AtlantaNet: Inferring the 3D Indoor Layout from a Single 360° Image beyond the Manhattan World Assumption

Giovanni Pintore[1 [0000−0001−8944−1045]], Marco Agus[2,1 [0000−0003−2752−3525]], and Enrico Gobbetti[1 [0000−0003−0831−2458]]

[1] Visual Computing, CRS4, Italy
giovanni.pintore@crs4.it enrico.gobbetti@crs4.it
[2] College of Science and Engineering, HBKU, Qatar
magus@hbku.edu.qa

Fig. 1. Examples of automatically recovered 3D layouts. Our method returns a 3D room model from a single panorama even in cases not supported by current state-of-the-art methods, such as, for example, vertical walls meeting at non-right angles or with a curved 2D footprints.

Abstract. We introduce a novel end-to-end approach to predict a 3D room layout from a single panoramic image. Compared to recent state-of-the-art works, our method is not limited to *Manhattan World* environments, and can reconstruct rooms bounded by vertical walls that do not form right angles or are curved – i.e., *Atlanta World* models. In our approach, we project the original gravity-aligned panoramic image on two horizontal planes, one above and one below the camera. This representation encodes all the information needed to recover the *Atlanta World* 3D bounding surfaces of the room in the form of a 2D room footprint on the floor plan and a room height. To predict the 3D layout, we propose an encoder-decoder neural network architecture, leveraging Recurrent Neural Networks (RNNs) to capture long-range geometric patterns, and exploiting a customized training strategy based on domain-specific knowledge. The experimental results demonstrate that our method outperforms state-of-the-art solutions in prediction accuracy, in particular in cases of complex wall layouts or curved wall footprints.

Keywords: 3D floor plan recovery, panoramic images, 360 images, data-driven reconstruction, structured indoor reconstruction, indoor panorama, room layout estimation, holistic scene structure

1 Introduction

Automatic 3D reconstruction of a room's bounding surfaces from a single image is a very active research topic [20].

In this context, 360° capture is very appealing, since it provides the quickest and most complete single-image coverage and is supported by a wide variety of professional and consumer capture devices that make acquisition fast and cost-effective [31]. Since rooms are full of clutter, single images produce anyway only partial coverage and imperfect sampling, thus reconstruction problem is difficult and ambiguous without prior assumptions. In particular, current approaches, see Sec. 2, are either tuned to simple structures with a limited number of corners [6] or bound by the *Indoor World* assumption [16] (i.e., the environment has a single horizontal floor and ceiling, and vertical walls which all meet at right angles). In this context, recent data-driven approaches [33,26,30] have produced excellent results in recovering the room layout from a single panoramic image [34]. However, state-of-the-art data-driven methods usually follow a costly and constraining framework: a heavy pre-processing to generate Manhattan-aligned panoramas (e.g., edge-based alignment and warping of generated perspective views [16]), a deep neural network that predicts the layout elements on a rectified equirectangular image, and a post-processing that fits the (Manhattan) 3D layout to the predicted elements.

In this work, we present *AtlantaNet*, a novel data-driven solution to estimate a 3D room layout from a single RGB panorama. As its name suggests, we exploit the less restrictive *Atlanta World* model [23], in which the environment is expected to have horizontal floor and ceiling and vertical walls, but without the restriction of walls meeting at right angles or having a limited number of corners (supporting, e.g., curved walls). In our approach, the original equirectangular image, assumed roughly aligned with the gravity vector, is projected on two arbitrary horizontal planes, one above and one below the camera (see Fig. 2(a)). Exploiting the *Atlanta World* assumption, this representation encodes all the information needed to recover 3D bounding surfaces of the room, i.e., the 2D floor plan and the room height (see Sec. 4). To predict the 3D layout from this representation, we propose an encoder-decoder architecture, leveraging Recurrent Neural Networks (RNNs) to capture the long-range geometric pattern of room layouts. The network maps a projected image, represented as a tensor, to a binary segmentation mask separating the interior and exterior space, respectively for the ceiling and for the floor projection. The walls footprint is found by extracting a polygonal approximation of the contour of the mask generated from the above-camera image (ceiling mask), and the room height is determined by the scale that maximizes the correlation between the lower and upper contour (see Sec. 4). A customized training strategy based on domain-specific knowledge makes it possible to perform data augmentation and reuse the same network for both projected images. For training, we exploit previously released annotated datasets [33,26,30,6]. Our experimental results (see Sec. 5) demonstrate that our method outperforms state-of-the-art methods [33,26,30] in prediction accuracy, especially on rooms with multiple corners or non-Manhattan layouts. Fig. 1 shows some 3D layouts predicted by our method.

Our contributions are summarized as follows:

- We introduce a *data encoding based on the* Atlanta World *indoor model*, that allows layout prediction on planar projections free from spherical image deformations, unlike previous approaches that are predominantly based on features extracted from the equirectangular view [33,26,30,6]. As supported by results, working on such a transformed domain simplifies structure detection [19,21,30]. In addition, representative tensors can be treated as conventional 2D images, simplifying, for example, data augmentation and the use of powerful network architectures such as RNNs [2,1].
- We reconstruct the 3D layout, in terms of 2D footprint and room height, by *inferring the 2D layout from the contour of a solid segmentation masks and the room height from the geometric analysis of the correlation between two contours*. Our approach is more stable and well suited to modeling complex structures, such as curved walls, than previous approaches that infer layout from sparse corner positions [33,26,6]. Moreover, we do not need an additional dense network [30] or a post-processing voting scheme [26] to infer the layout height, which can directly determined from a geometric analysis of the masks.
- We propose an *end-to-end network* that, differently from current state-of-the-art approaches [33,26,30], does not require heavy pre-processing, such as detection of main Manhattan-world directions from vanishing lines analysis [34,32,16] and related image warping, nor complex layout post-processing, such as Manhattan-world regularization of detected features [33,26,30]. Our only requirement is that input images are roughly aligned with the gravity vector, a constraint which is easily met by hardware or software means [9], and is verified in all current benchmark databases. As a result, our method, in addition to being faster, does nor require complex per-image deformations that make multi-view analysis difficult (see discussion in Sec. 6).
- We propose a *training strategy* based on feeding both ceiling and floor view on the same network instance, improving inference performance compared to a dual joined branches architecture or on separate training for ceiling and floor (see results and ablation study at Sec. 5.3).

We tested our approach on both conventional benchmarks (see Zou et al. [34]) and more challenging non-Manhattan scenes annotated by us (see Sec. 5.1). Results demonstrate how our method outperforms previous works on both testing sets (Sec. 5). Code and data are made available at *https://github.com/crs4/AtlantaNet*.

2 Related work

3D reconstruction and modeling of indoor scenes has attracted a lot of research in recent years. Here, we analyze only the approaches closer to ours, referring the reader to a very recent survey for a general coverage of the subject [20].

A noticeable series of works concentrate on parsing the room layout from a single RGB image. Since man-made interiors often follow very strict rules, several successful approaches have been proposed by imposing specific priors.

Delage et al. [4] presented one the first monocular approaches to automatically recover a 3D reconstruction from a single indoor image. They adopt a dynamic Bayesian network trained to recognize the *floor-wall* boundary in each column of the image, assuming the indoor scene consists only of a flat floor and straight vertical walls. However, in its original formulation, such a reconstruction is limited to partial views (e.g., a room corner).

Full-view *geometric context* (GC) estimation from appearance priors, i.e., the establishment of a correspondence between image pixels and geometric surface labels, was proposed as a method to analyze outdoor scenes by Hoiem et al. [13]. In combination with Orientation Maps (OM) [16], which are map of local belief of region orientations computed from line segments through heuristic rules, GC is the basis for almost all methods based on geometric reasoning on a single image. Hedau et al. [12], in particular, successfully analyzed the labeling of pixels under the cuboid prior, while Lee et al. [16] considered the less constraining *Indoor World Model* (IWM), i.e., a *Manhattan World* with single-floor and single-ceiling, by noting that projections of building interiors under the Indoor World can be fully represented by corners, so a valid structure can be obtained by imposing geometric constraints on corners. Such a geometric reasoning on IWM supports several efficient reconstruction methods. A notable example is the work of Flint et al. [8,7], who, exploiting the *homography* between floor and ceiling, reduce the structure classification problem to the estimation of the y-coordinate of the ceiling-wall boundary in each image column.

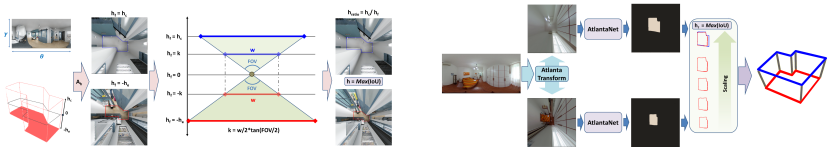
One of the main limitations of single-image methods lies, in fact, on the restricted field of view (FOV) of conventional perspective images, which inevitably results in a limited geometric context [32]. With the emergence of consumer-level 360° cameras, a wide indoor context can now be captured with one or at least few shots. As a result, most of the research on reconstruction from sparse imagery is now focused in this direction. Zhang et al. [32] propose a whole-room 3D context model that maps a full-view panorama to a 3D bounding box of the room, also detecting all major objects inside (e.g, *PanoContext*). By combining OM for the top part and GC for the bottom part, they demonstrate that by using panoramas, their algorithm significantly outperforms results on regular-FOV images. More recently, Xu et al. [27] extended this approach of by assuming IWM instead of a box-shaped room, thus obtaining a more accurate shape of the room, and Yang et al. [28] proposed an algorithm that, starting from a single full-view panorama, automatically infers a 3D shape from a collection of partially oriented super-pixel facets and line segments, exploiting the *Manhattan World* constraint. Pintore et al. [19] tackle the problem of recovering room boundaries in a *top-down 2D* domain, in a manner conceptually similar to that of dense approaches. To recover the shape of the room from the single images they combine the ceiling-floor homography [8] to a spatial transform (E2P - i.e., *equirectangular to perspective*) [19], based on the *Unified projection model* for spherical images [10]. Such E2P transform highlights the shape of the room projected on a 2D floorplan, generating two projections, respectively for the floor and for the ceiling edges. Applying the ceiling-floor homography, they

recover the height of the walls and enforce the 2D shape estimation from the projected contours. As for all feature-based methods, the effectiveness of these approaches depend on the quality of extracted features (e.g., edges or flat uniform patches). To overcome these problems, more and more solutions are turning towards data-driven approaches [34].

