3D Room Layout Estimation from a Single RGB Image

Chenggang Yan[1,2], Biyao Shao[1], Hao Zhao[3], Ruixin Ning[1], Yongdong Zhang[4] *Senior Member, IEEE* and Feng Xu[3]

[1]Hangzhou Dianzi University, China
[2]Shandong University, China
[3]Tsinghua University, China
[4]University of Science and Technology of China

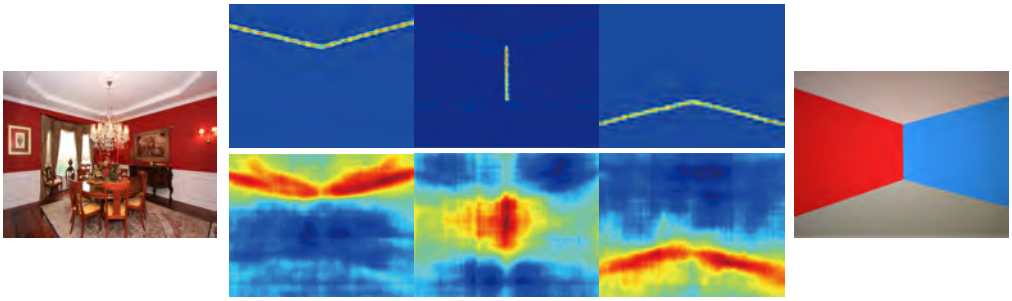

Fig. 1: Our method takes a single image of an indoor scene (left) as input and estimates the corresponding 3D topology layout (right) automatically. The three columns in the middle indicate the result of 2D room layout, which consists of three semantic edges. The lower row shows the result of the baseline method, while the upper row is the result of our method. The 3D topology layout is reconstructed based on the 2D room layout.

***Abstract*—3D layout is crucial for scene understanding and reconstruction, and very useful in applications like real estate and furniture design. In this paper, we propose a fully automatic solution to estimate 3D layout of an indoor scene from a single 2D image. Our technique contains two key components. Firstly, we train a neural network that directly estimates room structure lines from the input image. Secondly, we propose a novel technique to automatically identify the layout topology of an input image, followed by a nonlinear optimization with equality constraints to estimate the final 3D layout of a scene. Based on our knowledge, this is the first fully automatic technique to achieve single image-based 3D layout estimation of an indoor scene. We evaluate our method on the public datasets $LSUN$, $Hedau$ and $3DGP$ and the results show that the proposed method achieves accurate 3D layout reconstruction on various images with different layout topologies.**

***Index Terms*—indoor scene, 2D topology , 3D topology , convolutional neural networks , nonlinear optimization.**

Chenggang Yan was with the Department of Automation, Hangzhou Dianzi University, Hangzhou, 310018, China and the School of Mechanical,Electrical & Information Engineering, Shandong University, Weihai, 264209, China.
E-mail: cgyan@hdu.edu.cn.

Biyao Shao was with the Department of Automation, Hangzhou Dianzi University, Hangzhou, 310018, China.
E-mail: 161060024@hdu.edu.cn.

Hao Zhao was with the Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China.
E-mail: hao.zhao@intel.com.

Ruixin Ning was with the Department of Automation, Hangzhou Dianzi University, Hangzhou, 310018, China.
E-mail: ningruixin_work@163.com.

Yongdong Zhang was with School of Information Science and Technology, University of Science and Technology of China, Anhui, 230026, China.
E-mail: zhyd73@ustc.edu.cn.

Feng Xu was with the BNRist and School of Software, Tsinghua University, Beijing, 100084, China. Feng Xu is the corresponding author. E-mail: feng-xu@tsinghua.edu.cn.

I. Introduction

3D layout estimation aims to reconstruct the intrinsic 3D structure of a room (the geometry of the floor, ceiling and walls) which is full of various objects and furniture. 3D layout is very useful in applications of virtual reality [1], [2] and design [3]. Furthermore, It could contribute to many high-level scene understanding tasks like object segmentation [4], pose estimation and recognition [5], [6].

