Virtual Experience Toolkit: Enhancing 3D Scene Virtualization from Real Environments through Computer Vision and Deep Learning Techniques

1st Clara Garcia
Human-Tech
Universitat Politècnica de València
Valencia,Spain
cgarmol1@htech.upv.es

2nd Pau Mora
Human-Tech
Universitat Politècnica de València
Valencia, Spain
pamogar@htech.upv.es

3rd Mario Ortega
Human-Tech
Universitat Politècnica de València
Valencia, Spain
mortega@htech.upv.es

4th Eugenio Ivorra
Human-Tech
Universitat Politècnica de València
Valencia, Spain
euivmar@htech.upv.es

5th Gaetano Valenza
NeuroCardiovascular Intelligence Lab
Bioengineering and Robotics
Research Center "E. Piaggio"
Pisa, Italy
gaetano.valenza@unipi.it

6th Mariano L. Alcañiz
Human-Tech
Universitat Politècnica de València
Valencia, Spain
malcaniz@htech.upv.es

***Abstract*—Over the past few decades, Virtual Reality (VR) has emerged as a popular topic in a wide range of fields, such as information technology and psychology, among others. One reason for its importance was due to the ability to virtualize real-world scenes. This process typically involves capturing data from those scenes to generate accurate, detailed, and immersive 3D models. However, the creation of virtual content from real-world scenes has traditionally relied on manual techniques, as well as photogrammetry or Computer Vision (CV) algorithms. This frequently yields time-intensive, less accurate, intricate, and semi-automatic results. To tackle these limitations, a novel framework called Virtual Experience Toolkit (VET) has been proposed. It employs CV and Deep Learning (DL) techniques to swiftly and seamlessly virtualize any 3D scenario from real indoor environments. To demonstrate the effectiveness of VET, a diverse dataset of virtualized 3D scenes was generated, supplementing the information from the ScanNet dataset. VET has the potential to significantly enhance the virtualization of 3D indoor scenarios from real scenes, making the process easier, more precise, unified, consistent, automated, and effective for a broad spectrum of VR applications.**

***Index Terms*—3D Scene Understanding, Indoor Scenes, Virtual Reality (VR), ScanNet, Scene Reconstruction**

I. Introduction

The growing fascination with VR, coupled with advances in CV and graphic technologies while increasing the availability of affordable hardware devices, has expanded the range of applications for VR, including areas such as psychology, gaming, medicine, and education, further contributing to its significance in various fields [1].

The research leading to these results has received partial funding from the European Commission under grant agreement N. 101017727 for the project EXPERIENCE.

One of the key components of VR is the ability to create realistic and interactive 3D scenes; this is one example where 3D scene virtualization becomes highly valuable. This refers to transforming real-world environments into accurate digital representations, offering immersive simulations. This technology finds applications beyond VR, such as gaming and architecture [6], due to its ability to simulate 3D spaces, providing realistic, detailed, and interactive experiences in digital environments. In particular, its main advantages include creating a faithful representation of the real-world environment and cost savings compared to physical prototyping.

Traditionally, 3D scene virtualization had been done by a team of graphic designers, manually creating and inserting each of the 3D objects into the designed 3D virtual scene. This solution is time-consuming and requires highly detailed information about the scene, such as images from different points of view and measures of the whole scene [4], [9]. In contrast, due to advances in CV and DL techniques, 3D scene virtualization became semi-automatic, using inputs such as text, 2D images, and more recently, sets of RGB-D images captured with cameras such as the Intel Realsense D4XX family. These new methods perform each of the steps separately, and each of these are performed either automatically, semi-automatically, or manually.

Therefore, integrating each step into the same workflow is complex, hindering the usability of this type of tool. As mentioned, the whole process requires different stages to perform a 3D scene completely immersive, interactive, and similar to the real-word. Specifically, it needs to perform a 3D reconstruction and then apply a 3D scene understanding process. Traditionally, photogrammetry was employed for scene reconstruction, but this method tends to be costly in terms of time, mainly as

it necessitates highly textured images to yield precise results. Alternatively, accurate and fast methods to reconstruct 3D information have emerged based on Dense SLAM techniques like BundleFusion [2]. Regarding the 3D scene understanding step, it was commonly carried out manually. However, since last years, this process has been done using CV and DL techniques that allow the system to identify, classify, semantically segment and represent the different classes of objects in the scene. Specifically by using the most similar CAD models in the same position, orientation, and scale. Nevertheless, there are still some limitations related to scalability, automation, and integration in the same workflow [8].

