VASAD: a Volume and Semantic dataset for Building Reconstruction from Point Clouds

Pierre-Alain Langlois, Yang Xiao, Alexandre Boulch, Renaud Marlet

VASAD: a Volume and Semantic dataset for Building Reconstruction from Point Clouds

Pierre-Alain Langlois[1,3], Yang Xiao[1], Alexandre Boulch[2] and Renaud Marlet[1,2]
[1]LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Vallée, France [2]Valeo.ai, Paris, France

***Abstract*— 3D scene reconstruction has important applications to help to produce digital twins of existing buildings. While the community has mostly focused on surface reconstruction or semantic segmentation as separate problems, the joint reconstruction of both volumes and semantics has little been discussed, mostly due to the lack of large scale volume datasets with semantic annotations. In this work, we introduce a new dataset called VASAD for Volume And Semantic Architectural Dataset. It is composed of 6 building models, with full volume description and semantic labels. It approximately represents 62,000 m^2 of building floors, making it large enough for the development and evaluation of learning-based approaches. We propose several methods to jointly reconstruct both geometry and semantics and evaluate on the test set of the dataset. We show that the proposed dataset is challenging enough to stimulate research. The dataset is available at https://github.com/palanglois/vasad.**

I. Introduction

In recent years, the construction industry has developed a new type of digital model called *Building Information Model* (BIM), for a better conception, construction and maintenance of buildings. While 3D models have already been used for building design, BIMs include a richer information: the semantic class of each building component (e.g., windows, walls), as well as other technical information. BIM creation and update of existing buildings require reconstructing both the volumetric geometry and the semantics of the current building state [1].

To reconstruct 3D models, buildings are first scanned, often using a lidar. This sensor provides accurate depth measurements, that can then be used to produce high-quality basic geometric features, such as normals [2]. Although not as accurate, depth cameras are also used for building scanning. Nevertheless, both kinds of sensors suffer from inherent weaknesses: noise depending on depth and surface inclination, missing or wrong measurements due to reflecting or transparent surfaces. Moreover, measurements are often taken from fixed viewpoints, which generates a non-uniform surface sampling. Last, and as is the case for any sensor, building parts may be missing because of occlusions or inattention from the operator. 3D reconstruction from such incomplete, inhomogeneous and possibly noisy data necessitates strong priors, that are better learned automatically from existing data.

Many learning-based approaches have been developed for the 3D semantic segmentation of point clouds [3], [4], [5], and a few for 3D scene reconstruction [6], [7], [8]. However few methods tackle the semantic volume reconstruction [9] and for full building reconstruction, related work is even scarcer [10].

[3]The publication was written prior to the employee joining Amazon.

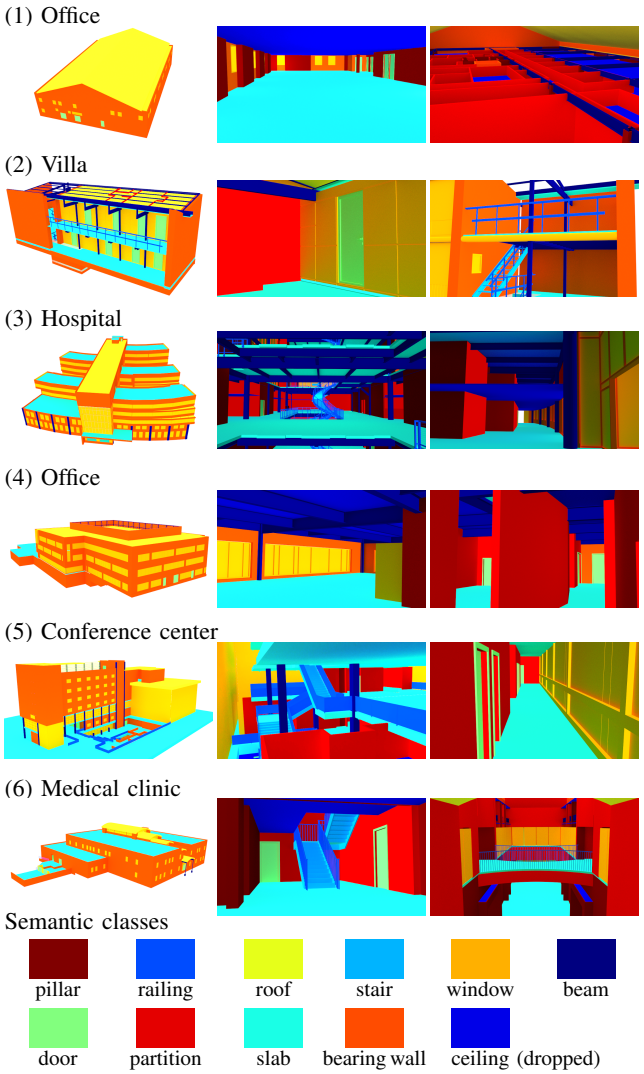

Fig. 1. Outdoor views, interior details & colors of semantic classes in VASAD.

The main reason, we argue, is the lack of suitable datasets. Indeed, most existing datasets address one or the other task, and those that hold both geometric and semantic information feature objects (e.g., furniture), not building components. Besides, these objects are often given as open surfaces instead of closed volumes, which is however required for BIM modelling.