However, 3D layout estimation from a single 2D image is an ill-posed and challenging problem. The difficulty lies in two aspects. First, the walls, floor and ceiling are usually occluded by objects in the scene. In many cases, the wall-floor, wall-wall, and wall-ceiling edges (as shown in Figure 2, also called room structure lines), which are the most important cues to identify the room structure, are

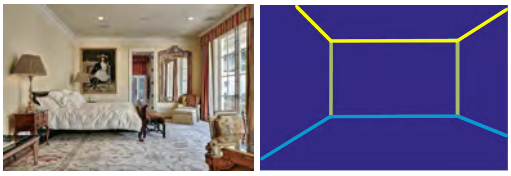

Fig. 2: The task of 2D layout estimation. The left is an input RGB image. The right is the output 2D layout, which is consisted of three semantic lines: the wall-floor (e_{wf}), wall-wall (e_{ww}), and wall-ceiling (e_{wc}) lines.

partially occluded or even fully occluded in images. Second, images usually record just a part of the room, and thus the topologies of a room in different images may be totally different. For example, one image may contain the ceiling, the floor and three walls, while another image may only contain the floor and two walls. Traditionally, we need to define different possible room layout topologies in advance [7] for optimization.

In this paper, we propose a fully automatic solution and overcome the aforementioned difficulties. Our method first estimates the room structure lines from a 2D image, and then reconstructs the 3D layout from the 2D edges. In the edges detection step (also called layout estimation as the edges indicate the 2D layout of the room), we propose a novel end-to-end network which is trained to identify the edges very accurately no matter they are occluded or not. In the 3D layout estimation step, we propose a novel method called *topology anchor point optimization* ($TAPO$) to automatically determine the topology of a scene by identifying the 2D locations of our defined topology anchor points. Then after a nonlinear optimization, the final 3D layout could be reconstructed.

In summary, we mainly have three contributions.

1) An end to end system which automatically estimates 3D layout of a room from a 2D image;
2) A novel CNN-based scheme that directly regresses the room structure lines from a single stage. It largely improves the regression quality compared to the current state-of-the-art techniques;
3) The $TAPO$ method which directly identifies the layout topology of a scene from the detected edge pixels without requiring any user interactions.

II. Related Works

Spatial layout is an important source of information of an indoor scene [8], which provides a strong prior for scene understanding [9]–[11] and objects recognition [12]–[14]. For example, floors usually serve as the supporting surfaces for objects, such as chairs and tables. Many objects, like paintings, are usually aligned with walls. As a consequence, layout estimation has been widely investigated by many previous works, and can be categorized as geometry-based methods and learning-based methods.

A. Geometry-based methods

2D layout estimation. Hoiem et al. [15] propose a method to learn appearance-based models of geometric classes in the scene, which coarsely describes the 3D scene orientation of each image region. Hedau et al. [16] derive geometric context labels for an indoor scene, which assigns to each pixel a class set. This method first estimates three orthogonal vanishing points by clustering line segments in the scene according to the Manhattan assumption [17]. Schwing et al. [18] further demonstrate the decomposition of higher-order potentials to improve the efficiency of this kind of algorithms. Del Pero et al. [19] propose a model that predicts the geometry and objects presented in a room using Bayesian inference with 3D reasoning. The authors emphasize the importance of detecting cues such as faint wall edges to improve the layout estimation of a room. In [20], Wang et al. use latent variables. This method eliminates the need for labeled clutter while modeling cluttered scenes.

3D layout estimation. Given only one RGB image, depth information is inherently ambiguous. There are many algorithms that use multiple images to recover depth information. Some researchers use structure from motion [21] to estimate depth, or use shape from defocus [22]. On the other hand, some researchers have attempted to recover 3D information from a single image. Shape from shading [23] is one well-known approach, but is not applicable to clustered scenes. Given sufficient human labeling/human-specified constraints, methods based on "3D metrology" [24]–[27] can be used to generate a 3D reconstruction of clustered scenes. However, these methods are not automatic and require a lot of human input, and are not widely used in reality.