To tackle the mentioned challenges, we propose an end-to-end framework called VET to carry out the 3D virtualization of the scene, starting with the scene capture and automatically performing all the necessary steps until the virtualized scene is obtained. Concretely, it performs a 3D reconstruction, then applies a 3D scene understanding step, and finally, the information obtained is integrated into a digital scene. The proposed solution is supported by our own dataset (consisting of various scenes such as bathrooms, bedrooms, kitchens, conference rooms and offices among others) and ScanNet [11] to prove its effectiveness and precision. In summary, the main contributions proposed in this work are:

- A fully automatic and user-friendly framework integrated into a single graphical application developed in C++, Python, and Unity3D, that uses CV and DL techniques, which is adaptable for any user profile.
- Our framework's broad applicability across a wide range of indoor scenes, owing to its ability to work with an extensive array of classes, particularly 200 classes.
- An accurate solution that integrates most of the current state of the art methods for each pipeline step, like Mask3D [17] for instance detection or ScanNotate [19] for CAD retrieval and pose estimation.
- A dataset that complements and it is similar to ScanNet dataset [11]. Composed of RGB-D images, camera pose information, the 3D reconstructed scenes, and the 3D scene understanding results. The indoor environments in size and type. This dataset is open source and available at **https://github.com/Pamogar/VET-IndoorDataset**.
- A qualitative validation process using ScanNet dataset and our own dataset.

The remainder of this paper is organized as follows. Section II briefly reviews existing 3D scene virtualization methods according to the input and methods required to perform the reconstruction. Section III presents the proposed framework. Section IV introduces the results obtained, the discussion about them, and a brief comparison with other known methods. Finally, Section V includes the overall conclusions and some suggestions for future work.

II. State of the Art

In recent years, 3D scene virtualization has gained widespread attention due to increased automation [3]. Several solutions have been proposed for 3D scene generation, with varying approaches depending on the input data and the techniques used to perform the virtualization.

Many automatic 3D scene generation solutions use text or voice as input, benefiting non-graphic domain users. For instance, Seversky et al. [4] present a system that processes the input description by a speech tagger extracting remarkable information and keywords to locate the objects using spatial relation. These methods have some drawbacks, firstly, with the usage of two engines: a language engine and a graphics engine, the process becomes highly time-consuming. And secondly, textual descriptions can be ambiguous, and translating them accurately into a 3D scene can be challenging.

To remove the need for a language engine, and thanks to the advances in CV and DL, Vouzounaras et al. [5] proposed a method based on CV approaches that use 2D images as input for the 3D reconstruction. Specifically, they used the information about vanishing lines while removing perspective distortion to produce the 3D reconstruction. The main drawback is the lack of 3D scene understanding step to perform the 3D virtualization. Moreover, Marullo et al. [6] presented another method based on CV techniques that use Google Cloud Vision API to extract the context and object information, and it retrieves the most similar CAD model to each instance object of the 3D scene using a Models database. The main problems of using 2D images as input are the limitation on the viewpoint, the ambiguity due to the lack of spatial and texture information, and the issues with occlusions. As a result, the obtained 3D scene needs to be completed.

Different methods that use 3D information appeared, precisely RGB-D information. In this field, two main groups are observed depending on the nature of the environment: 3D virtualization of outdoor and indoor scenes. For outdoor scenes, Yang et al. [7] propose a system that obtains the 3D virtualization of the environment for a VR application using CV techniques. However, it is limited to the detection of walkable areas and obstacles, therefore, it does not perform instance detection or CAD alignments.