Our work intends to help to fill this gap, both in terms of dataset and methodology, towards the reconstruction of full digital mockups. Our contributions are twofold.

First, we introduce a new dataset called VASAD for *Volume And Semantic Architectural Dataset*, aimed at building

reconstruction from point clouds. It is composed of 6 complete building models, with volume description and semantic labels for each building component. It represents over 62,000 m^2 of building floors, making it large enough for the development and evaluation of learning-based methods. In fact, we believe this dataset can be particularly valuable to train and evaluate methods that generate BIM models from raw Lidar scans.

Second, we present a deep neural network for joint semantic and geometric reconstruction. It is built over of a semantic feature extractor followed by a dense voxel-reconstruction network. At any location in space, we predict both the occupancy (in or outside matter) and the semantic label of the component. Quantitative evaluations on VASAD show that it outperforms a baseline method that we created from a state-of-the-art reconstruction method [7] to also include semantics.

II. Related Work

A. Methods

1) Surface Reconstruction from Point Clouds: Assuming that points are sampled densely enough and without much noise, proven methods allow meaningful mesh reconstruction [11]. When points are noisy, a smoothness prior is required, also to handle outliers [12], [13]. As man-made environments have specific geometric features, e.g., planarity and orthogonality, similarly-specific reconstruction methods have been proposed with piecewise-planar [14], [15] or Manhattan-world assumptions [16], producing idealized models, possibly with remaining free-form parts [17]. However, to avoid such handcrafted priors, recent methods learn priors by leveraging large datasets of 3D objects [18] and scenes [19], [20]. Common data representations for learning-based 3D reconstruction are voxels [21], meshes [22], [23], and implicit functions [24], [25], including for large scenes [6], [7], [8], [26]. Piecewise planarity may also be integrated as an extra prior [27].

2) Semantic Segmentation of Point Clouds: Early methods [28] have formulated the point cloud segmentation problem as region growing [29], model fitting [30], clustering [31] or graph-cut optimization [32]. These formulations allow handcrafted priors, e.g., underlying basic shapes such as planes or cylinders [30], or local properties as descriptors [32] . While these methods can be effective in particular cases, they fail at handling semantic classes that involve complex priors.

Recent methods directly learn complex priors from large datasets. A pioneering work is PointNet [4], possibly in a multi-scale architecture [33], which use a multi-layer perceptron (MLP) with a pooling operation to get an invariance to point ordering. To obtain translation invariance and to better scale to large scenes, convolution was extended to point clouds [34], [5], [35], [3], or applied (sparsely for efficiency) to a regular 3D grid after point voxelization [36], [37], [38].

3) Joint Reconstruction and Semantic Segmentation: Intuitively, geometric reconstruction and semantic segmentation should help each other. Pioneering work in this direction [9] simultaneously completes and labels voxels obtained from a single depth image. Recent work try to directly leverage the synergy between both tasks by affinity learning [39], by supervising the reconstruction task thanks to a pre-trained segmentation network [40], by introducing a content-consistency constraint [41], or by independently estimating the geometry on the segmented classes and adding a merging step [42], [43].

Implicit representations are a major tool to achieve both tasks, because they allow the problem to be formulated as an optimization task over the whole 3D space, whose data term can be handcrafted [44] or learned [45]. Yet, very few projects aim at analyzing and reconstructing whole buildings [46]. Most works stay essentially at the scale of a room and its furniture, focusing more on surface than volume. The reason is mainly the lack of data for learning to model the full structure of a volumetric building geometry with semantics.

B. Datasets

Existing datasets mainly focus on a single aspect, either geometry or semantics, and with little emphasis on volumes.

1) Semantic information: Besides object detection with bounding boxes, recovering semantics in 3D often takes the form of point cloud segmentation, either in automotive environments (e.g., SemanticKITTI [47], [48] NuScenes [49]), outdoors (e.g., Semantic3D [50], NPM3D [51]), or indoor scenes (e.g., S3DIS [52], MatterPort3D [53], ScanNet [20]).

2) Surface information: Except a few unsupervised methods that can learn surface information from raw points clouds [54], most learning-based reconstruction methods rely directly or indirectly on supervision from mesh datasets, synthetic (CAD-based) like ShapeNet [18] or SceneNet [19], or real (based on actual scans) like DFaust [55] or MatterPort3D [53].

3) Volume information: Though many reconstruction needs do not require watertight meshes, a number of approaches, including methods based on implicit functions, necessitate the supervision of closed meshes (i.e., volumes) as they sample points and label them as inside or outside the shape. However, due to their creation process, whether it is manual (for CAD-based datasets) or algorithmic, many mesh datasets have poor topological properties (e.g., self-intersection) and feature open meshes. Consequently, a substantial preprocessing step is required to close the meshes and create volumes [56], [57].

4) Semantic and volume information: Datasets associating both volume and semantics are very rare, generally limited to single rooms or flats with furniture [58], [19], [59], [60], or to building exteriors [61]. S3DIS [52] contains semantics and volumetric information at building level, but the (coarse) volume data is also mostly focused on furniture; it does not properly represents the building structure. In particular, volumes are independently built for each room, and floors and walls are given an arbitrary thicknesses, cf. Fig. 2. Reliably creating consistent building volumes from semantized surfaces would be extremely difficult due to numerous special cases.

