CUS3D: A New Comprehensive Urban-Scale Semantic Segmentation 3D Benchmark Dataset Supplementary Material

Anonymous authors
Paper under double-blind review

Appendix

A Details of the data collection

Our dataset has been obtained by reconstructing 2D aerial image sequences using SFM technology, which recovers the camera extrinsics for every image. We have collected the 2D aerial image sequences through UAV oblique photography, using the DJI M300 RTK quadcopter and the five-lens oblique camera SHARE PSDK 102s. The resolution of each image is 6144×4096. Figure 1 shows the image acquisition equipment, and Table 1 describes the relevant parameters of the SHARE PSDK 102s camera.

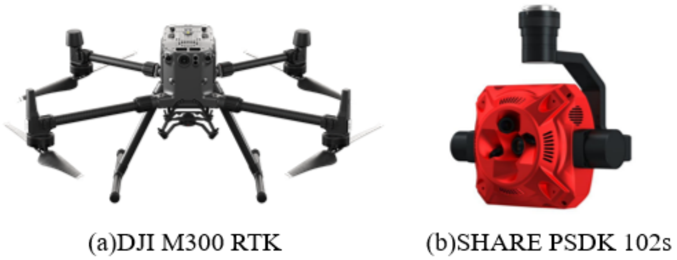

Figure 1: Left: DJI M300 UAV used for data collection; Right: SHARE PSDK 102s five-camera used for data collection.

Table 1: The parameters of the SHARE PSDK 102s camera

Performance parameters	Numerical value
Lens number	5
Tilt angle	45°
Image resolution	6144×4096
Focal lens	Downward:25 mm, sideward:35 mm
Sensor size	APS-C Format(23.5×15.6 mm)

During the UAV shooting process, we use a preplanned Z-shaped route. The image capture interval is 2 seconds. The flight altitude for capturing the original aerial image sequence with UAV is 100 meters; the weather conditions on the day of data collection are clear, without any obstructive factors such as clouds, haze, or strong winds, providing the drone with a good field of vision, visibility, and stable flight conditions. The UAV is equipped with a five-lens camera, which can capture five images in different directions every time. One lens captures images from a downward angle, while the other four lenses capture images from the sides. To ensure the accuracy of the subsequent 3D reconstruction model, we set up six manual control points at six take-off points for verification during the shooting process. Figure 2 shows the locations of 6 artificial control points.

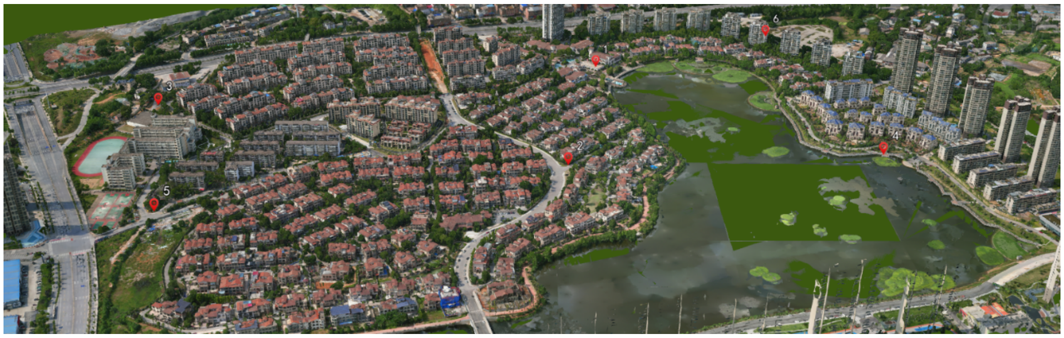

Figure 2: When capturing raw aerial images with a UAV, the positions of ground control points are manually set by the personnel on the ground.