Regarding indoor scenes, Li et al. [8] propose a solution that generates indoor BIM reconstruction automatically implementing DL algorithms. However, this method only processes five different classes: floor, ceiling, walls, doors, and windows, limiting the virtualization. Similarly, the VRFromX method [9] creates the virtual environment using CV solutions, following this pipeline: scan the real world and then replace the 3D instances detected with CAD models. However, in order to detect the object, users must select the ROI where it is located. Thus, the method is not fully automatic. Finally, another semi-automatic method is presented by Moro et al. [10] to avoid obstacles in an indoor scene. To achieve this, they scan the environment and then estimate the layout using CV techniques.

In summary, while many different methods have been presented, almost all of them are either not fully automatic [6], [9] or not end-to-end 3D virtualization frameworks [5], [10]. Analyzing the solutions previously disclosed, we propose a framework that tackles these limitations, which will be detailed in the next section.

III. METHOD

In this section, we deeply explain the different stages that our proposed framework, VET, carries out to achieve the 3D virtualization of real indoor scenes. The complete pipeline is split into different parts done sequentially (Figure 1).

A. 3D Reconstruction

3D reconstruction is the first critical step in our 3D virtualization pipeline, which uses RGB-D images to precisely replicate real-world scenes in three dimensions.

VET utilizes an adapted version of BundleFusion [2], a technique originating from Dense SLAM methodologies. This approach has proven its robustness, accuracy, and effectiveness in creating real-time color 3D reconstructions of indoor spaces using an RGB-D camera. Through close-loop and bundle adjustment techniques, BundleFusion ensures global consistency, refines camera poses, and improves overall reconstruction quality. It is a reliable and powerful approach for creating high-quality real-time 3D reconstructions of indoor environments. Additionally, VET integrates a volumetric fusion method based on the recent work of Dong et al. [12], which is more precise than BundleFusion technique.

Furthermore, VET demonstrates substantial versatility, accommodating the majority of RGB-D cameras currently available on the market, including the Intel Realsense D435 and D415, and the ZED2i cameras used for our own dataset, and the Structure Sensor used in ScanNet.

Once the 3D reconstruction is obtained, some post-processing of the scene is carried out. This step is executed in order to reduce the computational cost, clean the resulting 3D reconstruction, and align it to ScanNet's coordinate system. First, it was employed an algorithm known as Quadric Decimation [13] to considerably reduce the number of polygons from the initial reconstruction in order of a 100. Lately, a method to remove the artifacts created due to the limitations of the sensor [14] was introduced. Specifically, it determines the number of connected components and removes those small clusters. Finally, the obtained 3D scene is aligned to the same coordinate system as ScanNet [11] automatically.

B. 3D Scene Understanding

As mentioned in previous sections, once the 3D reconstruction is obtained, the result requires some processing to identify the instance objects in the scene and then align this to the most similar CAD model. VET applies this process to achieve the information required to virtualize the information in a digital scene. Therefore, an indoor scene understanding approach is carried out to automatically comprehend the information presented in the scanned scene. In particular, two processes are done in parallel: on the one hand, semantic segmentation and layout estimation, and on the other hand, instance segmentation, CAD retrieval, and alignment, as indicated in Figure 1.

First, it is performed the semantic segmentation of the scene to obtain the information used to compute the layout. The main goal of this step is to label the 3D reconstructed scene, splitting it into different regions based on the semantic classes. In order to carry out this, we select O-CNN (Octree-based Convolutional Neural Networks) [16] because this approach has proved its efficiency and precision when using ScanNet dataset. After applying the inference of the pre-trained model with ScanNet dataset, the segmented 3D scene is obtained. For this specific case, it is used the classes in ScanNet dataset because it already includes classes such as wall, floor, cabinet, door, and windows, which are the labels required to perform the layout assessment step. Therefore, the scene is filtered to obtain a reduced version that only contains those classes.

Layout estimation is another crucial phase because it provides information about where are the limits in the room while delineating the 3D planes and the corners of the scene obtained from the intersection of the planes. Currently, most of the methods in the literature are based on RANSAC for plane detection [8], [10]. Despite the popularity of RANSAC in this field, a novel method called Robust Statistics-based Plane Detection (RSPD) [15] is used in this work due to the improvement in computational time, the reduced number of initial constraints needed and the precision obtained. In particular, the planes are detected on the filtered point cloud mentioned above. Once the planes are obtained, these are filtered depending on their normal direction, and finally, the selected planes and corners are obtained to define the layout.

