A Synthetic Depth Map Creation Mechanism based on RGB images for Object Distance Estimation

Jaehong Kim
Department of Computer Engineering
Dongguk University
Seoul, Republic of Korea
imjahk@dgu.ac.kr

Misbah Bibi
Department of Computer Engineering
Jeju National University
Jeju, South Korea
misbahbibi@stu.jejunu.ac.kr

Muhammad Faseeh
Department of Computer Engineering
Jeju National University
Jeju, Republic of Korea
faseeh@stu.jejunu.ac.kr

***Abstract*— The scarcity of annotated data remains a critical bottleneck in training robust computer vision models, particularly for depth-aware tasks such as object detection and object distance estimation. Manual annotation is both labor-intensive and error-prone, often leading to inconsistencies in ground truth data. To address this challenge, we present a synthetic depth maps creation mechanism based on procedurally generated RGB images. The proposed approach simulates diverse scenes by varying object placement, illumination, and camera parameters to produce photorealistic RGB data along with corresponding depth and distance maps. All generated outputs are stored in HDF5 format to ensure efficient data handling and large-scale usability. This mechanism enables the automatic generation of high-quality annotations, significantly reducing manual effort. Experimental evaluations demonstrate that models trained on the resulting synthetic dataset achieve strong performance in object detection and distance estimation tasks, highlighting the effectiveness of the proposed method in supporting depth-aware computer vision applications.**

Keywords— Synthetic data generation, BlenderProc, computer vision, object detection, depth estimation, annotation automation.

I. INTRODUCTION

Recent advancements in computer vision have enabled significant progress across various domains, including autonomous driving, surveillance, and robotics. These achievements are largely attributed to the availability of large-scale annotated datasets and the growing capabilities of deep learning models [1]. However, acquiring high quality annotations particularly for depth aware tasks such as object detection and distance estimation remains a considerable challenge. Manual annotation is not only time consuming and expensive, but it also often lacks the precision necessary for accurate 3D spatial understanding [2].

To overcome these limitations, synthetic data has emerged as a promising alternative for generating large volumes of annotated data efficiently and consistently. By simulating realistic environments and automatically producing ground truth labels, synthetic data provides a scalable solution to the data bottleneck in supervised learning [3], [10]. Among the tools enabling such capabilities, BlenderProc [4] a procedural pipeline built on the Blender 3D engine offers flexible scene generation, object placement, camera configuration, and lighting control. These features make it particularly well suited for generating photorealistic datasets with pixel level annotations.

In this paper, we present a synthetic data generation pipeline based on BlenderProc to produce annotated RGB images, depth maps, and distance maps, all stored efficiently in HDF 5 format. The pipeline is fully automated through Python scripting, enabling scalable and consistent dataset creation while minimizing manual effort. We validate the effectiveness of the generated data by applying it to depth aware tasks such as object detection and distance estimation, demonstrating the practical utility of our approach for training vision models in 3D understanding applications.

II. LITERATURE REVIEW

The reliance of deep learning models on large annotated datasets has fueled interest in synthetic data as a scalable, cost-effective alternative to manual data collection. Synthetic datasets enable automated labeling and offer full control over scene parameters. Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

BlenderProc, introduced by Denninger et al. [5], is a procedural framework built on Blender that supports photorealistic rendering, scene customization, and physics simulation. It produces RGB images, depth maps, and 3D annotations, making it well-suited for vision tasks in robotics and AR/VR. Its Python-based automation has contributed to its broad adoption.

Building on this direction, PhysXGen by Kopicki et al. [6] introduced physics-aware interactions for robotics. Several works [7], [8], [11] explored synthetic-to-real domain adaptation for depth prediction, showing improved generalization. Similarly, DexYCB++, developed by Kar et al. and Uddin et al. [9], [12], offers high-fidelity synthetic data with 6DoF pose and depth information for hand-object interaction.

Our work builds on these foundations with a BlenderProc-based pipeline featuring enhanced scene variability, depth and

distance map generation, and efficient HDF5 storage optimized for depth-aware object detection with minimal human intervention.

III. Proposed Methodology

To overcome the limitations of manually annotated datasets in depth aware object detection, we developed a synthetic data generation pipeline using BlenderProc, a procedural framework built on the Blender 3D engine. The pipeline is tailored for cluttered environments and simulates realistic object interactions commonly encountered in surveillance and robotic vision applications.