The peculiarity of indoor reconstruction makes generic segmentation solutions (e.g., U-Net [22] or DeepLab [3]) not appropriate. In particular, defining a graphical model at the pixel-level makes it hard to incorporate global shape priors. Recent data-driven approaches have demonstrated impressive performance in recovering the 3D boundary of a single room meeting the Manhattan World constraint. Zou et al. [33] predict the corner probability map and boundary map of directly from a panorama (e..g, *LayoutNet*). They also extend Stanford 2D-3D dataset [25] with annotated layouts for training and evaluation. Yang et al. [30] propose a deep learning framework, called *DuLa-Net*, which exploits features fusion between the original panoramic view and the ceiling E2P transform [19], to output a floor plan probability map. A Manhattan regularization step is then performed to recover the 2D floor plan shape, through a grid aligned to the main Manhattan axes. Similarly to *LayoutNet* approach [33], a number of recent works [6,26] focus on inferring the room layout from the sparse corners position in the panoramic image. Sun et al. [26] represent room layout as three 1D vectors that encode, at each image column, the boundary positions of floor-wall and ceiling-wall, and the existence of wall-wall boundary. The 2D layout is then obtained by fitting Manhattan World segments on the estimated corner positions.

Recently, Zou et al. [34] have presented an extensive evaluation of the latest high-performance methods. In their classification, such methods basically share the same pipeline: a Manhattan World pre-processing step (e.g., based on Zhang et al. [32]), the prediction of layout elements and a post-processing for fitting the 3D model to the predicted elements after a series of regularization. Differently to almost all recent methods [33,26,30], we do not need complex pre-processing steps, such as computation of Manhattan vanishing lines [16] and warping the panoramic image according to them, but only perform projection along the gravity vector. While our method, like many recent ones, shares with HorizonNet [26] and Dula-Net [30] the encoder-decoder concept, we introduce important novelties in the network architecture. In particular, HorizonNet fully works in a 1D domain derived from the equirectangular projection, while we work entirely in a 2D domain derived from projections on horizontal planes. Moreover, in contrast to Dula-Net, we use a single branch working in the transformed domain (both for floor and ceiling), while Dula-Net uses two parallel branches for the ceiling-view probability and for ceiling-floor probability in the equirectangular domain, plus an additional linear branch for deriving the height. Our results show the advantages of our solution. Furthermore, in contrast to many other works, we predict the room layout from dense 2D segmentation maps by simply extracting the largest connected component, rather than from a sparse number of inferred corner positions [6,26]. Such an approach is more robust, particularly in cases of non-Manhattan shapes.

3 Overview



(a) *Dataencoding* (b) *Layout recovery*

Fig. 2. Data encoding and layout prediction. Fig. 2(a): the *Atlanta Transform* A_h maps all the points of the equirectangular image in 3D space as if their height was h_f (focal height), where h_f can assume only two possible values: $-h_e$ (eye height) and h_c (ceiling height). Since at least h_c is an unknown value, we apply the transform by imposing a single, fixed, h_f, which depends by a fixed FOV. Fig. 2(b): We infer through our network the ceiling or floor shapes. The height is directly proportional to the ratio h_r (height ratio) between these shapes, and the 2D footprint of the room is recovered from the ceiling shape.

Our method takes as input a single panoramic image, that we assume aligned to the gravity vector. This is easily obtained on all modern mobile devices that have an IMU on board, or can be achieved prior to the application of our pipeline through standard image processing means [9]. Starting from the oriented image, our approach, depicted in Fig. 2 determines the room structure.

The first module generates, from the input equirectangular image (e.g., panorama original size), an *Atlanta Transform* (e.g., $3 \times 1024 \times 1024$) on two horizontal planes placed above and below the camera. For training, the ground truth annotations, conventionally provided on a panoramic image, are transformed in the same way. To simplify discussion, we call the projection on the upper plane the *ceiling projection*, and the projection on the lower plane the *floor projection*. Note, however, that the selected planes do not need to be exactly corresponding to the ceiling or for the floor plane, since the room dimensions are determined automatically by our method and are not known in advance.

During training, the network (see Fig.3) is fed by alternating ceiling or floor images, according to a probability function (see Sec. 4.3 and Sec. 5.3). In prediction mode, the same trained network is used to infer ceiling or floor shapes.

The height of the layout is directly proportional to the ratio h_r between the ceiling shape and the floor shape (i.e., a scaling factor). Since in real cases, the floor shape is partially occluded by the clutter, we assume as inferred h_r the value that maximizes the intersection-over-union between the contours of the ceiling and the floor shapes (see Fig. 2(b)).

On output, the 2D shape of the room is simply the contour of the largest connected region of the mask resulting from the network, without applying any post-process regularization, as opposed to, e.g., solutions based on Manhattan-world constraints. The final 3D layout is then determined by extruding a 2D shape from the ceiling shape using the recovered layout height.

4 Approach

4.1 Data encoding

Assuming the *Atlanta World* model [23], we project the panoramic image on two horizontal planes, building, respectively, one representative tensor (i.e., $3 \times 1024 \times 1024$) for the ceiling and one for the floor horizontal plane (see Fig. 2(b)). To transform the equirectangular map we adopt the following relation:

$$A_h(\theta, \gamma, h_f) = \begin{cases} x = h_f / \tan\gamma * \cos\theta \\ y = h_f / \tan\gamma * \sin\theta \\ z = h_f \end{cases} \quad (1)$$

The function A_h, called *Atlanta Transform*, maps all the points of the equirectangular image in 3D space as if their height was h_f [19]. Compared to a classic pin-hole model, h_f can be seen as the focal length for a 180 degree field-of-view. In the specific case of the Atlanta World model, h_f can assume only two possible values: $-h_e$, that is the floor plane below camera center, and h_c, that is the distance between the camera center and the ceiling plane (see Fig. 2(a)).