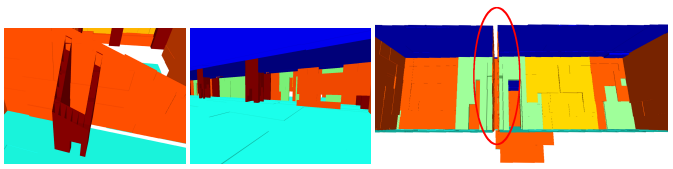

Fig. 2. Empty columns, holes and hollow structural parts in S3DIS [52].

III. VASAD DATASET

To address the lack of data covering both volume and semantics at building level, we introduce a new dataset, called VASAD for *Volume And Semantic Architectural Dataset*. This synthetic dataset aims at leveraging machine-learning techniques for architectural reconstruction, towards BIM creation.

A. Overview

As opposed to other datasets, our primary goal is to create a realistic database for building reconstruction, focused on the structure and components of the building instead of furniture.

VASAD is derived from six BIM models made by architects that can be freely used for research purposes. The buildings vary from a villa to a large-scale hospital, totalling more than 62,000 m^2. They are listed in Tab. I and illustrated in Fig. 1, together with the semantic classes of their components (door, pillar, etc.). A few common inconsistencies in BIM modelling remain, e.g., a slab being used both as floor and flat roof.

These classes were obtained by filtering the component names in the BIM models. We strove to create classes that are relevant to properly model arbitrary building structures, avoiding overlap between classes, though some of them can be challenging to tell apart, e.g., partition vs bearing wall, slab vs flat roof, partition vs pillar. As opposed to S3DIS [52], the dataset contains full and accurate structure information. In particular, the volumes represent the complete structure components and are not limited to a few-millimeter thick envelope. Furniture present in a few original BIMs was discarded.

B. 3D Representation of Closed Shapes

We provide access to volume and semantic information via a function that returns the class of an arbitrary query point in 3D (including full or empty), like an implicit representation does. Explicit semantized 3D representations can then be generated from this function, including point clouds, voxels and meshes.

In order to build this function, the input shapes (i.e., the various building components in the BIM model) have to be closed, as discussed above. It is almost the case for our cured VASAD models, though some construction errors still remain. To nevertheless create a consistent oracle, we built on the idea that any ray starting from a point inside a shape intersects its surface an odd number of times (before going to infinity). To be robust with respect to the construction errors such as meshes not well closed, we cast 3 axis-aligned rays to determine the label of a point. A majority vote on the label of the first intersected surface decides on the correct label.

C. Point Cloud Scanning Simulation

Rather than consider input points uniformly sampled on the surface, as if often the case with synthetic data [7], we more realistically do virtual scans, shooting rays from viewpoints.

1) Generation of Viewpoints: To be realistic, besides being in empty space, viewpoints should be spread across the model so that most surface pieces are visible from at least one viewpoint. Enforcing this kind of constraint is difficult because it could require shooting many rays, which is computationally expensive. Our heuristics is to iteratively add new viewpoints as long as they are not visible from viewpoints already placed. While it does not guarantee that every bit of surface is visible from a viewpoint, it empirically produces a fair surface covering and is much more tractable computationally.

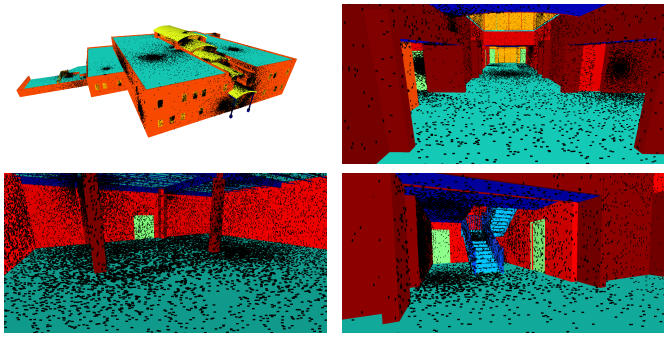

Fig. 3. Fully-automated virtual scanning of the ground-truth mesh (test set).

TABLE I
BUILDING COMPOSITION IN THE VASAD DATASET

Id	Building name	Building type	split	area (m^2)	#compon.
1	NBU-OfficeBuilding	office	train	3,700	1,241
2	Sextant	residential (villa)	train	228	1,444
3	WestRiverSideHospital	hospital	train	29,600	23,661
4	Trapelo	office	train	18,400	5,500
5	OTC-ConferenceCenter	conference center	train	5,800	3,657
6	NBU-MedicalClinic	medical clinic	test	4,500	3,094

2) Ray Shooting: For virtual scanning, we do not try to comply to a particular lidar type. From each viewpoint, we shoot rays uniformly in all directions. The scanned point is the first intersection with the mesh. We save the semantic label of the intersected object instance (building component) of the mesh, and surface normal (which could also be estimated [2]).