While performing the previous stages, the instance segmentation is also carried out. This process involves identifying and labeling the instances of the various classes in a 3D scene. Despite the fact that different methods exist, it was selected one of the state of the art solutions for the ScanNet dataset. This approach, called Mask3D [17], automatically segments each instance in the scene, obtaining accurate results for both datasets, ScanNet and ScanNet200 [18]. Specifically, this method consists of a feature backbone, a transformer decoder that utilizes mask modules, and transformer decoder layers to refine queries. Therefore, after applying the inference of the pre-trained model with ScanNet200 dataset, the result is a 3D reconstructed scene with the different instances labeled.

Once the instance segmentation is carried out, it is required to automatically and precisely replace the instances of the different objects in the 3D scene with CAD models in the same position, orientation, and scale as real-world objects. This process is done to be able to modify the shape, appearance, and spatial location of the elements in the scene while reducing the polygonal load of the scene. To do so, VET incorporated the recent method proposed by Ainetter et al. [19] known as ScanNotate, because it replaces the objects detected in the scene with semantically and geometrically similar CAD models. ScanNotate first estimate the 7-DoF pose of the objects, and then it retrieves the closest matching CAD by comparing the labeled object with the CAD models of the same class. In addition, this method joins objects of the same class into clusters in order to assign the same CAD to all the objects in one cluster, and finally, it applies a refinement step to make the results more precise.

One of the limitations found in ScanNotate is the usage of a reduced number of indoor classes of CAD model, in particular,

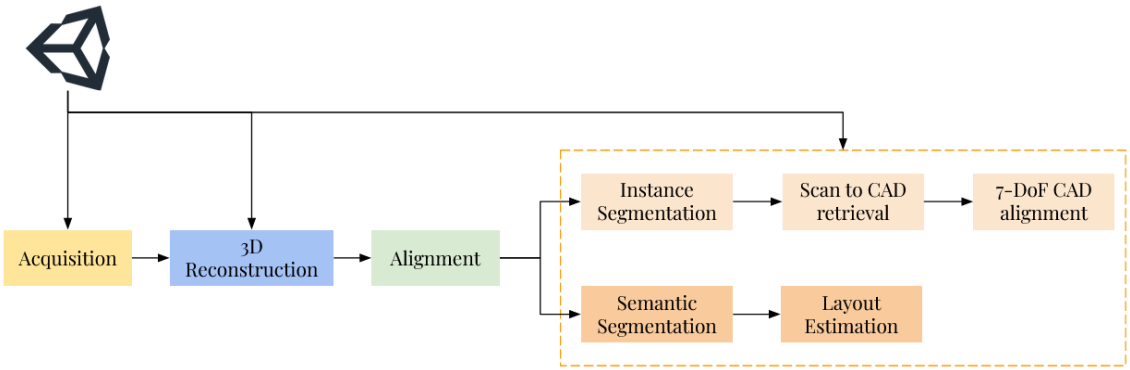


Fig. 1: Workflow of the proposed approach.

ShapeNet [20] dataset. To solve this issue, we complement ShapeNet with some classes of ModelNet [21] that could be detected by the instance segmentation method. For instance, toilet and range hood, among others. Additionally, simple object classes located at the walls, such as doors, windows or pictures, are substituted using generic models, and placing them in the same planes as the wall.

Finally, VET is able to virtualize the 3D scene using the layout information and the best CAD models aligned for each element in the scene.

C. Integration

VET was developed using Unity3D (Version 2020.3.39f1). It has a GUI (Figure 2) to guide the user during the whole pipeline. At the same time, it is responsible for executing in background all the different C++ and Python processes configured to work automatically using a single configuration file. In addition, the different processes are performed sequentially apart from the 3D scene understanding, where two stages are carried out in parallel.

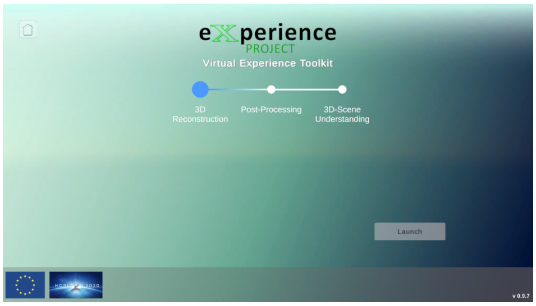


Fig. 2: GUI created for the VET framework.

