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

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

VII. Labeling points in space from BIM meshes

BIM data include individual meshes for each building component (or building "object"), together with their semantics. In practice, although (or because) they are made by humans, these meshes often have defects: they may be non-watertight, self-intersecting, overlapping, etc.

Section III.B of the main paper describes an algorithm to transform this kind of data into a volumetric function that gives a semantic label (including *void* for emptyness) to any 3D point in space. We provide here, on Figure 6, a pseudo-code for this point labeling algorithm.

The notation $\mathrm{Ray}(p, \mathbf{e}_x)$ designates the infinite ray starting from point p, and parallel to the x-axis $\mathbf{e}_x$. For a perfectly closed surface, this ray intersects the surface an odd number of times if and only if p is in the surface interior. Since a minority of meshes are not perfectly closed in the original source BIM models of VASAD, we make use of of only three axis-aligned rays per point and use a majority vote to decide whether p is in the interior of an object. In practice, we also noticed little to no overlap between the different BIM components. Therefore, we assign an interior p with the label of the first mesh for which p is decided to be in the interior of.

VIII. Generation of Viewpoints

Section III.C of the main paper describes how to automatically choose viewpoints to create artificial scans. We provide on Figure 7 a pseudo-code for this viewpoint generation.

This algorithm is based on the observation that a set of viewpoints that are invisible from each other are likely to scan different parts of the surface; they thus have good chances to be well distributed in space to scan the building.

Our procedure is iterative and greedy. We set a budget of N_{iter} trials to find a new viewpoint which is in the void space and which is not seen from already positioned viewpoints. If we are not able to find such a new location after N_{iter} trials, the algorithm stops. Otherwise, we keep on adding viewpoints until we select N_{max} viewpoints.

Please note that, even if two viewpoints do not see each other, they may see a few building parts that are the same. Scans from such a set of viewpoints may thus have overlapping areas, as is the case in practice for human-based acquisition campaigns in real buildings, that try to minimize the number of scans while ensuring a maximum coverage of the building surface.

As said in the main paper, this procedure does not ensure that every bit of surface is visible from some viewpoint. To reduce the likeliness of having parts of the building that are not seen by any viewpoint, one can increase N_{iter} and N_{max}. However, because of the heuristic nature of the procedure, it becomes "asymptotically" harder to sample a point in empty space that is not visible from previous points. In this case, it could make more sense to generate a few full sets of viewpoints and to merge them, although it would most likely break the sensor reciprocal invisibility rule. Complementary viewpoint candidates could also be created by sampling points uniformly on the surface and offsetting them from the surface along the normal direction, but it remains time consuming.

In any case, it is unrealistic to try to ensure that every bit of surface is visible from at least one viewpoint; there would be too many of them, and in unpractical locations. For instance, in real life, no actual scan sees or tries to see behind or below banister rails, or every part of a ceiling with exposed beams. Learned priors have to be robust to input data incompleteness, thanks to a supervision that, on the contrary, operates on complete data. It is the case for VASAD as supervision provides exact answers to any point query (exact up to the vote heuristics of point labeling in case input meshes have defects, see Section VIII).

IX. Metrics

In this section, we provide a more detailed formulation of the metrics used in the main paper.

Our metrics measure volumetric information. Evaluation is carried out by uniformly sampling points $\mathcal{P}$ in space in the union of the bounding box of the ground truth and the predicted 3D models. In practice, we sample 10M points and we observe that increasing this number of points does not affect the metrics noticeably. (It is similar to point sampling to estimate Chamfer distances.) For each point $p \in \mathcal{P}$, we then find the ground-truth label l_p^{gt} thanks to the algorithm in Section VII, and we consider the predicted label l_p^{pred}.

As most points in space are void (typically 70%), we do not want to the metrics to be dominated by a measure of correct empty space labeling. The metrics is thus computed