We conduct aerial triangulation, 3D reconstruction, and other technical steps on the 2D aerial image sequence to generate a 3D textured mesh model. To obtain the correspondence between 2D pixels and 3D points, we constructed an affine transformation. First, we computed the relative pose transformation between cameras by matching feature points in the images, and constructed the extrinsic matrix to establish the relationship between the 3D vertex coordinate system and the camera coordinate system. Then, using camera parameters such as focal length, we constructed the intrinsic matrix to establish the relationship between the camera coordinate system and the pixel coordinate system. Through coordinate transformation, we established the relationship between 2D pixels and 3D points. To identify the outliers, during the feature matching stage, the matching error of feature points between two images is eliminated using the RANSAC algorithm to remove outliers, avoiding impact on the pose estimation process. During the dense matching stage, the point cloud data is denoised and smoothed to remove outliers and sparse point clouds. Figure 3 shows the dimensions and overall appearance of the reconstructed 3D textured mesh model.

The entire area covers an approximate land area of 2.85 square kilometers and consists of approximately 289 million triangular meshes. The terrain in this area is relatively flat, typical of suburban urban scenes, with a diverse range of features including buildings, roads, water systems, farmland, and vegetation. Note that the entire reconstructed scene is divided into 93 tiles, but only 89 tiles contain scene data. There are four tiles without any scene data, and their distribution is shown in Figure 4. These four blank tiles will be ignored in subsequent labeling and experiments. To provide a greater variety of 3D structures, we have also released point cloud data with true scene color information. The textured mesh data have been down-sampled to a point-cloud density of 0.15 meters, resulting in approximately 152 million points.

B Details of the data annotation

We use the DP-Modeler software for semi-automated 3D annotation to ensure that every triangle mesh is assigned the corresponding semantic labels. No triangle meshes are left unmarked. To prevent annotation errors and omissions, we ensure that all of the labels undergo two rounds of manual cross-checking to ensure accuracy. Figure 5 shows the issues discovered during the manual cross-checking process and promptly corrected. The semantic annotation process starts by inputting the source data, configuring the annotation categories according to requirements, and then performing manual labeling. After labeling has been completed, the classified data are subjected to quality inspection. Abnormal data are reclassified, resulting in labeled output data. The entire annotation process is illustrated in Figure 6, and the tool interface used in the semantic annotation process is shown in Figure 7.

In the selection of semantic labeling strategiesthe CUS3D dataset abandons the previous fixed urban semantic labeling strategy and adopts a dynamic urban semantic labeling strategy that considers both fully developed and developing semantic categories. For example, the categories of ”road” and ”ground” may have some overlapping characteristics. However, ”road” belongs to fully developed functional objects, which can be used for urban transportation planning and peak traffic control optimization research. ”Ground” belongs to undeveloped objects and labeling and recognition can

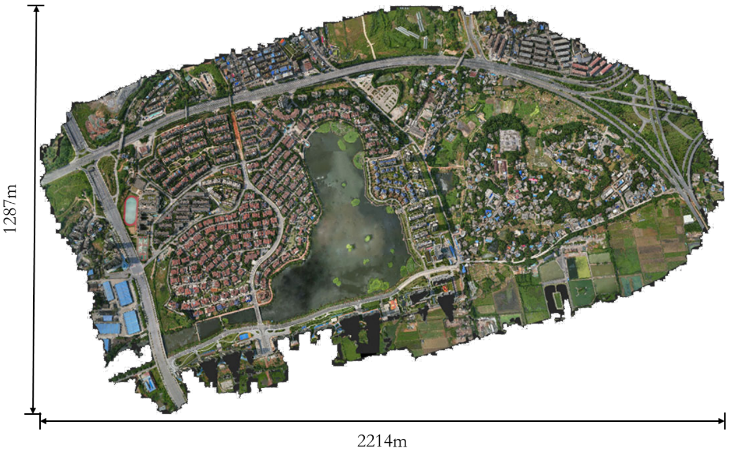

Figure 3: Original unmarked 3D texture mesh model. High-resolution texture mesh model reconstructed in 3D from a sequence of aerial images taken by a UAV, covering an area of 2.85 square kilometers.