Our implementation focuses on generating synthetic scenes populated with objects such as bottles, cups, and Blender's native Suzanne monkey head. These are procedurally arranged to introduce occlusions, overlaps, stacking, and varied orientations mimicking complex indoor environments. Using this application, we generate depth maps and distance maps for both indoor and outdoor scenarios, enabling model training across diverse spatial contexts. Each scene is procedurally generated with randomized object placement, material textures, and environmental attributes to increase data variability.

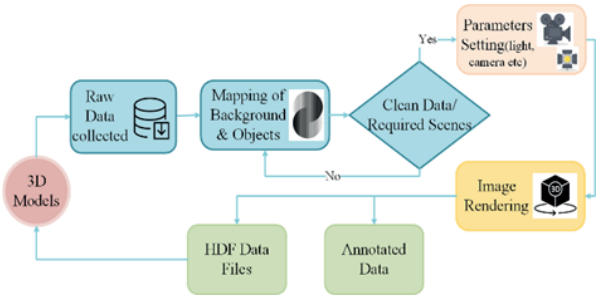

Figure 1: Process flow of proposed synthetic data generation pipeline.

Figure 1 illustrates the overall workflow of the proposed synthetic data generation pipeline. The process begins with the import of 3D object models, which are used to construct virtual scenes. Raw data, including object and background information, is collected and mapped to create diverse environments. A conditional check ensures the scenes meet the required criteria (e.g., occlusion, realism); if satisfied, parameters such as lighting and camera settings are configured. The finalized scenes undergo image rendering to produce RGB images, while corresponding annotations such as depth maps, distance maps, and bounding boxes are automatically generated. All outputs are organized and stored efficiently in HDF5 format, enabling seamless access to annotated data for training depth aware object detection models.

Camera positions are dynamically generated using a spherical pose sampling method, where the virtual camera is placed at random or strategic points on the surface of a sphere surrounding the scene. This allows variation in distance, angle, and elevation, simulating a wide range of real-world viewpoints. By capturing scenes from different perspectives including oblique angles and partially occluded views the dataset reflects realistic visual challenges commonly encountered in environments such as indoor navigation, robotics, and surveillance. This variability is essential for training robust object detection and depth estimation models, as it improves generalization and enables the models to perform effectively across diverse and unpredictable conditions.

As part of the pipeline, a monocular RGB camera is simulated with real-world intrinsic parameters such as focal length, principal point, and resolution to closely replicate actual imaging sensors. This realistic configuration improves the domain transferability of models trained on synthetic data, making them more effective when deployed in real-world scenarios involving varied object layouts, lighting conditions, and occlusions.

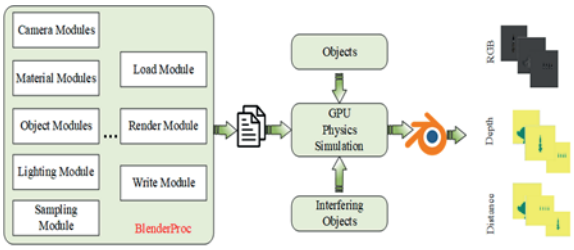

Figure 2: Workflow architecture of the proposed synthetic data generation pipeline using BlenderProc.

Figure 2 presents the core workflow of the proposed synthetic data generation process using the BlenderProc framework. The pipeline begins with a set of modular components, including camera, material, object, lighting, and sampling modules, which are configured to define scene parameters. These modules are orchestrated by the load, render, and write modules, enabling the automated setup and execution of synthetic scene generation.

Configured scenes are then processed through a GPU based physics simulation engine, which ensures realistic interactions between target objects and interfering objects. This simulation introduces naturalistic behaviors such as object stacking, collisions, and occlusions critical for generating complex and realistic environments.

Following the simulation, Blender is used to render the scenes based on the defined parameters. The rendering process produces three key outputs: RGB images for visual input, depth maps for pixel-level spatial information, and distance maps indicating the distance from the camera to object centroids. These outputs form the multimodal synthetic dataset used to train depth-aware object detection and estimation models.