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

```
INPUT: a 3D point $p$ and a mesh $\mathcal{M}$ of the BIM model, structured as a set of object meshes $\mathcal{M}_s$ with a semantic label
label = $label(void)$            // Initialize the point label to void (i.e., empty)
queries ← {Ray($p, \mathbf{e}_x$), Ray($p, \mathbf{e}_y$), Ray($p, \mathbf{e}_z$)}  // Consider 3 orthogonal directions
for $\mathcal{M}_s \in \mathcal{M}$ do            // For every semantic instance $\mathcal{M}_s$ in the global mesh
  $n_{\textit{odd-inter}} \leftarrow 0$            // Initialize the counter of rays having an odd number of intersections with the object mesh $\mathcal{M}_s$
  for query ∈ queries do
    if $|\text{query} \cap \mathcal{M}_s| \mod 2 = 1$ do    // Count one for each query ray having an odd number of intersections with $\mathcal{M}_s$
      $n_{\textit{odd-inter}} \leftarrow n_{\textit{odd-inter}} + 1$
    end if
  end for
  if $n_{\textit{odd-inter}} \geq 2$ do            // If at least two rays vote for being inside $\mathcal{M}_s$
    label ← $label(\mathcal{M}_s)$            // Assign the label of $\mathcal{M}_s$
    break
  end if
end for
OUTPUT: label for point $p$
```

Fig. 6. Algorithm to label points in space from possibly defect-laden BIM meshes.

```
INPUT: mesh $\mathcal{M}$ representing the whole BIM model.
$\mathcal{V} \leftarrow \emptyset$                    // Initialize the set of viewpoints as empty
while $|\mathcal{V}| < N_{\mathsf{max}}$ do
  repeat $N_{\mathsf{iter}}$ times
    //// Sample a candidate viewpoint in void
    Pick $v \in bbox(\mathcal{M})$          // Uniform sampling
    while $label(v) = void$ do  // Using the algorithm described in Figure 6
      Pick $v \in bbox(\mathcal{M})$
    end while
    is_valid ← true
    for $v_c \in \mathcal{V}$ do                // Compare the current viewpoint to those we already have
      if segment$(v, v_c) \cap \mathcal{M} = \emptyset$ then    // If the viewpoints "see" each other
        is_valid ← false        // Then the candidate viewpoint is discarded
        break
      end if
    end for
    if is_valid then
      $\mathcal{V} \leftarrow \mathcal{V} \cup \{v\}$
      break
    end if
  end repeat
end while
OUTPUT: set of point of views $\mathcal{V}$
```

Fig. 7. Algorithm to generate synthetic viewpoints.

over all material points in $\mathcal{P}$, i.e., all points in $\mathcal{P}$ whose label corresponds to full (i.e., non void), either in the ground truth or in the prediction. The metrics are defined as follows, where $\varnothing$ denotes the void label.

A. Semantic Intersection over Onion (Sem. IoU)

The semantic IoU is the proportion of points that have the same label among the set of points labeled as non empty in either the ground truth and the prediction.

$$\text{IoU}_{\text{sem}} = \frac{\left|\{p \in \mathcal{P} \mid l_p^{\text{pred}} = l_p^{\text{gt}} \neq \varnothing\}\right|}{\left|\{p \in \mathcal{P} \mid l_p^{\text{pred}} \neq \varnothing \text{ or } l_p^{\text{gt}} \neq \varnothing\}\right|} \quad (1)$$

B. Geometric Intersection over Onion (Geom. IoU)

Some methods can predict a correct occupancy of the 3D space (i.e., a correct partition into binary labels void and non void) while they misclassify the full space (e.g., mistake a window for a wall). In this case, even if the semantics is

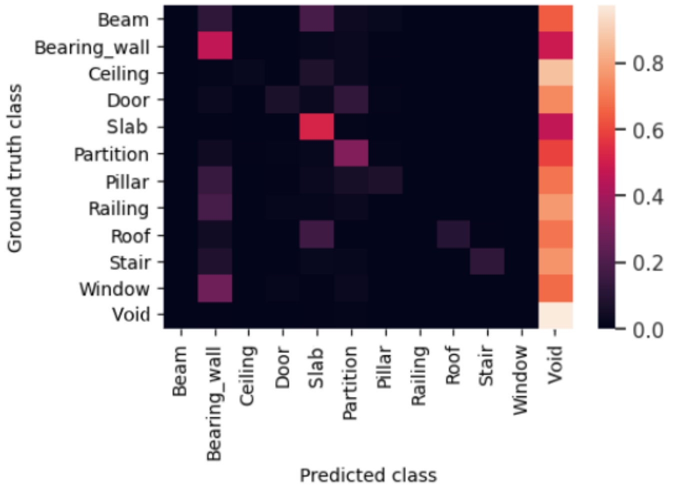


Fig. 8. Confusion matrix for SemConvONet (with normals as input).