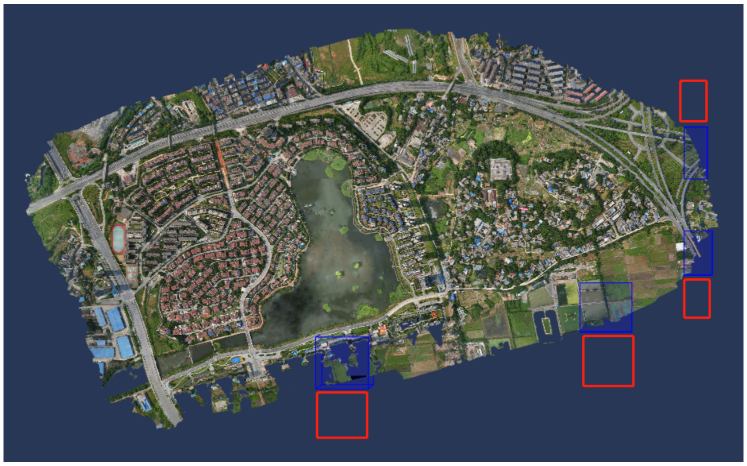

Figure 4: Blank Tile Distribution Chart (as indicated by the red box in the image).

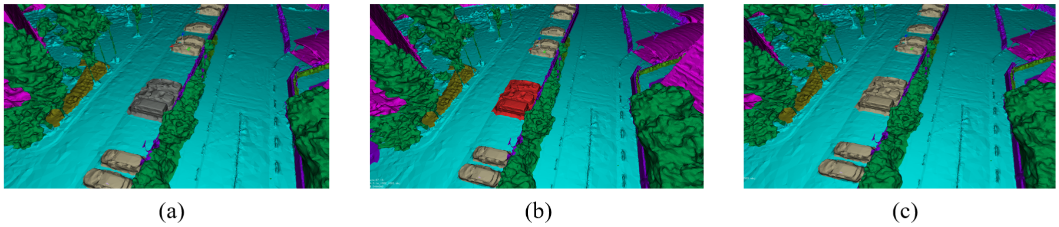

Figure 5: Artificial cross-checking detection of unmarked vehicles. (a) Unmarked vehicles detected; (b) Select unmarked vehicles; (c) Change the marking to vehicles.

help with early planning judgments in urban development. ”Building sites” belong to the category of ongoing semantic objects and can be transformed into building in future semantic updates. The semantic labeling strategy of the CUS3D dataset considers the application, functionality, and temporal aspects of objects, making it more suitable for practical applications such as urban planning, transportation planning, and construction decision-making.

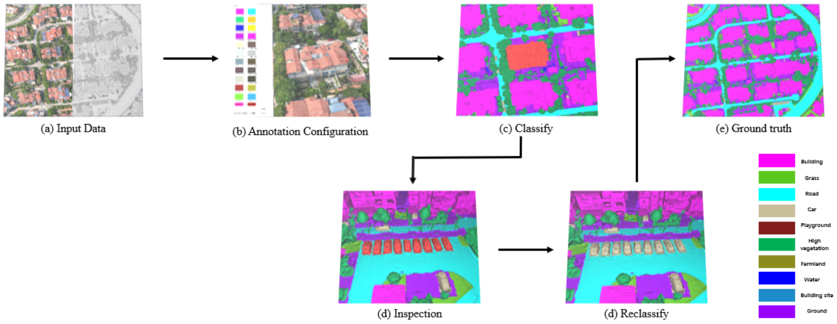

Figure 6: The pipeline of semantic tagging work.

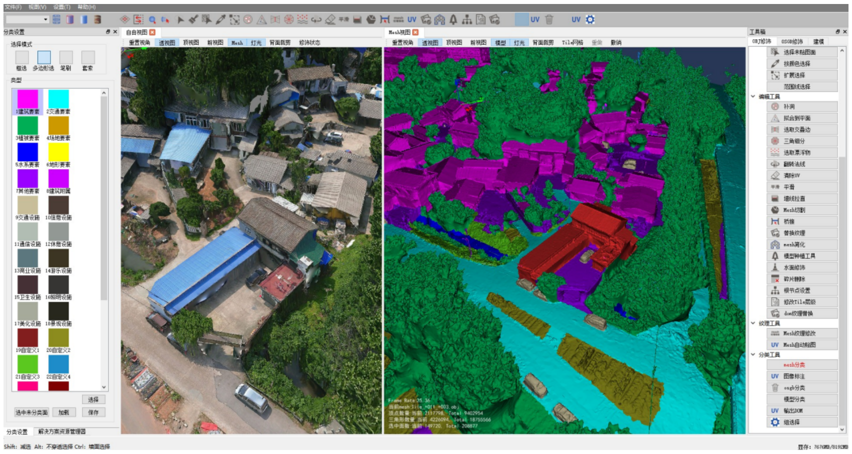

Figure 7: Semantic labeling software work interface.