For each rendered frame, the proposed BlenderProc-based pipeline generates multiple data modalities essential for training depth-aware vision models. These include RGB images, which serve as the primary input for object detection by capturing visual appearance and texture information. In addition, depth maps are produced to provide pixel-level distance measurements from the camera to scene surfaces, enabling supervised training of monocular depth estimation models. Distance maps derived from the depth maps quantify the Euclidean distance between the camera and the centroid of each object, supporting spatial reasoning tasks such as object localization. Furthermore, bounding box annotations are automatically generated and stored in JSON format, ensuring consistent and accurate labeling without manual intervention.

To manage this large-scale, multimodal dataset efficiently, we employ the Hierarchical Data Format version 5 (HDF5). This format enables the compact storage of diverse data types including RGB images, depth maps, distance maps, and annotations within a unified file structure, significantly reducing disk space requirements. HDF5's high-speed input/output capabilities facilitate fast data loading during training, thereby improving model development time. Moreover, its hierarchical organization simplifies access to and integration of various data modalities, making it particularly well suited for deep learning pipelines involving complex vision tasks.

IV. Results

The performance of the proposed synthetic data generation pipeline is evaluated qualitatively through visual inspection of the rendered outputs. As shown in Figure 3 and 4, the generated RGB images accurately simulate indoor environments, exhibiting realistic textures, diverse object arrangements, and consistent lighting conditions. These scenes capture complex spatial interactions such as occlusion, overlap, and stacking, which are essential for training robust object detection models.

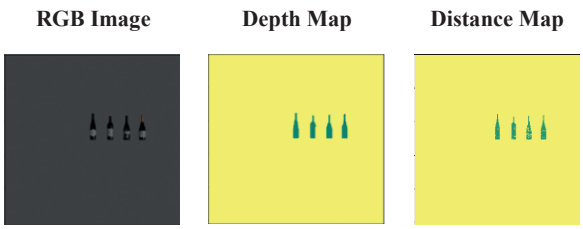

Figure 3: Illustration of process of creating synthetic data for object detection including RGB images, depth map and distance map.

The corresponding depth maps provide detailed, pixel-level geometric information, enabling precise supervision for monocular depth estimation tasks. Additionally, distance maps, computed from the depth data and 3D object centroids, offer scalar distance annotations between the camera and each object. These annotations are particularly valuable for spatial reasoning tasks in robotics and surveillance applications.

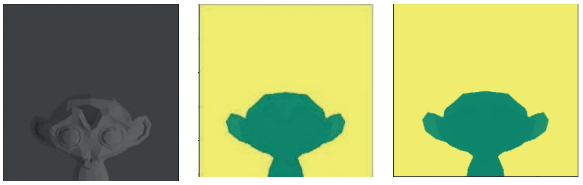

Figure 4: Genenration of RGB images,depth maps, and distance maps for indoor object scenes using the proposed synthetic data pipeline.

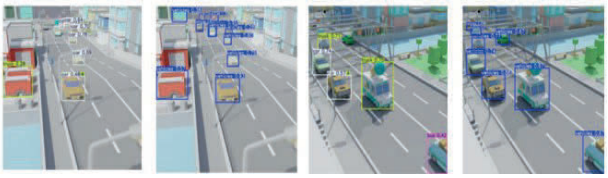

Figure 5: Visulaization of object detection results on synthetic data, demonstrating accurate detection and identification of vehicles at varying distances.

To further assess the pipeline's flexibility and generalization capability, we extended the data generation process to include outdoor environments. As illustrated in Figure 5 and 6, the synthetic outdoor scenes feature natural backgrounds, varied terrain, and lighting conditions that reflect real-world variability. The RGB images, along with their corresponding depth and distance maps, demonstrate the pipeline's effectiveness in generating high-quality multi-modal data beyond indoor settings.

Figure 6: Synthetic generation of outdoor scene data, including RGB images, depth information, and object distance mapping.

Figure 7: Synthetic generation of outdoor scene data, including RGB imagery, depth information, and object

distance mapping.

The consistent quality of outputs across both indoor and outdoor scenarios confirms the capability of our BlenderProc-based pipeline to produce scalable, diverse, and richly annotated datasets. These results validate the pipeline's applicability for training deep learning models in tasks involving object detection, depth estimation, and spatial awareness across a range of environments.