IV. Results

In analyzing the advantages and disadvantages of the proposed framework concerning other frameworks, we identify that VET is highly scalable, fully integrated, automatic, and precise. This work's accuracy was measured quantitatively by reviewing the results of the different methods used during the whole pipeline evaluated on the ScanNet dataset. At the same time, different qualitative results will also be introduced to strengthen the numerical results and compare them with other methods' performances (Table I).

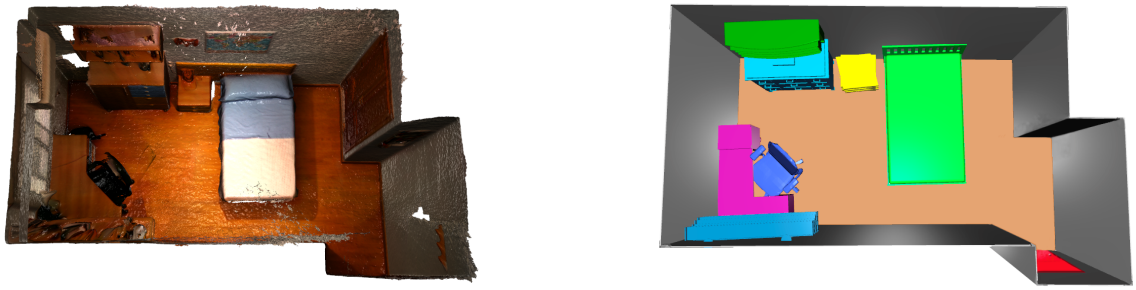
TABLE I: Comparison of the features of different virtualization methods.

Frameworkds	VRFromX [9]	Automatic BIM [8]	VET
Number of classes	40	5	200
Automatic	Semi	Fully	Fully
Virtualization	No	No	Yes
3D Reconstruction	No	Yes	Yes

To prove the achievements in terms of precision, the framework was tested by creating a dataset of 35 different instances acquired from the Universitat Politècnica de Valencia (UPV), which contains rooms like living rooms, meeting rooms, and bathrooms, among others. In particular, it comprises RGB-D images, camera pose information, 3D scene reconstructions, and the results of the 3D scene understanding process.

In order to present the qualitative results, the performances of all the different stages will be depicted. First, it was done a 3D Reconstruction illustrated in Figure 3a. Concretely, this scene contains different objects of different classes that must be substituted by the most similar CAD model in the same position, orientation, and scale.

From the 3D reconstructed scene, it was applied the segmentation method (O-CNN) to obtain the information required to carry out the layout estimation. In particular, O-CNN approach obtains a $mIoU$ (mean Intersection over Union) of 0.762 on ScanNet dataset, which is currently one of the top methods for semantic segmentation tasks, obtaining a better score than Fully Convolutional Networks (FCN) used in [8]. This result is coherent to the visual ones illustrated in Figure 4 where almost all the different objects are well segmented. Furthermore, the quantitative results obtained for the classes used to create the layout are presented in Table II. Specifically, the classes that mostly compose the layout, wall, and floor are the ones that obtain higher results. For instance, in Figure 4, the window is not correctly segmented, but it is detected as a wall, so it will not have a heavy impact on the layout performance.



(a) 3D Reconstruction.

(b) 3D virtualization obtained by VET framework.

Fig. 3: An example of the 3D reconstruction (a) and virtualization (b) of a scene from our own dataset.

TABLE II: mIoU for layout classes evaluated on ScanNet dataset [11].

Classes	Wall	Floor	Cabinet	Door	Window
mIoU	0.868	0.958	0.770	0.640	0.744

Moreover, Figure 4 also depicts the results obtained in the layout estimation process, where it is estimated correctly, as the different planes and corners are well detected.