We classify different objects in the scene into 10 semantic categories. Considering that the scarcity of certain objects does not affect the planning and research of large-scale scenes, we categorize some high-granularity object information (e.g., pedestrians, utility poles, and solar panels) into their respective larger categories. These 10 semantic categories comprehensively represent the scene information in cities and suburbs. Every semantic label is assigned a specific color information. Table 2 provides the RGB values and grayscale values corresponding to every semantic label. Figure 8 shows the semantic labeling results of certain regions.

Regarding 2D image semantic labeling, the entire 2D image sequence consists of 10,840 images from 5 different perspectives. Due to the high similarity in the poses of four cameras tilted at 45°, we only selected 4,336 image sequences with a 90° top-down view and one 45° oblique view for semantic annotation. We have adopted the ABAVA data engineering platform developed by an outsourced company in our project team. The data labeling module of this platform provides an automated instance segmentation tool based on the SAM algorithm. During the labeling process, the samples are first pre-labeled using the automated labeling module, and then manual adjustments are made to achieve high-precision pixel-level 2D image semantic labeling.

C Details of the Experiment

To verify the applicability of the CUS3D dataset on existing semantic segmentation networks, we conduct benchmark tests on seven 3D baseline methods. The hardware configurations are standardized for the seven 3D benchmark tests, and the detailed hardware information is shown in Table 3. To ensure consistency in the benchmark tests, we ensure that the training, testing, and validation sets use the same regions and data quantities for the seven 3D test networks: PointNet(Qi et al., 2017a), PointNet++(Qi et al., 2017b), RandLA-Net(Hu et al., 2020), KPConv(Thomas et al., 2019),

Table 2: Semantic label color information table

Category	**RGB value**	**Grayscale value**
Building	(254,1,252)	105
Road	(0,255,255)	179
Car	(200,191,154)	189
Grass	(91,200,31)	99
High vegetation	(0,175,85)	112
Playground	(130,30,30)	0
Water	(0,0,255)	29
Farmland	(140,139,30)	59
Building sites	(30,140,201)	201
Ground	(154,0,255)	75

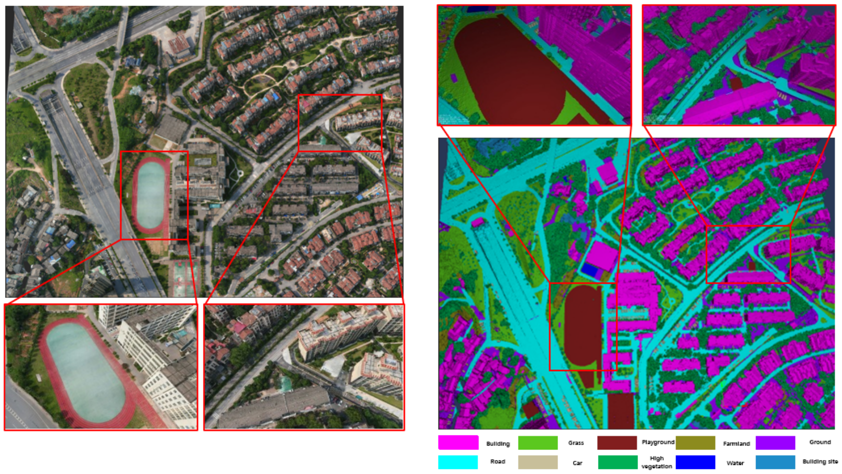

Figure 8: Partial region semantic labeling results. (a) Partial region original mesh; (b) Partial region semantic labeling results.