Considering h_e a known constant or at most fixed as a scale factor, the 3D layout of an Atlanta model is fully defined by a two-dimensional shape - i.e. the 2D footprint of the layout on the floorplan, and by the ceiling distance h_c. Ideally in order to directly apply equation 1 we should also know the value of h_c. Since, in our case, h_c is unknown before reconstruction and must be inferred by the network, we apply a modified version of the transform [30] by imposing a single, fixed, h_f, which depends by a fixed field-of-view (FOV), i.e., $h_f = w/2 * \tan(FOV/2)$, where $w \times w$ is the extent in pixels of each transform (that we assume square). As a consequence, the height of the room is determined by the ratio between h_c and h_e, and is directly proportional to the ratio h_r between the ceiling shape and the floor shape. Ideally, h_r should be the value that makes the floor shape match with the ceiling shape. Since in real cases, the floor shape is heavily occluded by clutter, we assume as inferred h_r the value that maximizes the intersection-over-union between the contours of the ceiling and the floor shapes (see Fig. 2(b)).

4.2 Network architecture

Fig. 3 shows an overview of *AtlantaNet*. The network takes as an input a transform of size $3 \times w \times w$ (see Sec. 4.1) and produces a segmentation mask of size $1 \times w \times w$. We tested different sizes for the input transform, and we found that 1024×1024 is the best size in terms of performance, so as to guarantee sufficient detail for the most complex forms and not to require large memory resources (see Sec. 5). The size of the output is $1 \times 1024 \times 1024$, that is a binary segmentation mask describing the ceiling or floor shape. We adopt *ResNet* [11] as feature extractor, which has proven to be one of the most effective encoder for both panoramic and perspective images [34]. The output of each *ResNet* block has half spatial resolution compared to that of the previous block. To capture both low-level

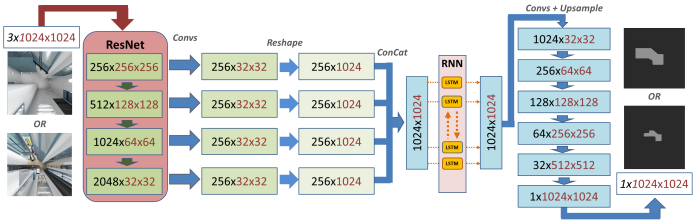


Fig. 3. Network architecture. The network takes as input a transforms of size $3 \times w \times w$ (see Sec. 4.1) and passes it to a ResNet encoder. To capture both low-level and high-level features, we keep the last four feature maps of the encoder. Each feature map is then reduced to the same size, $256 \times 32 \times 32$, through a sequence of convolutional layers and reshaped to 256×1024. The 4 features maps are concatenated to a *sequential feature map* of 1024×1024. We feed such a sequence to a RNN, obtaining, after a reshaping, a $1024 \times 32 \times 32$ map. We upsample such map to recover a $1 \times 1024 \times 1024$ binary segmentation mask describing the ceiling or floor shape.

and high-level features, we keep the last four feature maps of the encoder [26]. Each feature map is then reduced to the same size, $256 \times 32 \times 32$, through a sequence of convolutional layers (*Convs* in Fig. 3), where each layer contains: a 2D convolution having stride 2 (e.g., except for the last block, having stride 1), a batch normalization module and a rectified linear unit function (ReLU). Finally, we reshape the 4 features maps to 256×1024, and we concatenate them layers to obtain a single *sequential feature map* of 1024×1024 (i.e., 1024 layers for a sequence having length 1024).

We feed such a sequence to a RNN, that is exploited to capture the shape of the object and thus make coherent predictions even in ambiguous cases such as occlusions and cluttered scenes. In particular, we employ convolutional LSTM [24] modules in our model as the decoder core. Specifically we adopt a bi-directional LSTM with 512 features in the hidden state and 2 hidden internal layers. The output of the RNN decoder is a 1024×1024 feature map, which collect all the time steps of the RNN layers.

We reshape the RNN output to $1024 \times 32 \times 32$, and, after a a drop-off, we up-sample it through a sequence of 6 convolutional layers (same of *Convs* but with stride 1) each one followed by an interpolation (e.g., factor 2 for each layer). In the final layer of the decoder the ReLU is replaced by Sigmoid. As a result we obtain a prediction mask $1 \times 1024 \times 1024$ of the targeted shape (see Fig. 3).

At inferring time the same trained network is applied to the ceiling and floor transform respectively (See Fig. 2(b)). The 2D room layout $F2D$ is obtained with a simple polygonal approximation of the ceiling shape contour, while the ratio of heights h_r (and therefore h_c - see Sec. 4.1), is obtained from the ratio between the contours of the two inferred shapes. In particular, being h_r actually a scale factor between the ceiling and floor transform, it is determined by the scale that maximizes the matching points between the two contours (see Fig. 2(b)). We build the final 3D model just extruding $F2D$, using h_r to determine h_c and h_e (see Sec. 4.1).