V. Conclusion

This work presents a robust and scalable synthetic data generation pipeline tailored for object detection and object distance estimation across both indoor and outdoor environments. The proposed mechanism simulates a monocular RGB camera setup and incorporates diverse lighting conditions, realistic textures, and varied object arrangements to create high-quality RGB images, depth maps, and distance annotations. The inclusion of automated 2D bounding box generation and structured HDF5 storage ensures efficient annotation, scalability, and streamlined data management.

The resulting multi-modal synthetic dataset supports the training of deep learning models that require both visual recognition and spatial understanding, significantly reducing dependence on manual annotation. Qualitative results validate the effectiveness of the approach in generating consistent and diverse data suitable for depth-aware vision applications in fields such as surveillance, robotics, and autonomous systems.

Future research will focus on integrating the synthetic dataset into real-world training pipelines, evaluating its cross-domain generalization performance, and extending the framework to support additional vision tasks such as instance segmentation and 6DoF pose estimation.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT)(No. RS-2024-00346238) and this work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT)(No. RS-2024-00423362). . Any correspondence related to this paper should be addressed to DoHyeun Kim.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in Proc. 25th Int. Conf. Neural Information Processing Systems (NeurIPS), Lake Tahoe, NV, USA, Dec. 2012, pp. 1097–1105.

[2] T.-Y. Lin, M. Maire, S. Belongie, et al., "Microsoft COCO: Common objects in context," in Proc. Eur. Conf. Comput. Vis. (ECCV), Zurich, Switzerland, Sep. 2014, pp. 740–755.

[3] J. Tremblay, A. Prakash, D. Acuna, et al., "Training deep networks with synthetic data," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Salt Lake City, UT, USA, Jun. 2018, pp. 969–977.

[4] M. Denninger, et al., "BlenderProc: Reducing the reality gap by randomizing synthetic images for domain adaptation," arXiv preprint arXiv:2006.02771, 2020.

[5] M. Kopicki, M. Do, S. R. Ahmadzadeh, et al., "PhysXGen: Physics-aware synthetic data generation for generalizable manipulation," in Proc. Robotics: Science and Systems (RSS), 2021.

[6] S. Zhang, Y. Zhang, Y. Li, and H. Zhao, "Synthetic-to-real self-supervised robust depth estimation via learning with motion and structure priors," arXiv preprint arXiv:2503.20211, 2025.

[7] A. Kar, C. Sweeney, H. Liu, et al., "DexYCB: A benchmark for capturing hand grasping of objects," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 9044–9053.

[8] Y. Jung and S. Seo, "Synthetic image generation using 3D rendering software and object detection research: Focused on pine wilt disease," Digital Arts Engineering Multimedia Paper, vol. –, pp. 51–60, 2024.

[9] M.-S. Jang and Y.-H. Ha, "3D rendering and deep learning-based mine burial rate measurement," J. Korea Academia-Industrial Cooperation Society, vol. 24, no. 12, pp. 244–252, 2023, doi: 10.5762/KAIS.2023.24.12.244.

[10] Y. Jung and Y. H. Jung, "Research on the utilization of 3D virtual synthetic data for enhancing deep learning model performance: Focused on the detection of pine wilt disease damaged trees," unpublished, 2024.

[11] M. A. Uddin, M. N. Ahsan, and M. Das, "A comparative study on various ML models using synthetic data for privacy preservation," in Proc. 4th Interdisciplinary Conf. Electrics and Computer (INTCEC), 2024, pp. 1–6. doi: 10.1109/INTCEC61833.2024.10602971.

[12] M. Bibi, A. N. Khan, M. Faseeh, Q. W. Khan, R. Ahmad, and D.-H. Kim, "A synthetic data generation approach with dynamic camera poses for long-range object detection in AI applications," IEEE Access, vol. 12, pp. 194505–194520, Dec. 2024, doi: 10.1109/ACCESS.2024.3517717.

[13] P. Gutierrez, M. Luschkova, A. Cordier, M. Shukor, M. Schappert, and T. Dahmen, "Synthetic training data generation for deep learning-based quality inspection," in Proc. 15th Int. Conf. Quality Control Artif. Vis., Jul. 2021, pp. 9–16.

[14] T. Kikuchi, T. Fukuda, and N. Yabuki, "Development of a synthetic dataset generation method for deep learning of real urban landscapes using a 3D model of a non-existing realistic city," Adv. Eng. Informatics, vol. 58, Art. no. 102154, Oct. 2023.