While not fully realistic, this automated virtual scanning is still significantly more realistic than the usual uniform mesh sampling. It takes into account sampling density variations due to view incidence on the surface, and it prevents sampling on surfaces pieces that are invisible in practice (see Fig. 3).

(Creating more realistic scans taking into account the shooting geometry and resolution of different lidar models, as well as depth and incidence-based noise, is future work.)

D. Train/Test Split

S3DIS [52] features 6 areas from 3 different buildings. Its train/test split is made to avoid that similar parts are seen in both sets. We leverage the diversity of buildings in VASAD to propose a train/test split where the test set consists of a full building. As our buildings have different sizes, we choose as a test building *Medical Clinic*, whose size is average, cf. Tab. I.

IV. SEMANTIZED RECONSTRUCTION

We present our method, that extracts point-level semantic features and does semantic reconstruction with a voxel network. As volume-and-semantic reconstruction at building scale is a new task, there is no method to compare to. Therefore, we also define a baseline, that is a direct extension of a state-of-the-art reconstruction method to also handle semantics.

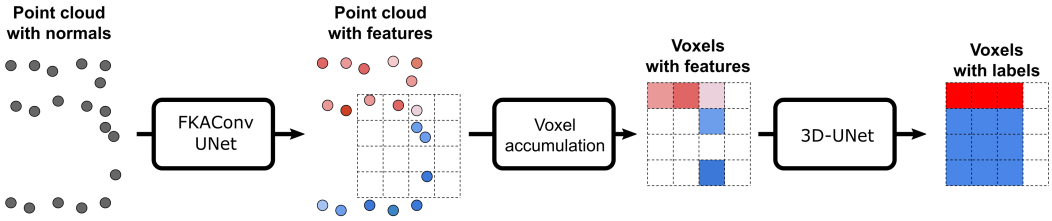

Fig. 4. PVSRNet extracts semantic features via U-Net point convolution, aggregates them into voxels, and infers semantic volume occupancy via a 3D U-Net.

A. Point-Voxel Semantic Reconstruction Network (PVSRNet)

The voxel representation is a natural way to operate in 3D, allowing a direct transposition of proven methods used for images. In particular, the U-Net architecture [62], first used for image segmentation, has been successfully generalized to 3D for dense volume segmentation from sparse annotations [63]. However, one of the main drawbacks of voxels is memory consumption, that may impose coarse resolutions and/or small analysis contexts. Methods have thus been proposed to exploit input sparsity [36], [37], [38]. It is however little inappropriate for volume reconstruction from surface points, where empty voxels at input may turn out to be inferred as full, unless using a recently-proposed sparse voxel completion method [64].

1) Our method: We chose to first leverage a point convolution network, which is able to process a large space region at once, and therefore to exploit a wide context. And only in a second stage, a voxel-based network performs semantized reconstruction. Having access to a large context is particularly important for the semantic analysis of buildings. For instance, while ceilings and slabs are locally similar, their relative position in a building can help to disambiguate them.

This method, called Point-Voxel Semantic Reconstruction Network (PVSRNet), is illustrated in Fig. 4. The input point cloud, possibly with normals, is first processed at large scale by a U-Net based on FKAConv [3], computing a feature vector per point. Point features are then averaged to form voxel features. At this stage, only voxels close to the surface, where points are sampled, receive information. Last, a voxel-based 3D U-Net propagate this information into the whole volume.

2) Scaling to Entire Buildings: The input is split into adjacent cubic chunks of size 2.40 m, each chunk corresponding to 48^3 voxels of size 5 cm. However, the point convolution for semantic feature extraction has access to a larger context: a 6 m-diameter ball centered on each chunk.

Regarding supervision, voxels are labeled with a strategy biased towards full space, as to balance the fact that 70% of the models is made of void. We split each voxel into 3^3 subvoxels and randomly sample (at most) one point in each subvoxel, that gets the label of the point, if any. If all 27 subvoxels are labeled with void, then the voxel gets a void label. Otherwise, the voxel gets the label of the most represented full class among the subvoxels. This allows us to recover thin objects such as windows, which otherwise often would fail to be represented by a majority vote of 27 subvoxels, dominated by void.

This model is trainable end-to-end. However, as it consumes too much memory for middle-range GPUs, we first train the FKAConv U-Net with point label supervision, using a cross-entropy loss; the point features are the raw logits, just before the softmax layer. Then we train the 3D U-Net with voxel label supervision (see above), using a cross-entropy loss too.

B. Semantic Convolutional Occupancy Network (SemConvONet)

Our baseline, called SemConvONet, is a direct extension of ConvONet [7] to semantics. ConvONet reconstructs a volume from a point cloud via an implicit occupancy function, from which a mesh is created via the Marching cubes [65]. Scaling to scenes requires using ConvONet in sliding-window mode.

1) Method: We use the 3D-grid variant of ConvONet, that performs best on scenes [7]. Feature vectors are first extracted on a regular 3D grid with a local PointNet [4], then refined using a 3D U-Net [63] into latent vectors. Last, the latent vector of a query point is trilinearly interpolated from nearby grid latent vectors, and fed into an MLP to yield an occupancy.