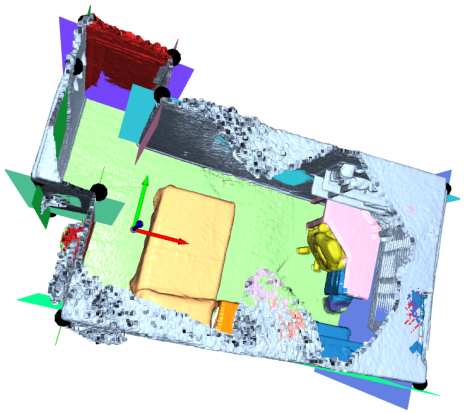


Fig. 4: Semantic segmentation & layout results obtained from 3D reconstructed scene in Figure 3a.

In parallel, from the reconstructed scene (Figure 3a) it was estimated the instances using Mask3D method. Particularly, it obtains an Avg AP50 $= 0.780$ on ScanNet dataset, and an Avg AP50 $= 0.388$ on ScanNet200 dataset. This quantitative result is related to the ones obtained qualitatively in Figure 5, where the different 3D objects are segmented into different classes and instances represented by colors. Specifically, the classes detected in this scene are: bed, nightstand, cabinet, shelf, chair and desk. These being some of the classes present in the ScanNet200 dataset.

Moreover, using previous results, it was applied ScanNotate method for CAD retrieval and 7-DoF pose estimation. This novel method outperforms the results obtained by Scan2CAD approach, as exposed in the original paper [19] using the scale, translation and rotation differences, and a visual evaluation

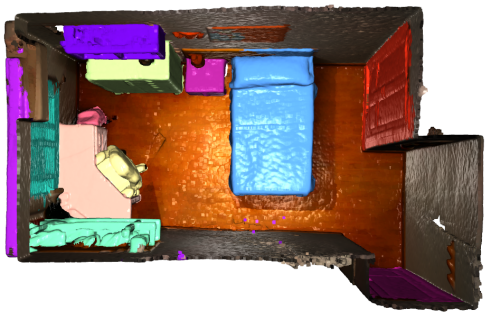


Fig. 5: Instance segmentation result obtained from 3D reconstructed scene in Figure 3a.

by experts. Figure 6 compares the results obtained by both methods. In particular, it is seen that ScanNotate obtained a better CAD retrieval result and also outperformed the pose estimation task. This could be related to the module incorporated by ScanNotate that joins instances of the same class that are similar and assign to that cluster the same CAD model ID.

Finally, using the results obtained from ScanNotate and layout estimation approach, it is obtained the virtualization depicted in Figure 3b from the 3D reconstructed scene in Figure 3a. As could be seen in the virtualization, the objects segmented by Mask3D are correctly replaced by the model CAD most similar in the same position, orientation, and scale, obtaining an accurate digitized scene.

After presenting the different steps performed in the virtualization process and its corresponding results, regarding computational cost, it is possible to obtain a full virtualization using VET, after having performed the 3D reconstruction, in an average of three and a half minutes, both with our own dataset and ScanNet dataset. This time was performed using a custom-built PC with an NVIDIA GeForce RTX 3060 and an Intel Core i7 CPU, and may vary depending on the size of the scene, and the number and types of objects present in it.

V. Conclusions

In conclusion, VET is a novel framework integrated into a single application that automates the complete virtualization of 3D scenes using CV and DL methods. It is versatile, capable of handling diverse indoor scenes and capable of working

Fig. 6: Comparison between 2 CAD retrieval and alignment methods using a ScanNet validation scene. From left to right: ScanNotate and Scan2CAD results.

with 200 object classes. This approach implements most of the current state of the art methods for each pipeline step, generating an accurate virtualization of the 3D scene. This framework has been qualitatively analyzed using a variety of indoor scenes from the ScanNet dataset and our own.

Future work includes training Mask3D to segment walls and floors, using these predictions to extract the layout information. Additionally, this framework could be extended to outdoor scene, contributing to outdoor VR applications, which remains a challenge. Finally, this framework will be applied in a real use case for psychological treatment. Using a 3D virtualization of a safe virtual world created with VET, the user will experience new realities without feeling threatened, as well as approaching traumatic or phobic situations gradually.

Acknowledgment

The research leading to these results has received partial funding from the European Commission under grant agreement N. 101017727 for the project EXPERIENCE. The author P.M. is the beneficiary of a University Teacher Training scholarship granted by the Spanish Ministry of Universities.

