. J. Civil Eng. Technol.*, vol. 8, no. 12, pp. 324–331, 2017.

[17] D. C. Lee, M. Hebert, and T. Kanade, "Geometric reasoning for single image structure recovery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 2136–2143.

[18] N. I. A. Abdulqawi and M. S. A. Mansor, "A computer method for generating 3D point cloud from 2D digital image," *J. Image Graph.*, vol. 4, no. 2, pp. 89–92, 2016.

[19] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, Dec. 2014, pp. 1–9.

[20] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," in *Proc. Neural Inf. Process. Syst. (NIPS)*, Barcelona, Spain, Dec. 2016, pp. 82–90.

[21] Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 1–20.

[22] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5967–5976.

[23] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, San Juan, Puerto Rico, May 2016, pp. 1–16.

[24] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. (Nov. 2017). "High-resolution image synthesis and semantic manipulation with conditional GANs." [Online]. Available: https://arxiv.org/abs/1711.11585

[25] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[26] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, Dec. 2014, pp. 1–9.

[27] A. Saxena, M. Sun, and A. Y. Ng, "Make3D: Learning 3D scene structure from a single still image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 824–840, May 2009.

[28] L. Ladicky, J. Shi, and M. Pollefeys, "Pulling things out of perspective," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 89–96.

[29] M. Liu, M. Salzmann, and X. He, "Discrete-continuous depth estimation from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 716–723.

[30] K. Karsch, C. Liu, and S. B. Kang, "Depth transfer: Depth extraction from video using non-parametric sampling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2144–2158, Nov. 2014.

[31] F. Liu, C. Shen, and G. Lin, "Deep convolutional neural fields for depth estimation from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 5126–5170.

[32] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy, Oct. 2012, pp. 746–760.

[33] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vilamoura, Portugal, Oct. 2012, pp. 1–8.

PHUONG MINH CHU received the B.Eng. degree in information technology from Le Quy Don Technical University, Hanoi, Vietnam, in 2011, and the M.Eng. degree in multimedia engineering from Dongguk University, Seoul, South Korea, in 2016, where he is currently pursuing the Ph.D. degree with the Department of Multimedia Engineering. From 2011 to 2014, he was a Research Assistant with the Institute of Simulation Technology, Le Quy Don Technical University. His current interests are focused on 3D modeling, artificial intelligence, and human–computer interaction.

YUNSICK SUNG received the B.S. degree in the Division of electrical and computer engineering, Pusan National University, Busan, South Korea, in 2004, and the M.S. degree in computer engineering and the Ph.D. degree in game engineering from Dongguk University, Seoul, South Korea, in 2006 and 2012, respectively. From 2006 to 2009, he was a Researcher with Samsung Electronics, South Korea. From 2012 and 2013, he was a Post-Doctoral Fellow with the University of Florida, FL, USA. He is currently an Assistant Professor with the Department of Multimedia Engineering, Dongguk University. His research interests are focused on the areas of superintelligence in the fields of immersive software, UAVs, and robots.

Dr. Sung has been a Managing Editor of *Human-centric Computing and Information Sciences*, since 2015, and the *Journal of Information Processing Systems*, since 2017. He was a Guest Editor in one of special issues of *Symmetry*, in 2017 and one of the special issues of the *Journal of Ambient Intelligence and Humanized Computing*, in 2018. He has a lot of conference service experience at the International Conference on Multimedia and Ubiquitous Engineering, International Conference on Future Information Technology, World Congress on Information Technology Applications and Services, Global Conference on Information Technology, Computing, and Applications, International Conference on Big data, IoT, and Cloud Computing, International Conference on Parallel and Distributed Computing Applications, and Technologies, International Conference on Ubiquitous Computing Application and Wireless Sensor Network, International Conference on Ubiquitous Information Technologies and Applications, and International Conference on Computer Science and its Applications.

KYUNGEUN CHO (M'12) received the B.Eng. degree in computer science and the M.Eng. and Dr.Eng. degrees in computer engineering from Dongguk University, Seoul, South Korea, in 1993, 1995 and 2001, respectively. From 1997 to 1998, she was a Research Assistant with the Institute for Social Medicine, Regensburg University, Germany, and a Visiting Researcher with the FORWISS Institute, Technical University of Munich, Germany. She has been a Full Professor with the Department of Multimedia Engineering, Dongguk University, since 2003. Her current research interests are focused on the areas of intelligence of robot and virtual characters and real-time computer graphics technologies. She has led a number of projects on robotics and game engines and also has published many technical papers in these areas.