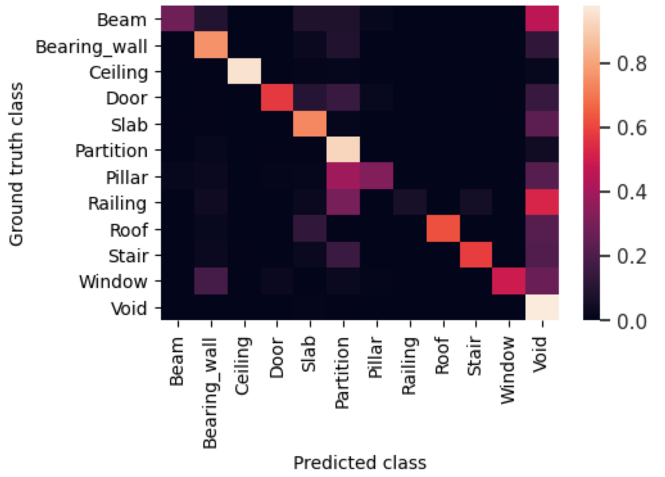


Fig. 9. Confusion matrix for our PVSRNet method (with normals as input).

wrong, the geometry still makes sense. We evaluate it thanks to the following geometric IoU (which is the standard IoU in dense 3D):

$$\mathrm{IoU}_{\mathrm{geom}} = \frac{\left|\{p \in \mathcal{P} \mid l_p^{\mathrm{pred}} \neq \varnothing \text{ and } l_p^{\mathrm{gt}} \neq \varnothing\}\right|}{\left|\{p \in \mathcal{P} \mid l_p^{\mathrm{pred}} \neq \varnothing \text{ or } l_p^{\mathrm{gt}} \neq \varnothing\}\right|} \quad (2)$$

C. Object Instance Evaluation

There are numerous ways to split most semantic materials into individual parts, e.g., dividing walls into different cuboids where two or more walls "encounter". In fact, there is no convention. Architects differ widely in their choices. A few of them sometimes thus hint at the underlying structure of the building, or the intended construction. Therefore, assessing an instance-based decomposition quality for these semantic classes is ill-defined.

Consequently, in VASAD, we decided we would not try to evaluate any instance segmentation of building components, even for classes as doors or windows for which connected components can easily create instances. We thus do not assess either any relationship between building components, as exists in real BIM files. We only evaluate the presence of the right volume with the right label.

X. Confusion Matrices

To better compare the SemConvONet baseline method to our PVSRNet approach, we study here classwise results.

As classification errors are usually not evenly spread across the different classes, we quantitatively represent these errors by building a confusion matrix C whose terms $C_{i,j}$ count the number of points in $p \in \mathcal{P}_{\mathrm{vol}}$ whose ground-truth class is i, and whose predicted class is j, including the void class.

$$C_{i,j} = \left|\{p \in \mathcal{P} \mid l_p^{\mathrm{gt}} = i \text{ and } l_p^{\mathrm{pred}} = j\}\right| \quad (3)$$

A perfect classification would therefore yield a diagonal confusion matrix.

Concretely, we show on Figure 8 the confusion matrix for SemConvONet and, on Figure 9, the confusion matrix for PVSRNet. For the sake of visualization, each row C_i is normalized by the total number of points whose ground-truth class is i, i.e., $\sum_j C_{i,j}$.

The confusion matrices highlight the fact that, in spite of achieving a reasonable recovery of the major building components (bearing walls, slabs, partitions), SemConvONet fails at recovering smaller classes, which are not reconstructed (points wrongly labeled as void).

Conversely, PVSRNet is able to properly classify most of the classes. Yet, we observe remaining errors. Most of them come from:

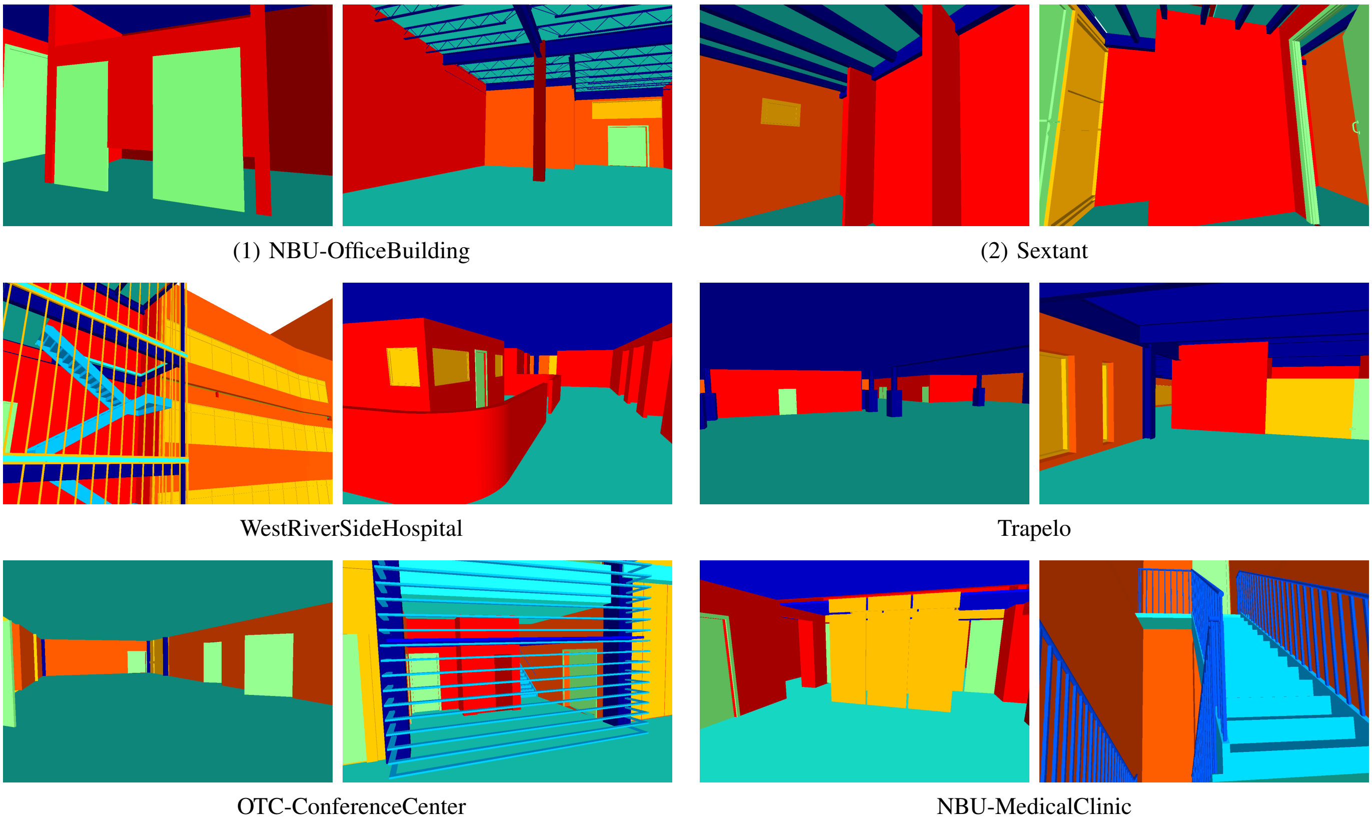
- small or little-represented classes, i.e., railings or beams,
- classes that are hard to separate, e.g., windows on walls,
- architectural ambiguities (which may be labeled differently from one BIM model to another one, sometimes even inconsistently within the same model), e.g., deciding between a flat roof and a slab, or between a pillar inside a building and a wall (including partitions).

XI. Semantized Reconstruction Methods

Very few methods can reconstruct both volume and semantics. We forgot to cite in the main paper the pioneering of work of Häne and colleagues [1], [2], [3], although we cite their survey [4]. Another very recent (yet unpublished) and interesting approach is [5], but it take as input single images.

In this paper, we introduce two methods (with variants) for the task of volumetric and semantic reconstruction from point clouds, that are the first to operate at large scale and with available code (upon publication).

- *SemConvONet* is an extension of ConvONet [6] to also provide semantics.
- *PVSRNet*, which we put forward, pipelines point convolution for semantics [7] and a 3D U-Net [8]. The 3D U-Net that we use is actually a modern, more powerful and scalable variant of SSCNet [9] (whose code relied on obsolete pycaffe).


(1) NBU-OfficeBuilding

(2) Sextant

WestRiverSideHospital

Trapelo

OTC-ConferenceCenter

NBU-MedicalClinic

Fig. 10. Additional visuals of the VASAD dataset

XII. The VASAD Dataset

A. Empty vs furnished rooms.

Some initial BIM files of VASAD were featuring some furniture, but the room contents and distribution was little realistic, and we could not gather a large-enough dataset of this kind anyway. We thus decided to remove this clutter to have a homogeneous dataset. Populating it automatically afterwards with furniture, as in Synthetic Rooms [6] or using a similar approach as SceneNet [10]), is doable, but making it realistic is substantial future work.