4.3 Training

To train our network, we adopt a specific loss function based on the binary cross entropy error of the predicted pixel probability in the mask M and in its gradient M', compared to ground truth:

$$-\frac{1}{n}\sum_{p=M}(\hat{p}\log p+(1-\hat{p})\log(1-p))-\frac{1}{n}\sum_{q=M'}(\hat{q}\log q+(1-\hat{q})\log(1-q)) \tag{2}$$

where p is the probability of one pixel in M, $\hat{p}$ is the ground truth of p in M, q is the pixel probability in M', $\hat{q}$ is the ground truth, and n is the number of pixels in M and M' which is the transform resolution. The gradient of binary masks is obtained by a Sobel filter of kernel size 3. Even though the gradient component provides a value only near edges, its presence improves the sharpening of the contour in cases of small boundary surface details. This is very important in our case, since in our approach we extract the contour of the largest detected component without performing any post-processing. It also improves noise filtering in highly textured images (see ablation study in Sec. 5.3).

Working completely in a plane-projected domain clearly simplifies data augmentation, compared to panorama augmentation [26]. In practice, for each training iteration, we augment the input panorama set with random rotations and mirrorings, performing all operations in 2D space.

We could separately perform training of an instance for floor prediction and a second instance for ceiling mask prediction, or create an architecture that performs parallel training with a common loss function, or use a single instance capable to handle both ceiling and floors.

In the first case, we experienced, for the ceiling branch training, a tendency to over-fit and a rapid decay of the learning rate after a small number of iterations. At the same time, training the floor branch with only floor images results in rough shapes. This is a predictable behavior, taking into account that the ceiling part usually has cleaner areas but with less features, while in the floor part the architectural structure is more occluded and therefore more difficult to match, alone, with the ground-truth shape of the room[30].

In the second case, we tested two parallel branches by jointly training two instances of *AtlantaNet*, where the loss function is the sum of the ceiling and floor loss respectively. It should be noted that in this case a direct feature fusion is not possible, since this would imply knowledge of the scale factor between the two transformed tensors, which is itself an unknown value. In this case, we obtained an appreciable improvement of the performance compared to single training. However, the resulting shape is not accurate enough, especially in cases of multiple corners or more complex shapes (see results in Sec.5.3).

We thus adopted a strategy that uses a single *Atlanta Net* instance, but trained to predict indifferently the ceiling or floor shape. To do this, we feed the same network with examples of ceiling and floor transforms, coupled with their respective ground truth. As showed by comparative results (see Sec. 5.3), such a strategy boosts the performances, as it guides the network to find commonalities between clean structures, mostly present in the ceiling transforms, and highly cluttered structures, mostly present in floor transforms.

5 Results

We implemented our method with PyTorch [18], adopting *ResNet50* as feature encoder. The presented results are obtained using the Adam optimizer [14] with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and learning rate 0.0001. We trained the network on 4 NVIDIA RTX 2080Ti GPUs for 300 epochs (best valid around 200 epoch, varying with dataset), with a batch size of 8 ($3 \times 1024 \times 1024$ input size). As an example, training with the *MatterportLayout* [34] dataset takes about 2 minutes per epoch. The final layout extraction is obtained by applying a simple polygonal approximation [5] to the larger connected region contour (see Sec. 4.2), thus eliminating excess vertices and saving the resulting model as a json file (we adopt the same convention as *MatterportLayout* [34] and *PanoAnnotator* [29]).

5.1 Datasets

We trained *AtlantaNet* using publicly available datasets: *PanoContext* [32], *Stanford 2D-3D* [25] and *Matterport3D* [17]. To simplify comparison, we arrange testing by following the split (*cuboid layout*, or general Manhattan World), adopted by other works [33,6,26,30]. In addiction, we introduce a specific testing set of a hundred images to benchmark more complex Atlanta World cases (*AtlantaLayout*). The testing set was created by annotating a selection of images from *Matterport3D* [17] and *Structured3D* [15]. For cuboid and simple Manhattan layout, we follow the same training/validation/test splitting proposed by LayoutNet [33] and HorizonNet [26], while for general Manhattan World we follow the data split and annotation provided by Zou et al. [34] (e.g., *MatterportLayout*).

To test Atlanta World layouts, we extend existing testing set with annotated 3D layouts having less restrictive assumptions, as, for example, rooms with curved walls or non-right corner angles. In this case, to ensure a fair evaluation we have prepared the test set by combining the new annotations with a subset of test images taken from the *MatterportLayout* testing set.

5.2 Performance

We evaluate the performance of our approach by following the standard evaluation metrics proposed by Zou et al. [34] and adopted by others [33,26,30]. Specifically, we considered the following metrics: $3DIoU$ (volumetric intersection-over-union), $2DIoU$ (pixel-wise intersection-over-union), *cornererror* (L2 distance normalized to bounding box diagonal), *pixelerror* (floor, ceiling, wall labeling accuracy of the original image) and δ_i (percentage of pixels where the ratio between the prediction label and the ground truth label is within a threshold of 1.25). Following Zou et al. [34], we adopt $3DIoU$, *cornererror* and *pixelerror* for cuboid layouts, and $3DIoU$, $2DIoU$, δ_i for other layouts.