SPGraph(Landrieu & Simonovsky, 2018), SQN(Hu et al., 2022), and Stratified Transformer (Lai et al., 2022). Therefore, the dataset is uniformly partitioned. Additionally, owing to the influence of down-sampling on RandLA-Net(Hu et al., 2020) and SQN(Hu et al., 2022), some tile files have too few points. Therefore, the following seven LAS files are excluded when partitioning the dataset: Tile_+001+000, Tile_+001+006, Tile_+004+007, Tile_+005+000, Tile_+010+001, Tile_+011+007, and Tile_+012+002. In the main text, we show the distribution of the training, testing, and validation sets in the entire scene, as well as the distribution of every category in the mesh data. Table 4 shows the parameter settings of the baseline experiment.

Table 3: Baseline test experimental hardware environment configuration table.

Name	**Model**
System	4029GP-TRT2
CPU	Intel Xeon 4210R
Memory	SAMSUNG 32GB DDR4 ECC 293
System Disk	Intel S4510
Data Disk	Intel S4510
GPU	Nvidia Tesla V100

Table 4: Baseline experiment parameter settings

	Epoch	Batch_size	Num_point	Learning_rate	Optimizer	Momentum	Parameters	Time
PointNet	100	24	4096	0.001	Adam	0.9	0.97M	6.4h
PointNet++	200	32	4096	0.001	Adam	0.9	1.17M	6h
RandLA-Net	100	16	4096	0.01	Adam	0.9	4.99M	7.5h
KPConv	500	10	-	0.01	Adam	0.9	14.08M	8h
SPGraph	500	2	4096	0.01	Adam	0.9	0.21M	5.5h
SQN	100	48	4096	0.01	Adam	0.9	3.45M	7h
Stratified Transformer	500	8	4096	0.001	Adam	0.9	34.63M	41h

During the data partitioning for the experiment, a total of 82 tiles were used for the training/validation/testing sets. Among them, 4 tiles did not have mesh data because they were located at the edge of the measurement area, where the number of feature points for 3D reconstruction was too small to construct mesh data. Additionally, 7 tiles were not included in the dataset partition because, during the testing of the RandLA-Net network, the point cloud data of these 7 tiles had fewer than 4000 points, which did not meet the network's num_points requirement. In order to ensure the consistency of the input data for the network, these 7 tiles were deliberately ignored during the data partitioning.

To evaluate the performance of the CUS3D dataset, we choose ACC, Recall, F1 score, and IoU as evaluation metrics for every category. Overall, we choose mIoU, OA, and mAcc as evaluation metrics. In the main text, we present the metrics results for overall category testing and mIoU for every category. Tables 5, 6, 7, 8, 9, 10, and 11 show the test results for every semantic category in the seven networks.

Table 5: Evaluation metrics results of PointNet for every category

	Ground	Building	Road	High vegetation	Water	Car	Playground	Farmland	Grass	Building site
Acc(%)	95.96	96.47	90.44	98.03	92.14	98.77	99.90	99.90	86.72	99.18
Recall(%)	88.35	80.95	91.77	88.80	40.42	25.32	9.98	9.98	50.53	20.10
F1 Score(%)	87.85	79.34	87.84	84.56	40.17	33.58	16.17	16.17	56.21	28.40
IoU(%)	78.36	65.81	78.34	73.32	25.22	20.35	8.76	8.76	39.11	16.66

Table 6: Evaluation metrics results of PointNet++ for every category

	Ground	Building	Road	High vegetation	Water	Car	Playground	Farmland	Grass	Building site
Acc(%)	90.43	93.62	96.11	93.66	99.16	99.13	99.86	97.56	95.16	97.57
Recall(%)	42.37	89.62	83.14	91.72	65.01	46.83	18.64	72.42	53.42	16.09
F1 Score(%)	40.77	89.07	73.07	92.55	72.45	52.61	18.67	57.50	56.89	22.58
IoU(%)	25.78	80.36	57.77	86.18	57.52	36.13	11.38	40.88	40.17	13.28