References

[1] Zheng, J. M., Chan, K. W., and Gibson, I. (1998). Virtual reality. Ieee Potentials, 17(2), 20-23.

[2] Dai, A., Nießner, M., Zollhöfer, M., Izadi, S., and Theobalt, C. (2017). Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. ACM Transactions on Graphics (ToG), 36(4), 1.

[3] Ipsita, A., Li, H., Duan, R., Cao, Y., Chidambaram, S., Liu, M., and Ramani, K. (2021, May). VRFromX: from scanned reality to interactive virtual experience with human-in-the-loop. In Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1-7).

[4] Seversky, L. M., and Yin, L. (2006, October). Real-time automatic 3D scene generation from natural language voice and text descriptions. In Proceedings of the 14th ACM international conference on Multimedia (pp. 61-64).

[5] Vouzounaras, G., Daras, P., and Strintzis, M. G. (2014). Automatic generation of 3D outdoor and indoor building scenes from a single image. Multimedia tools and applications, 70, 361-378.

[6] Marullo, G., Zhang, C., and Lamberti, F. (2022). Automatic Generation of Affective 3D Virtual Environments from 2D Images.

[7] Yang, J., Holz, C., Ofek, E., and Wilson, A. D. (2019, October). Dreamwalker: Substituting real-world walking experiences with a virtual reality. In Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (pp. 1093-1107).

[8] Li, Y., Li, W., Tang, S., Darwish, W., Hu, Y., and Chen, W. (2020). Automatic indoor as-built building information models generation by using low-cost RGB-D sensors. Sensors, 20(1), 293.

[9] Ipsita, A., Li, H., Duan, R., Cao, Y., Chidambaram, S., Liu, M., and Ramani, K. (2021, May). VRFromX: from scanned reality to interactive virtual experience with human-in-the-loop. In Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1-7).

[10] Moro, S., and Komuro, T. (2021). Generation of Virtual Reality Environment Based on 3D Scanned Indoor Physical Space. In Advances in Visual Computing: 16th International Symposium, ISVC 2021, Virtual Event, October 4-6, 2021, Proceedings, Part I (pp. 492-503). Springer International Publishing.

[11] Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., and Nießner, M. (2017). Scannet: Richly-annotated 3d reconstructions of indoor scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5828-5839).

[12] Dong, W., Lao, Y., Kaess, M., and Koltun, V. (2022). ASH: A modern framework for parallel spatial hashing in 3D perception. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[13] Garland, M., and Heckbert, P. S. (1998, October). Simplifying surfaces with color and texture using quadric error metrics. In Proceedings Visualization'98 (Cat. No. 98CB36276) (pp. 263-269). IEEE.

[14] Kadambi, A., Bhandari, A., and Raskar, R. (2014). 3D Depth Cameras in Vision: Benefits and Limitations of the Hardware: With an Emphasis on the First-and Second-Generation Kinect Models. Computer vision and machine learning with RGB-D sensors, 3-26.

[15] Araújo, A. M., and Oliveira, M. M. (2020). A robust statistics approach for plane detection in unorganized point clouds. Pattern Recognition, 100, 107115.

[16] Wang, P. S., Liu, Y., Guo, Y. X., Sun, C. Y., and Tong, X. (2017). O-cnn: Octree-based convolutional neural networks for 3d shape analysis. ACM Transactions On Graphics (TOG), 36(4), 1-11.

[17] Schult, J., Engelmann, F., Hermans, A., Litany, O., Tang, S., and Leibe, B. (2022). Mask3D for 3D Semantic Instance Segmentation., in press.

[18] Rozenberszki, D., Litany, O., and Dai, A. (2022, November). Language-grounded indoor 3D semantic segmentation in the wild. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIII (pp. 125-141). Cham: Springer Nature Switzerland.

[19] Ainetter, S., Stekovic, S., Fraundorfer, F., and Lepetit, V. (2023). Automatically Annotating Indoor Images with CAD Models via RGB-D Scans. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 3156-3164), in press.

[20] Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., ... and Yu, F. (2015). Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012.

[21] Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., and Xiao, J. (2015). 3d shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1912-1920).