We present a comparison with recent state-of-the-art methods [33,6,26,30] for which comparable results are published or for which source code and data are available. For comparison purposes, we adhere to the methodology reported in the mentioned papers, and we split results into *Cuboid layouts*, *General Manhattan*

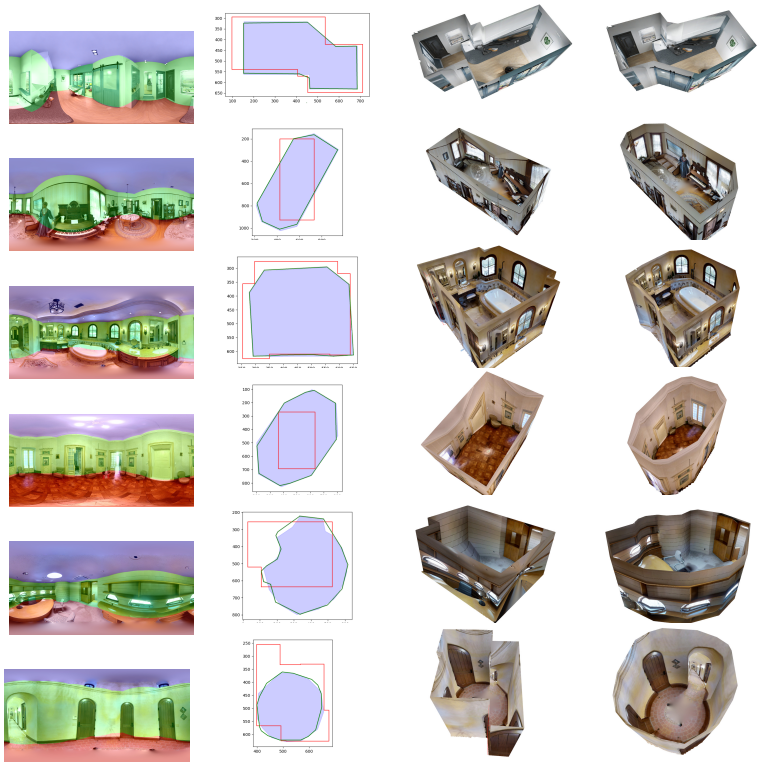


Fig. 4. Qualitative results and comparison. For each row, we show: the original panoramic image annotated with our reconstruction; an intersection-over-union visual comparison, between our approach (green line), HorizonNet [26] (red line) and ground truth (azure mask); the 3D layout obtained with the compared approach [26] (third column) and with ours (fourth column).

World and *Atlanta World*, preserving the same metrics and setup of the original papers. All results are collected with the same $ResNet50$ feature encoder. Missing fields in tables indicate cases not reported in original papers.

Tab. 1 reports on performance obtained on *Cuboid layouts*, a worst-case comparison for our method, since, in contrast to competitors, we do not assume that walls must meet at right angles. Following the same convention presented by Sun et al. [26] and Zou et al. [34], the networks have been trained with three different datasets (i.e., PanoContext, Stanford 2D-3D-S, both of them), and tested with same testing set - e.g. Stanford 2D-3D-S [25]. Results demonstrate how our approach, on these constrained indoors, has a performance similar to state-of-the-art approaches tuned for Manhattan-world environments, although it does not employ any specific post-processing and cuboid regularization.

In Tab. 2, we report on performance obtained on *General Manhattan World* and *Atlanta World* layouts (see Sec. 5.1). We compare our method results with results for methods having best performance in general Manhattan cases [30,26]. All the tested approaches are trained with the same *MatterportLayout* dataset [34]

Training dataset:	PanoContext			S-2D-3D			PC+Stanford		
Metrics [%]:	3D IoU	Corner error	Pixel error	3D IoU	Corner error	Pixel error	3D IoU	Corner error	Pixel error
CFL [6]	65.13	1.44	4.75	-	-	-	-	-	-
LayoutNet [33]	-	-	-	76.33	1.04	2.70	82.66	0.83	2.59
Dula-Net [30]	-	-	-	79.36	-	-	**86.60**	**0.67**	2.48
HorizonNet [26]	**75.57**	**0.94**	3.18	79.79	0.71	2.39	82.66	0.69	2.27
Ours	75.56	0.96	**3.05**	**82.43**	**0.70**	**2.25**	83.94	0.71	**2.18**

Table 1. Cuboid layout performance. All the methods have been tested with the same *S-2D-3D* testing set [25] and trained with the enlisted training sets. Our method, even without Manhattan World pre-processing and regularization, is aligned with the performance of the best state-of-art methods that exploit Manhattan-world constraints.

	Dula-Net [30]			HorizonNet [26]			Ours		
	3D IoU	2D IoU	δ_i	3D IoU	2D IoU	δ_i	3D IoU	2D IoU	δ_i
Manhattan 4 corners	77.02	81.12	0.818	81.88	84.67	0.945	**82.64**	**85.12**	**0.950**
Manhattan 6 corners	78.79	82.69	0.859	**82.26**	**84.82**	**0.938**	80.10	82.00	0.815
Manhattan 8 corners	71.03	74.00	0.823	71.78	73.91	0.903	**71.79**	**74.15**	**0.911**
Manhattan >10 corners	63.27	66.12	0.741	68.32	70.58	0.861	**73.89**	**76.93**	**0.915**
Manhattan Overall	75.05	78.82	0.818	79.11	81.71	0.929	**81.59**	**84.00**	**0.945**
Atlanta 6 corners	-	-	-	74.45	77.13	0.862	**84.26**	**88.78**	**0.972**
Atlanta 8 corners	-	-	-	65.00	66.93	0.820	**78.37**	**80.50**	**0.907**
Atlanta >10 corners-odd	-	-	-	64.40	67.72	0.812	**75.34**	**77.75**	**0.870**
Atlanta Overall	-	-	-	67.08	70.57	0.845	**72.50**	**76.49**	**0.879**
Atlanta FT Overall	-	-	-	73.53	76.38	0.851	**80.01**	**84.33**	**0.924**