Bayesian methods combine visual cues with some prior knowledge of a scene and are efficient to recover 3D information. For instance, Kosaka and Kak [28] propose a navigation algorithm that allows a monocular robot to track its position by visual cues, such as lines and corners in a building where a floor plan is available. Han and Zhu [29] propose a more flexible algorithm that uses models both of man-made objects and natural objects. Erick Delage et al. [30] use Markov random field model to identify different planes and edges in a scene and apply an iterative algorithm to obtain 3D reconstruction. Notice that all these methods focus on the visible scene reconstruction, while we aim to reconstruct the complete 3D layout that is blocked by objects only using 2D topology anchor points. Del Pero et al. [31] develop a comprehensive Bayesian generative model for understanding indoor scenes. They find that modeling detailed geometry improves recognition and reconstruction, and enables more use of appearance for scene understanding. Zhang et al. [32] advocate the use of 360 degree full-view panoramas in scene understanding, and propose a whole-room context model in 3D. It is the first to extend the frameworks designed for perspective images to panoramas. They recover both the layout, assumed as a 3D box (4 walls), and bounding boxes of the main objects inside the room.

And it has been used to develop algorithms for many fields, such as estimating camera calibration parameters [33] and

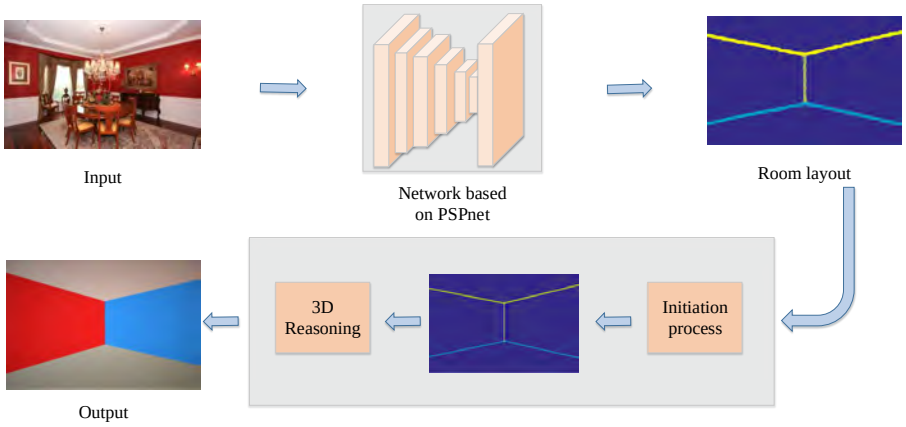

Fig. 3: The main process of our method. The input is a RGB image of an indoor scene. The output is the 3D layout of the indoor scene. The first part of our method is the 2D layout estimation step (shown in the top row), which basically is a semantic edge detection. And the network here is based on the PSPnet with some simplification in the architecture. As the edges are the room structure lines, estimating the edges equals to achieving 2D room layout estimation. The second part is 3D layout estimation from the 2D layout (shown in the bottom row).

getting camera poses from complex images [17], [34], [35].

B. Learning-based methods

2D layout estimation. Mallya and Lazebnik [36] use a fully convolutional neural network (FCN) for room layout estimation. They present two methods for the prediction of informative edge maps and achieve state-of-the-art results on indoor scene layout prediction by using just edge-based features. Dasgupta et al. [37] propose a method that uses FCN to generate layout estimations. The output of their neural network is per-pixel semantic labels for planes (ground, walls, ceiling). Their method is robust when handling a wide range of highly challenging scenes. However, there is an inherent ambiguity in semantically labeling layouts when the scene has two walls. There are some differences between [36] and [37]. While Dasgupta et al. use FCN for directly predicting per-pixel semantic labels, the method of Mallya and Lazebnik uses it solely for generating "informative edges". These informative edges were then integrated into a more conventional pipeline. Zhang et al. [38] propose a learning-based method to extract edges for room layout estimation. Zhao et al. [7] also propose a method named $semantictransfer$ to detect the edges of indoor scenes. Their method divides the training process into three stages to achieve better performance of room layout estimation. [7] solves the ambiguity in [37]. Chen-Yu Lee et al. [39] do 2D layout estimation by predicting the locations of the room layout keypoints using RoomNet. Their method does not have the ambiguity in [37] and extracts a set of room layout keypoints, and connect the keypoints in a specific order to obtain the layout.