To obtain a semantic information on top of geometry, we modify the prediction occupancy function: given a point, possibly with normal, instead of just predicting a binary occupancy (full or empty), we predict a semantic class, including void. The network is trained with a standard cross-entropy loss.

At inference time, ConvONet directly applies the Marching cubes on the output logits, which allows a smooth interpolation between Marching cube voxels. In our case, we apply a hard argmax on the logits, and the Marching cubes are applied for each non-void semantic class against all other classes. Therefore, the global reconstructed mesh includes one mesh per class, although possibly split into disconnected components. Using a hard argmax does not allow a smooth interpolation (hence the voxelized aspect of the output), but it allows to perfectly stitch two volumes of different class, without an in-between void or overlap. This is essential for BIM modelling.

2) Scaling to Entire Buildings: The input is split into adjacent cubic chunks of size 4 m (randomly rotated along the z-axis at training time). To allow comparison to PVSRNet, each chunk is similarly split as a 5 cm-voxel grid and one point is sampled in each 3^3-subvoxels. As it would result in too many points for a sliding window to see enough context, we further downsample the voxel points, making sure to have at least one point per class that is represented in the voxel. This typically yields more than 500k points. We complete them to 1M by randomly sampling among the remaining points.

TABLE II
QUANTITATIVE RESULTS ON VASAD.

Method	Input		Output	Super-vision	Volume metrics	
	Nor-mals	Surface sem.			IoU Sem. ↑	IoU Geom. ↑
Pure geometric reconstruction						
ConvONet	✗	✗	points	geom.	-	0.31
Joint geometric and semantic reconstruction						
SemConvONet	✗	✗	points	sem.	0.25	0.29
	✓	✗	points	sem.	0.23	0.28
3D U-Net	✗	✗	voxels	sem.	0.37	0.49
	✓	✗	voxels	sem.	0.41	0.55
PVSRNet	✓	✓	voxels	sem.	**0.53**	**0.59**

V. EXPERIMENTS

A. Evaluation Metrics

Volumetric semantic reconstruction is twofold. We evaluate joint geometry and semantics, as well as pure geometry. (It make no sense to assess semantics alone.) Evaluation is carried out by uniformly sampling 3D points in the union of the ground-truth bounding box and of the predicted 3D model. For each point, we compare the ground-truth label (obtained with the procedure from Sect III-B) to the predicted label. We evaluate the volume quality of the reconstruction with the intersection over union (IoU), either averaged over all semantic classes (IoU Sem.), or computed over empty/full labels (IoU Geom.), all non-void predictions being then considered as full. 10M points per model ensure stable metrics.

B. Results

We evaluate our PVSRNet method and the SemConvOnet baseline on VASAD. We present quantitative results in Tab. II and qualitative results in Fig. 5.

We first consider pure geometric reconstruction, where a vanilla ConvONet is trained on VASAD by aggregating all the non-void semantic labels into a full label. The relatively low IoU, while ConvONet is among the state-of-the-art reconstruction methods, illustrates the difficulty of the task.

We then train two versions of SemConvONet: with raw point coordinates as input, and with the addition of normal information (extending the input size of the PointNet encoder). The requirement, at loss level, to also predict semantics seems to be a heavy burden for the (Sem)ConvONet architecture, that does not succeed in leveraging the semantic supervision to eventually also improve the geometry. The slight performance decrease when inputting normals, which is within the variance margin of model training, could be due to the difficulty, for this architecture, to interpret the area of influence of normals in a point cloud with very high variations of sampling density.

As an ablation study for PVSRNet, we also train a model, reported as "3D U-Net" in Tab. II, that corresponds to the second stage only of PVSRNet, i.e., without semantic features as input. Like with SemConvONet, we train two variants, with or without normals as input. In spite of voxel discretization that inherently restricts the accuracy of the reconstructed surface (see differences in floor reconstruction on Fig. 5), our method does better than any version of SemConvONet regarding both in IoU Sem. and in IoU Geom. Experiments also show that SemConvONet is very sensitive to input sampling (see Fig. 5, row 4). A reason could be that ConvONet was developed with uniform sampling in mind, whereas our dataset features high variations of sampling density, as is the case with real scans. We also observe that 3D U-Net largely benefits from normals, which shows that the architecture is much more appropriate than SemConvONet to make use of this kind of information.

Finally, we evaluate the full-fledged PVSRNet method, that leverages both normals and rich semantic features. We observe that it outperforms all the other methods by a large margin, including SemConvONet. Compared to 3D U-Net alone, PVSRNet gains +12 IoU Sem. pts (semantic classes) and +4 IoU Geom. pts (void/full status). Note that the building semantic classes are distributed in a highly non uniform way. Structural classes such as bearing walls, partitions and slabs are more represented than smaller components such as beams, doors, pillars, railings, stairs or windows. Looking at actual reconstructions, we observe that SemConvONet strives to recognize less-represented classes, the prediction being then biased toward void, walls or floors. On the contrary, PVSRNet better recovers classes such as beams, stairs and windows, with less holes in thin surfaces. For instance, on Fig. 5, one can notice the windows in rows 1, 4, 5, 6, the door and the beams in row 2, and the railings in row 3.