Table 7: Evaluation metrics results of RandLA-Net for every category

	Ground	Building	Road	High vegetation	Water	Car	Playground	Farmland	Grass	Building site
Acc(%)	88.24	94.49	95.12	90.15	98.67	97.64	97.32	96.67	93.31	96.59
Recall(%)	42.56	80.52	83.02	80.44	81.33	55.31	57.31	49.56	70.68	65.00
F1 Score(%)	26.04	86.38	65.17	87.88	76.43	66.04	51.87	58.84	56.42	45.38
IoU(%)	15.29	76.18	49.51	78.42	62.30	49.89	35.91	42.11	39.74	9.86

Table 8: Evaluation metrics results of KPConv for every category

	Ground	**Building**	**Road**	**High vegetation**	**Water**	**Car**	**Playground**	**Farmland**	**Grass**	**Building site**
Acc(%)	93.79	95.00	97.50	93.94	99.51	99.26	99.90	98.21	96.18	98.76
Recall(%)	35.43	92.38	80.80	95.71	60.41	58.87	13.84	62.39	54.83	17.83
F1 Score(%)	40.19	90.54	65.16	94.52	63.81	60.27	12.58	57.87	61.53	25.14
IoU(%)	25.73	82.89	48.69	89.64	48.43	44.69	25.72	41.86	44.84	15.50

Table 9: Evaluation metrics results of SPGraph for every category

	Ground	**Building**	**Road**	**High vegetation**	**Water**	**Car**	**Playground**	**Farmland**	**Grass**	**Building site**
Acc(%)	92.73	69.59	96.03	21.90	99.03	98.98	99.89	97.16	92.39	98.09
Recall(%)	14.24	38.48	42.68	90.88	40.51	4.75	8.91	51.30	42.14	42.56
F1 Score(%)	21.30	48.48	46.59	77.49	47.91	8.78	16.04	53.30	47.54	46.18
IoU(%)	12.05	32.10	30.56	63.40	31.71	4.40	7.65	36.66	31.36	28.84

Table 10: Evaluation metrics results of SQN for every category

	Ground	**Building**	**Road**	**High vegetation**	**Water**	**Car**	**Playground**	**Farmland**	**Grass**	**Building site**
Acc(%)	86.09	95.36	95.06	91.98	98.28	98.04	97.87	96.52	93.22	96.45
Recall(%)	40.26	90.12	72.07	95.92	71.00	62.74	73.31	47.47	64.61	65.87
F1 Score(%)	46.44	87.76	74.74	87.73	73.81	70.70	50.94	56.37	61.96	46.32
IoU(%)	30.65	78.63	60.22	79.07	59.21	55.46	34.77	39.88	45.54	30.73

Table 11: Evaluation metrics results of Stratified Transformer for every category

	Ground	**Building**	**Road**	**High vegetation**	**Water**	**Car**	**Playground**	**Farmland**	**Grass**	**Building site**
Acc(%)	91.22	94.43	96.12	94.67	99.34	99.21	99.85	97.89	96.12	97.88
Recall(%)	33.44	90.1	83.3	91.56	59.32	57.01	14.55	62.45	55.78	18.32
F1 Score(%)	41.34	90.22	66.19	90.44	62.63	52.78	12.89	57.85	57.03	22.68
IoU(%)	60.23	70.02	75.44	83.36	58.47	58.66	32.45	39.66	34.59	39.33

REFERENCES

Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. RandLA-Net: Efficient semantic segmentation of large-scale point clouds. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020.

Qingyong Hu, Bo Yang, Guangchi Fang, Yulan Guo, Aleš Leonardis, Niki Trigoni, and Andrew Markham. SQN: Weakly-supervised semantic segmentation of large-scale 3D point clouds. In *Computer Vision – ECCV 2022*. Springer, 2022.

Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8490–8499, 2022. doi: 10.1109/CVPR52688.2022.00831.

Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2018.

Charles Ruizhongtai Qi, Hao Su, Mo Kaichun, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017a.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*. Curran Associates Inc., 2017b.

Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas Guibas. KPConv: Flexible and deformable convolution for point clouds. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2019.