Table 2. General layout performance. All methods are trained with the same *MatterportLayout* dataset [34] and tested on the *MatterportLayout* test set and on a specific set of complex Manhattan and non-Manhattan scenes (e.g., AtlantaLayout). For Dula-Net [30] performance we refer to the latest available results using *MatterportLayout* training [34]. *>10 - corners-odd* row refer to complex layouts, including curved walls.

and evaluated both on the *MatterportLayout* testing set (labeled Manhattan in tab. 2) and on a specific testing set (labeled Atlanta in tab. 2), containing more complex shapes, such as non-right angles and curved walls. For Dula-Net [30] performance, we refer to the latest available results obtained by training with the *MatterportLayout* dataset by Zou et al. [34]. The *Atlanta FT Overall* line presents, in addition, results that have been obtained by augmenting the *MatterportLayout* training dataset with selected Atlanta scenes for fine-tuning. The results demonstrate the accuracy of our approach with both testing sets, and how it outperforms other approaches as the layout complexity grows. It should be noted that a portion of the error depends, for all the approaches, by the approximated ground truth annotation, which clearly affects both training and performance evaluation.

In Fig. 4, we show a selection of scenes for a qualitative evaluation of our method compared to ground truth and *HorizonNet* [26]. At the first column we show the original panoramic image annotated with our results. It should be noted how, in these complex cases, even the manual labeling of an equirectangular image is not trivial, as well as the visual understanding of the room structure. In order to provide a more intuitive comparison, we show, besides, the intersection-over-union of the recovered layout (green) with the ground truth floorplan (azure mask) and the same layout reconstructed by HorizonNet [26] (red). In the third and fourth

column, we show the 3D layout obtained, respectively, with our approach and with the *HorizonNet* approach [26]. Visual results confirm numerical performances in terms of footprint and height recovery.

5.3 Ablation Study

Backbone	Setup	Gradient loss	3D IoU	2D IoU	δ_i	Train. params
ResNet50	Two instances trained separately		75.48	78.26	0.856	200M
ResNet50	Two instances trained jointly		76.04	79.92	0.815	200M
ResNet50	One instance and mixed feeding		79.26	83.35	0.854	100M
ResNet50	One instance and mixed feeding	V	80.79	84.12	0.902	100M
Resnet101	One instance and mixed feeding	V	83.22	86.96	0.940	119M

Table 3. Ablation. The ablation study demonstrates how our proposed designs improve the accuracy of prediction. Results are sorted by increasing performance, showing only those cases that actually increase it.

Our ablation experiments are presented in Tab. 3. We report the results averaged across general Manhattan and Atlanta World testing instances (Tab. 2). First, we tested, with the same *ResNet50* backbone and without gradient loss function (Sec. 4.3), different configurations: two instances trained separately, two instances trained jointly with a common (overall) loss function and the adopted mixed approach. While the difference between separate and joined training of two instances is quite small, results confirm instead that the mixed feeding approach (see Sec. 4.3) provides a consistent performance boost. For the winning set-up (One instance and mixed feeding), we also evaluate the contribution of the gradient loss component. Including the gradient leads to an accuracy improvement, mainly due to increased performance with more complex shapes.

At last, we show how our method changes its performance by adopting a deeper backbone - i.e., *ResNet101*. While the *ResNet50* encoder (also adopted by compared works) provides consistent results for the given datasets (see Sec. 5.1), increasing the backbone depth appears to be a better option for more complex layouts.

5.4 Limitations and Failure Cases

(a) (b) (c)

Fig. 5. Failure case. Fig. 5(a) shows a circular room where the ceiling level is not correctly identified, resulting in the wrong layout of Fig. 5(b) and Fig. 5(c) (ground truth as green line).

Our method is trained to return a single connected region for each projection (ceiling and floor), containing the information needed to recover the room layout

(see Sec. 4.2). Fig. 5 shows an example where the layout of a semi-circular room (Fig. 5(a)) is wrongly predicted. Although geometrically self-consistent (see recovered 3D at Fig. 5(b)), the recovered shape (yellow ceiling mask in Fig. 5(c)) does not describe the real room layout (green annotation). From the topological point-of-view, this happens where the horizontal planes are not clearly identifiable, so, in our example, when the horizontal ceiling is partially occluded by other horizontal structures.

6 Conclusions

We have introduced a novel end-to-end approach to predict the 3D room layout from a single panoramic image. We project the original panoramic image on two horizontal planes, one above and one below the camera, and use a suitably trained deep neural network to recover the inside-outside segmentation mask of these two images. The upper image mask, which contains less clutter, is used to determine the 2D floor plan in form of a polygonal layout, while the correlation between upper and lower mask is used to determine the room height under the Atlanta world model. Our experimental results clearly demonstrate that our method outperforms state-of-the-art solutions in prediction accuracy, in particular in cases of complex wall layouts or curved wall footprints. Moreover, the method requires much less pre- and post-processing than competing solutions based on the more constraining Manhattan world model.

Our current work is concentrating in several directions. In particular, we are planning to exploit multiple images to perform a multi-view recovery of rooms with large amount of clutter or complex convex shapes. Moreover, we are also working on the integration of this approach in a multi-room structured reconstruction environment, in order to automatically reconstruct complete building floors.

Acknowledgments. This work has received funding from Sardinian Regional Authorities under projects VIGECLAB, AMAC, and TDM (POR FESR 2014-2020). We also acknowledge the contribution of the European Union's H2020 research and innovation programme under grant agreements 813170 (EVOCATION).