C. Discussions and Perspectives

1) Dataset Ambiguities: Architects that create BIM models commonly make different design choices, reflecting the same global geometry but using different semantic components, or different subdivisions. For instance, given that some flat roofs can be accessed, there is an intrinsic ambiguity between slabs and roofs. Such components should have both labels, not to penalize learning and evaluation. Also, there are many ways to pave the wall volume of a building into cuboids.

2) Lack of Information: Sometimes information is missing to recover some building elements. For example, a closed door can be hard to distinguish from the partition in which the opening is made, in particular if there is no door frame. A possible perspective is to enrich representations with texture and/or material, and to model more realistic lidar scans featuring returned intensity or even lower-level signal shapes. Textures would enable the complementary use of virtual pictures too.

3) Inherent ambiguities: Some components are hard to distinguish from others. For instance, partitions tend to be thinner than bearing walls, and bearing walls tend to be mainly on the building envelope. However, there are still exceptions, which can make these classes hard to distinguish.

4) Small Details: One of the main limitations of PVSRNet is the voxel size, which prevents us from properly representing classes with small details, such as railings (see Fig. 5, row 3). A low resolution may also introduce ambiguities, e.g., windows being hard to distinguish in walls, as doors in partitions.

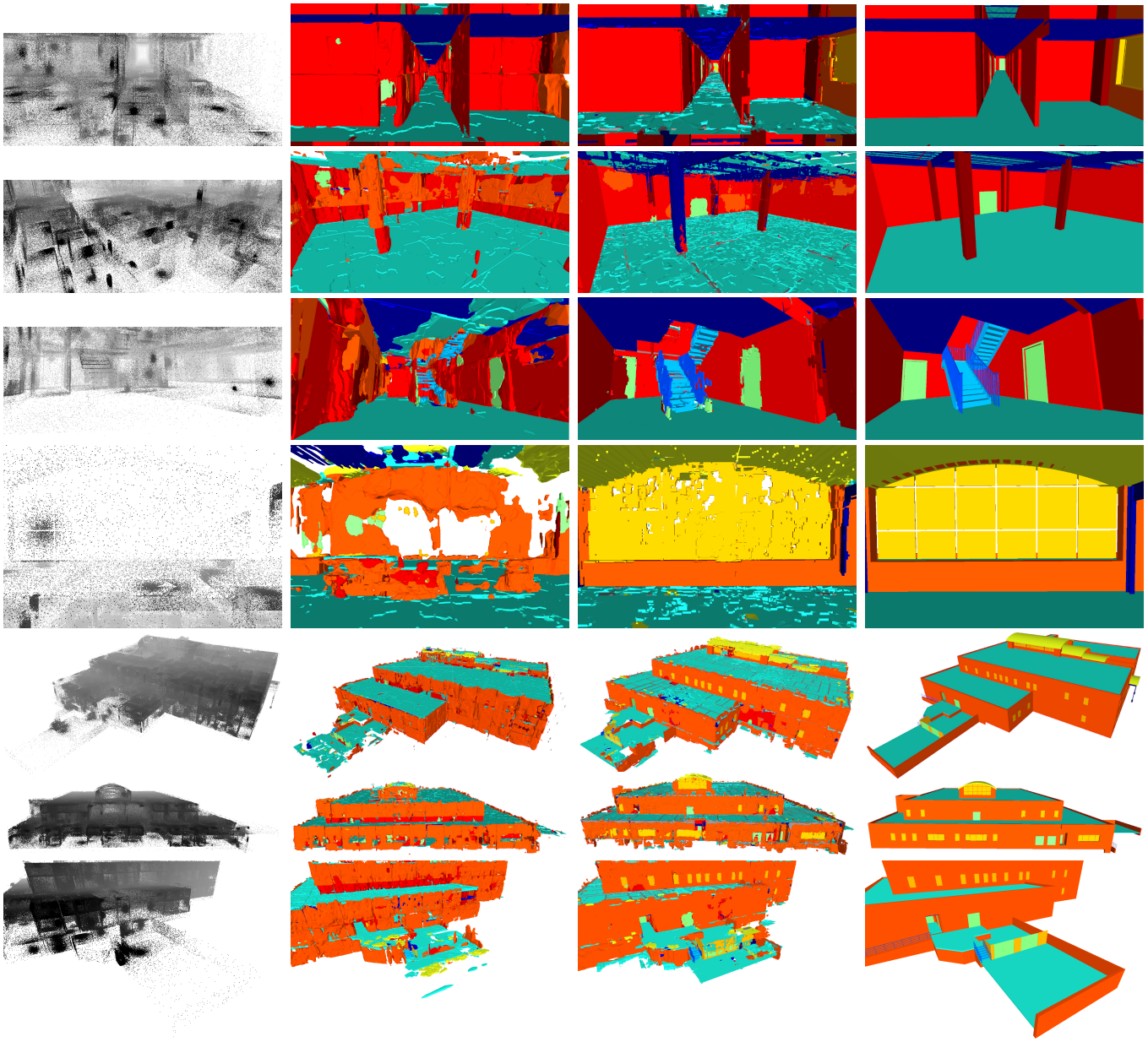

Fig. 5. Qualitative results of SemConvONet and PVSRNet on the test set of VASAD.