3D layout estimation. In recent years, the field of 3D reconstruction with deep neural network has become more and more active. Given a single image, Christopher B. Choy et al. [40] use a neural network to predict the underlying 3D object as a 3D volume. Zou et al. [41] propose an algorithm to predict 3D room layout from a single image. The algorithm

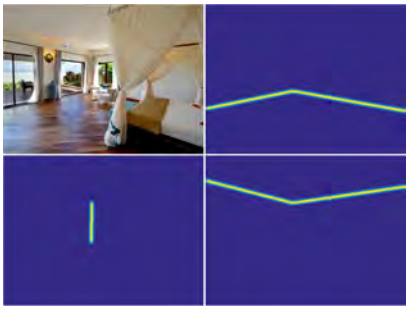

Fig. 4: A sample of the used ground truth for layout estimation. The first row is the original image and the ground truth of wall-floor edges. The second row are the ground truths of the wall-wall edge and wall-ceiling edges. The ground truth used in our method is a new representation derived from the existing ground truth in LSUN.

generalizes across panoramas and perspective images, cuboid layouts and more general layouts.

Datasets. High quality dataset is a key issue for leveraging supervised deep learning methods of layout estimation. Choi et al. [42] collect a new dataset of indoor scene to validate their proposed method. The dataset includes three scene types: living rooms, bedrooms, and dining rooms. Dai et al. [43] provide a dataset of richly-annotated RGB-D scans of real-world environments containing 2.5M RGB-D images in 1513 scans acquired in 707 distinct spaces. Zhang et al. [44] introduce a large scale synthetic dataset that is created from 45K realistic 3D indoor scenes. They explore the effects of rendering methods for three computer vision tasks: surface normal prediction, semantic segmentation, and object boundary detection.

III. Proposed Method

In this paper we propose a novel method to estimate the 3D layout of an indoor scene (ground, ceiling and walls) from

a single 2D image. The method is consisted of two parts as shown in Figure 3: the upper part indicates the 2D layout estimation step, while the lower part shows the 3D layout estimation step. 2D room layout estimation is used as the input of 3D layout reconstruction. We will introduce the two parts in detail in this section.

A. Layout Regression

In this subsection, we propose a new method to achieve 2D room layout estimation. Previous FCN-based methods [7], [37] take room layout estimation as a multi-classification problem essentially, and perform classification on each pixel. However, there are some inherent difficulties for this kind of methods. For example, the training data is unbalanced because background pixels take a large part in all pixels. It will be hard to train a classification model with good performance when the training data is unbalanced.

On the other hand, regression-based methods have shown their power in handling some similar tasks. In the area of human skeleton estimation from images, recent works [45]–[47] use heat maps to regress the location of skeleton points. Compared with a whole image, skeleton points take very little pixels. However, these regression-based methodologies can address the problem even in quite challenging cases. Our goal is to estimate wall-floor edges (e_{wf}), wall-wall edges (e_{ww}) and wall-ceiling edges (e_{wc}). It is similar with the task of [46], [47] essentially. Based on this observation, we propose to model room layout estimation as a regression problem of three kinds of edges. This solves the problem of unbalanced data caused by background pixels.

The training loss is defined as below:

$$loss = \sum_{e \in [1,2,3]} \|O_e - G_e\|_2 \quad (1)$$

G_e means the e-th channel of ground truth. O_e denotes the e-th channel of the output in each iteration. The first channel is the information of e_{wc}. The next two channels are the information of e_{ww} and e_{wf} respectively.