Even without furniture, VASAD remains anyway valuable for training and evaluating methods reconstructing volume and semantics. It is also valuable for the renovation industry, when a building is fully cleaned up before being restructured. Besides, most scan-to-BIM methods start by removing clutter using point cloud semantic segmentation; VASAD-based learned methods are then applicable to the cleaned-up point clouds.

B. Dataset Diversity

VASAD buildings are very different one from another, in purpose and in design (from a residential villa to a large hospital). Besides inter-model variety, the intra-model diversity is also high, as illustrated in Figure 1 of the main paper and in Figure 10 of this supplementary material: room sizes, irregular configurations, wall types, window shapes, staircases, etc.

Dataset	S3DIS	CVBE-3D	VASAD
Scan	real	real	simulated
Rooms with furniture	✓	✓	✗
No. buildings	3	?	6
No. floors	6	8	24
Size (m^2)	6,000	?	62,000
Volume information	✗	✓	✓
3D reconstruction type	-	abstract	actual
No. sem. classes (build. compon.)	7	7	11
Evaluation	auto.	human	auto.

TABLE III
Comparison of Semantized 3D Buidling Datasets.

XIII. Other Datasets

A few publications use homemade data with volume and semantics, that are not publicly available.

The recent (June 2021) CVPR workshop on Computer Vision in the Built Environment proposed a challenge with a dataset (CVBE-3D) [11]. We did not succeed in getting it (never hearing back from our account creation request) and it looks like it has not been available since the workshop took place. Yet, given what can be known from this dataset [11], VASAD is notably larger and more complex than CVBE-3D, as can be seen from Table III, where we include also S3DIS [12]. Besides the dataset itself, VASAD also includes tools to create more training and testing data from BIM files.

Concretely, CVEB-3D offers real scans of furnished buildings, while VASAD provides simulated scans with empty rooms. It is significantly smaller than VASAD (8 vs 24 floors) and features less semantic classes: no difference between bearing walls and partitions; no roof, staircases, or railing. (CVBE-3D actually is a subset of the larger CVBE dataset used for the 2D challenge, which is made of 91 floors from 31 buildings.)

Besides, the evaluation has to be done by a human expert. In fact, the evaluation is not possible anymore now that the challenge is finished (assuming training and testing data could be downloaded).

Another difference is that the 3D ground truth in CVEB-3D may abstract away from the actually-scanned geometry, giving a high-level, simplified representation of reality, whereas our volumes are closely aligned with the scans, since the scans are synthetically made from the volumes. While abstraction is good for visualization and planning, faithful geometry is better for actual renovation work.

XIV. Data and Code Release

At https://github.com/palanglois/vasad, we provide not only the VASAD dataset but also the full implementation of the methods and the data preparation tools. These tools produce usable volumetric and semantic data from raw IFC files. It will also allow the dataset to be enriched by more BIM models.

In fact, it could even be applicable to other contexts like interior reconstruction with furniture, based on datasets such as 3D-FRONT [13].

References

[1] C. Häne, C. Zach, A. Cohen, R. Angst, and M. Pollefeys, "Joint 3D scene reconstruction and class segmentation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[2] N. Savinov, C. Häne, L. Ladický, and M. Pollefeys, in *Conference on Computer Vision and Pattern Recognition (CVPR).*

[3] C. Häne, C. Zach, A. Cohen, and M. Pollefeys, "Dense semantic 3D reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 39, no. 9, pp. 1730–1743, 2017.

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

[5] A.-Q. Cao and R. de Charette, "MonoScene: Monocular 3D semantic scene completion," 2021.

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

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

[8] 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.

[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] 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.

[11] F. Li, Y. Turkan, M. Olsen, A. Morris, Y. Cho, R. Chen, J. Jung, E. Che, W. Wu, I. Armeni, M. Fischer, D. Hall, M. Pollefeys, and S. Savarese, "1st workshop and challenge on computer vision in the built environment for the design, construction and operation of buildings (cvpr workshop)," 2021. [Online]. Available: https://cv4aec.github.io

[12] 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.

[13] 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.