5) Voxels vs Implicit: In our experiments, PVSRNet outperfoms SemConvONet by a huge margin, including for less-represented classes, and we suspect it is partly due to the sensitivity of ConvONet to non-uniform sampling. It is future work to understand the pros and cons of voxel-based representations vs implicit functions, except that implicit functions can also be computed based on a 3D grid, as in ConvONet.

VI. Conclusion

We introduced VASAD, a novel and freely available dataset for the task of semantized building reconstruction. The dataset (https://github.com/palanglois/vasad) features both the volumetric geometry and the semantics of building components, which is key towards reconstructing BIM models of existing constructions. While our work focuses on volume occupancy and semantics, VASAD can also be used for instance segmentation, except for large components like walls, that can be partitioned in many different ways. To facilitate the creation of more data from other BIM models, we make our tools available too. They include our point labeling method, that is robust to mesh defects, and our procedure for lidar simulation, that features an automated viewpoint positioning. More details are provided in the supplementary material. Last, we proposed a "natural" baseline method to address the task, and a more sophisticated approach that largely outperforms the baseline. Yet there is still room for improvement: VASAD is challenging dataset for modern semantized-reconstruction methods.

We hope these contributions can pave the way towards fully automatic, so-called *scan-to-BIM* methods.

REFERENCES

[1] C. Thomson and J. Boehm, "Automatic geometry generation from point clouds for BIM," *Remote Sensing*, vol. 7, no. 9, pp. 11 753–11 775, 2015.

[2] A. Boulch and R. Marlet, "Fast and robust normal estimation for point clouds with sharp features," *Computer Graphics Forum (CGF)*, vol. 31, pp. 1765–1774, 2012.

[3] A. Boulch, G. Puy, and R. Marlet, "FKAConv: Feature-kernel alignment for point cloud convolution," in *Asian Conference on Computer Vision (ACCV)*, 2020.

[4] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *International Conference on Computer Vision (ICCV)*, 2019.

[6] A. Dai, C. Diller, and M. Nießner, "SG-NN: Sparse generative neural networks for self-supervised scene completion of RGB-D scans," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger, "Convolutional occupancy networks," in *European Conference on Computer Vision (ECCV)*, 2020.

[8] B. Ummenhofer and V. Koltun, "Adaptive surface reconstruction with multiscale convolutional kernels," in *International Conference on Computer Vision (ICCV)*, 2021.

[9] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[10] Z. Murez, T. van As, J. Bartolozzi, A. Sinha, V. Badrinarayanan, and A. Rabinovich, "Atlas: End-to-end 3D scene reconstruction from posed images," in *European Conference on Computer Vision (ECCV)*, 2020.

[11] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin, "The ball-pivoting algorithm for surface reconstruction," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, vol. 5, no. 4, pp. 349–359, 1999.

[12] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Eurographics Symposium on Geometry Processing (SGP)*, 2006.

[13] M. Kazhdan and H. Hoppe, "Screened Poisson surface reconstruction," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 3, 2013.

[14] A. Boulch, M. de La Gorce, and R. Marlet, "Piecewise-planar 3D reconstruction with edge and corner regularization," *Computer Graphics Forum (CGF 2014)*, vol. 33, no. 5, pp. 55–64, 2014.

[15] L. Nan and P. Wonka, "PolyFit: Polygonal surface reconstruction from point clouds," in *International Conference on Computer Vision (ICCV)*, 2017.

[16] C. A. Vanegas, D. G. Aliaga, and B. Benes, "Automatic extraction of manhattan-world building masses from 3d laser range scans," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, vol. 18, pp. 1627–1637, 2012.

[17] F. Lafarge and P. Alliez, "Surface reconstruction through point set structuring," *Computer Graphics Forum (CGF)*, vol. 32, 2013.

[18] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3D model repository," Stanford University — Princeton University — Toyota Technological Institute at Chicago, Tech. Rep. arXiv:1512.03012 [cs.GR], 2015.

[19] A. Handa, V. Patraucean, S. Stent, and R. Cipolla, "SceneNet: An annotated model generator for indoor scene understanding," in *International Conference on Robotics and Automation (ICRA)*, 2016.

[20] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[22] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry, "A papier-mâché approach to learning 3D surface generation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[23] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2Mesh: Generating 3D mesh models from single RGB images," in *European Conference on Computer Vision (ECCV)*, 2018.

[24] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[25] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3D reconstruction in function space," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[26] J. Tang, J. Lei, D. Xu, F. Ma, K. Jia, and L. Zhang, "SA-ConvONet: Sign-agnostic optimization of convolutional occupancy networks," in *International Conference on Computer Vision (ICCV)*, 2021.

[27] Z. Chen, A. Tagliasacchi, and H. Zhang, "Bsp-net: Generating compact meshes via binary space partitioning," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[28] A. Nguyen and B. Le, "3D point cloud segmentation: A survey," in *IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, 2013.

[29] P. Besl and R. Jain, "Segmentation through variable-order surface fitting," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 10, no. 2, pp. 167–192, 1988.

[30] R. Schnabel, R. Wahl, and R. Klein, "Efficient RANSAC for point-cloud shape detection," *Computer Graphics Forum (CGF)*, vol. 26, no. 2, pp. 214–226, Jun. 2007.