Figure 4 shows an input image and its ground truth wf, ww and wc edge maps. We first apply Gaussian Function on the ground truth edges where the edge pixels are with the value of 255 while other pixel values are 0. In this case, the pixel value turns to stand for the possibility of a pixel being on an edge.And then we have a three-channel map to represent the ground truth, each of which represents one kind of edge and takes the value of [0, 255]. Here, the value stands for the strength of a pixel being on the corresponding edge. Next, we build our network upon the recently proposed architecture PSPNet [48], which achieves the state-of-the-art performance on semantic segmentation. We pre-train a network based on PSPNet using SUNRGBD dataset and get a model for semantic segmentation, which has the ability to segment 37 types of objects. After pre-training the model on SUNRGBD [49], we fine tune it on LSUN [50]. Images in LSUN have a wide range of resolutions. The dimension of input data layer is fixed ($473 \times 473 \times 3 \times 1$) in the training process. The input of each iteration is a randomly cropped patch from an image. It may cause that the input of some iterations don't contain any edges and have no valid information. In order to make the information of the patches used in each iteration as stable as possible, we use bilinear interpolation to resize images and ground truth, and keep the original ratio of every image. The network is trained using stochastic gradient descent with momentum for fifty thousand iterations within the Caffe framework [51]. After training, the network will generate three maps of an input image, representing the possibility of a pixel belonging to the three edges. Notice that our method does not need to divide training process into several stages as the previous work did [7].

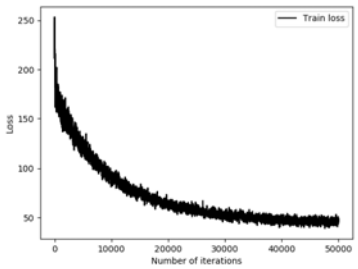

Fig. 5: Loss curve during training iteration. Learning rate is set to 10^{-5}. "Poly" is chosen as the learning rate decay policy. We use a power value of 0.9, a momentum value of 0.9 and weight decay value of 0.0001.

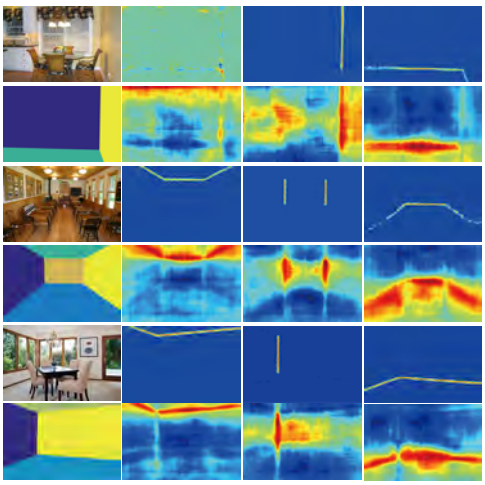

Fig. 6: Comparisons of semantic edge detection between our method and [7]. Every two rows form a pair result. First column of each pair is the original image and its ground truth. The following three columns are the comparisons of results obtained by the two methods. The above is the result of our proposed method. The below is the result obtained by semantic transfer. The results of our proposed method are sharper than that of semantic transfer and have less errors.

Besides, compared with the results of semantic transfer [7], results of our proposed method have made great progress. As shown in Figure 6, The results of our method have little noise and are more precise so that the following optimization step is largely eased.

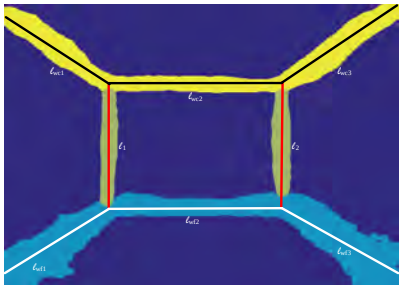

Fig. 7: l_1 and l_2 are wall-wall lines. The yellow lines are the intersection of walls and ceiling. The blue lines are the intersection of walls and floor. Red straight lines divide yellow lines and blue lines into segments. Each segment represents an intersection of two planes. For example, the three yellow segments are the intersections between the ceiling and the left, middle and right walls, respectively. We present a result of the semantic transfer method here because the edges are thick.