References

1. Acuna, D., Ling, H., Kar, A., Fidler, S.: Efficient interactive annotation of segmentation datasets with polygon-rnn++. In: Proc. CVPR (2018)
2. Castrejon, L., Kundu, K., Urtasun, R., Fidler, S.: Annotating object instances with a polygon-rnn. In: Proc. CVPR (2017)
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE TPAMI **40**(4), 834–848 (2017)
4. Delage, E., Honglak Lee, Ng, A.Y.: A dynamic Bayesian network model for autonomous 3D reconstruction from a single indoor image. In: Proc. CVPR. vol. 2, pp. 2418–2428 (2006)
5. Douglas, D.H., Peucker, T.K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica: The International Journal for Geographic Information and Geovisualization **10**(2), 112–122 (1973)
6. Fernandez-Labrador, C., Fácil, J.M., Perez-Yus, A., Demonceaux, C., Civera, J., Guerrero, J.J.: Corners for layout: End-to-end layout recovery from 360 images. arXiv:1903.08094 (2019)
7. Flint, A., Murray, D., Reid, I.: Manhattan scene understanding using monocular, stereo, and 3D features. In: Proc. ICCV. pp. 2228–2235 (2011)
8. Flint, A., Mei, C., Murray, D., Reid, I.: A dynamic programming approach to reconstructing building interiors. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) Proc ECCV. pp. 394–407 (2010)
9. Gallagher, A.C.: Using vanishing points to correct camera rotation in images. In: Proc. CVR. pp. 460–467 (2005)
10. Geyer, C., Daniilidis, K.: A unifying theory for central panoramic systems and practical implications. In: Proc. ECCV. pp. 445–461 (2000)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. CVPR. pp. 770–778 (2016)
12. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: Proc. ICCV. pp. 1849–1856 (2009)
13. Hoiem, D., Efros, A.A., Hebert, M.: Recovering surface layout from an image. International Journal of Computer Vision **75**(1), 151–172 (Oct 2007)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2014)
15. Kujiale.com: Structured3D Data. `https://structured3d-dataset.org/` (2019), [Accessed: 2019-09-25]
16. Lee, D.C., Hebert, M., Kanade, T.: Geometric reasoning for single image structure recovery. In: Proc. CVPR. pp. 2136–2143 (2009)
17. Matterport: Matterport3D. `https://github.com/niessner/Matterport` (2017), [Accessed: 2019-09-25]
18. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. In: Proc. NIPS (2017)
19. Pintore, G., Garro, V., Ganovelli, F., Agus, M., Gobbetti, E.: Omnidirectional image capture on mobile devices for fast automatic generation of 2.5D indoor maps. In: Proc. IEEE WACV. pp. 1–9 (2016)
20. Pintore, G., Mura, C., Ganovelli, F., Fuentes-Perez, L., Pajarola, R., Gobbetti, E.: State-of-the-art in automatic 3d reconstruction of structured indoor environments. Comput. Graph. Forum **39**(2), 667–699 (2020)

21. Pintore, G., Pintus, R., Ganovelli, F., Scopigno, R., Gobbetti, E.: Recovering 3D existing-conditions of indoor structures from spherical images. Computers & Graphics **77**, 16–29 (2018)
22. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241 (2015)
23. Schindler, G., Dellaert, F.: Atlanta world: an expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In: Proc. CVPR. vol. 1, pp. I–I (2004)
24. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.k., Woo, W.c.: Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: Proc. NIPS. p. 802–810 (2015)
25. Stanford University: BuildingParser Dataset. `http://buildingparser.stanford.edu/dataset.html` (2017), [Accessed: 2019-09-25]
26. Sun, C., Hsiao, C.W., Sun, M., Chen, H.T.: HorizonNet: Learning room layout with 1D representation and pano stretch data augmentation. In: Proc. CVPR (June 2019)
27. Xu, J., Stenger, B., Kerola, T., Tung, T.: Pano2CAD: Room layout from a single panorama image. In: Proc. WACV. pp. 354–362 (2017)
28. Yang, H., Zhang, H.: Efficient 3D room shape recovery from a single panorama. In: Proc. CVPR. pp. 5422–5430 (2016)
29. Yang, S.T., Peng, C.H., Wonka, P., Chu, H.K.: PanoAnnotator: A semi-automatic tool for indoor panorama layout annotation. In: Proc. SIGGRAPH Asia 2018 Posters. pp. 34:1–34:2 (2018)
30. Yang, S.T., Wang, F.E., Peng, C.H., Wonka, P., Sun, M., Chu, H.K.: DuLa-Net: A dual-projection network for estimating room layouts from a single RGB panorama. In: Proc. CVPR (2019)
31. Yang, Y., Jin, S., Liu, R., , Yu, J.: Automatic 3D indoor scene modeling from single panorama. In: Proc. CVPR. pp. 3926–3934 (2018)
32. Zhang, Y., Song, S., Tan, P., Xiao, J.: PanoContext: A whole-room 3D context model for panoramic scene understanding. In: Proc. ECCV. pp. 668–686 (2014)
33. Zou, C., Colburn, A., Shan, Q., Hoiem, D.: LayoutNet: Reconstructing the 3D room layout from a single RGB image. In: Proc. CVPR. pp. 2051–2059 (2018)
34. Zou, C., Su, J.W., Peng, C.H., Colburn, A., Shan, Q., Wonka, P., Chu, H.K., Hoiem, D.: 3d manhattan room layout reconstruction from a single 360 image (2019)