[31] J. M. Biosca and J. L. Lerma, "Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 63, no. 1, pp. 84–98, 2008, theme Issue: Terrestrial Laser Scanning.

[32] A. Golovinskiy and T. Funkhouser, "Min-cut based segmentation of point clouds," in *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2009.

[33] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

[34] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on X-transformed points," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[35] W. Wu, Z. Qi, and L. Fuxin, "PointConv: Deep convolutional networks on 3D point clouds," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[36] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree generating networks: Efficient convolutional architectures for high-resolution 3D outputs," in *International Conference on Computer Vision (ICCV)*, 2017.

[37] B. Graham, M. Engelcke, and L. van der Maaten, "3D semantic segmentation with submanifold sparse convolutional networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[38] C. Choy, J. Gwak, and S. Savarese, "4D spatio-temporal ConvNets: Minkowski convolutional neural networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3075–3084.

[39] Z. Zhang, Z. Cui, C. Xu, Y. Yan, N. Sebe, and J. Yang, "Pattern-affinitive propagation across depth, surface normal and semantic segmentation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[40] V. Guizilini, R. Hou, J. Li, R. Ambrus, and A. Gaidon, "Semantically-guided representation learning for self-supervised monocular depth," in *International Conference on Learning Representations (ICLR)*, 2020.

[41] P.-Y. Chen, A. H. Liu, Y.-C. Liu, and Y.-C. F. Wang, "Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[42] A. Atapour-Abarghouei and T. P. Breckon, "Monocular segment-wise depth: Monocular depth estimation based on a semantic segmentation prior," in *International Conference on Image Processing (ICIP)*, 2019.

[43] L. Wang, J. Zhang, O. Wang, Z. Lin, and H. Lu, "SDC-Depth: Semantic divide-and-conquer network for monocular depth estimation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[44] C. Häne and M. Pollefeys, "An overview of recent progress in volumetric semantic 3D reconstruction," in *International Conference on Pattern Recognition (ICPR)*, 2016.

[45] I. Cherabier, J. L. Schonberger, M. R. Oswald, M. Pollefeys, and A. Geiger, "Learning priors for semantic 3D reconstruction," in *European Conference on Computer Vision (ECCV)*, 2018.

[46] A. Dai, D. Ritchie, M. Bokeloh, S. Reed, J. Sturm, and M. Nießner, "ScanComplete: Large-scale scene completion and semantic segmentation for 3D scans," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[47] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[48] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *International Conference on Computer Vision (ICCV)*, 2019.

[49] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[50] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3D.net: A large-scale point cloud classification benchmark," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-1-W1, pp. 91–98, 2017.

[51] X. Roynard, J.-E. Deschaud, and F. Goulette, "Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification," *The International Journal of Robotics Research (IJRR)*, vol. 37, no. 6, pp. 545–557, 2018.

[52] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese, "Joint 2D-3D-semantic data for indoor scene understanding," 2017, arXiv preprint arXiv:1702.01105.

[53] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3D: Learning from RGB-D data in indoor environments," in *International Conference on 3D Vision (3DV)*, 2017.

[54] A. Boulch, G. Puy, and R. Marlet, "NeeDrop: Self-supervised shape representation from sparse point clouds using needle dropping," in *International Conference on 3D Vision (3DV)*, 2021.

[55] F. Bogo, J. Romero, G. Pons-Moll, and M. J. Black, "Dynamic FAUST: Registering human bodies in motion," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[56] G. Riegler, A. O. Ulusoy, H. Bischof, and A. Geiger, "OctNetFusion: Learning depth fusion from data," in *International Conference on 3D Vision (3DV)*, 2017.

[57] D. Stutz and A. Geiger, "Learning 3D shape completion from laser scan data with weak supervision," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[58] B.-S. Hua, Q.-H. Pham, D. T. Nguyen, M.-K. Tran, L.-F. Yu, and S.-K. Yeung, "SceneNN: A scene meshes dataset with annotations," in *International Conference on 3D Vision (3DV)*, 2016.

[59] H. Fu, B. Cai, L. Gao, L.-X. Zhang, J. Wang, C. Li, Q. Zeng, C. Sun, R. Jia, B. Zhao, and H. Zhang, "3D-FRONT: 3D furnished rooms with layouts and semantics," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[60] H. Fu, R. Jia, L. Gao, M. Gong, B. Zhao, S. Maybank, and D. Tao, "3D-FUTURE: 3D furniture shape with texture," *International Journal on Computer Vision (IJCV)*, 2021.

[61] P. Selvaraju, M. Nabail, M. Loizou, M. Maslioukova, M. Averkiou, A. Andreou, S. Chaudhuri, and E. Kalogerakis, "BuildingNet: Learning to label 3D buildings," in *International Conference on Computer Vision (ICCV)*, 2021.

[62] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[63] O. Cicek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2016.

[64] L. Yi, B. Gong, and T. Funkhouser, "Complete & Label: A domain adaptation approach to semantic segmentation of LiDAR point clouds," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[65] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," in *SIGGRAPH*, 1987.