B. Topology Anchor Points Optimization

There are different room layout topologies in 2D images. [50] defined 11 room layouts to cover most of the possible situations under typical camera poses.

1) Wall-Wall Line Estimation: Firstly, we will estimate the wall-wall line from the detected wall-wall pixels. T is the output of network, containing three $w \times h$ heat maps. Then, we define a label-map L as a matrix where the values in the matrix represent the categories of each pixel (0: background, 1-3: the three semantic edges), determined by a threshold and the channel of the maximum value in T. The value of the threshold is determined according to the data distribution of T. L is shown in Figure 7 and denoted as follows:

$$L_{a,b} = \begin{cases} f, & max_f T^f_{a,b} \geq threshold \\ & f = 1, 2, 3; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Here, (a, b) indicates the pixel coordinates on the image. In this step, we only consider the wall-wall lines and optimize their locations. If the scene doesn't have wall-wall lines, this step is skipped. To be specific, we fit straight line functions to the rough wall-wall edge pixels in L by least square optimization. For convenience, we define the straight line function to be $l_i = s_i \cdot x + b_i$. here s_i and b_i mean the slope and intercept of the i-th straight line. Based on l_i we can get pixel set $\{F = \{F_{ij}, F_{ij} \in l_i\}\}$. In order to evaluate results quantitatively, we define the following scoring metric:

$$score(l_i|T) = \frac{1}{N(F)} \sum_{i,j} T^2_{F_{i,j}} \tag{3}$$

Here superscript 2 means the second channel of T, which indicates the wall-wall edge. $N(F)$ means the number of pixel points in the set. F_{ij} is the jth point in l_i. Then we adjust the position of the straight lines until the $score$ doesn't increase any more. This fine tune step can make the estimated line more accurate as L lost the detailed information in T. L helps to estimate a rough line equation efficiently, but using T in a fine tuning step makes the result more accurate.

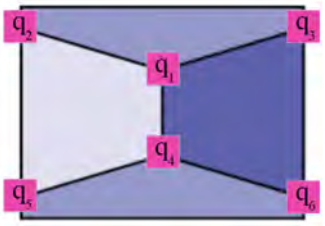

Fig. 8: One kind of topology, clipped from LSUN.

2) Line Segments Extraction: We've got reasonable wall-wall line functions l_i in the previous section. As shown in Figure 7, l_i can be used to divide pixels of wall-floor and wall-ceiling edges into segments. And each segment is the intersection of an unique pair of planes in the scene. The reason that we first estimate the wall-wall lines is just because it helps to perform the segmentation. On one hand, for either wall-floor or wall-ceiling edges, pixels on different lines may be connected, and it is not easy to distinguish them clearly without additional information. On the other hand, pixels on different wall-wall lines are not connected and thus can be easily grouped to estimate all wall-wall lines in the scene. Further more, the estimated wall-wall lines can be used to perform the segmentation.

3) All Line Optimization: Based on the segmentation, we fit straight line functions to every segments using least square optimization. For convenience, we take Figure 7 as an example and denote wall-floor, wall-ceiling edges as: l_{wc1}, l_{wc2}, l_{wc3}, l_{wf1},l_{wf2}, l_{wf3}. We first extract the intersections of two straight lines, and the intersections of the line and the image boundaries. If the intersection of l_{wc1} and l_1 is not coincident with that of l_{wc2} and l_1, we take their average. After this operation, we get the preliminary topology anchor points and the lines connecting the anchor points.

Let the topology points be $\{q = [q_k, c_k], k \in [1, num]\}$, with num as the number of topology anchor points. q_k denotes the point itself and c_k contains the information about the lines ended at q_k. The more edges q_k connects, the more channels c_k has. As shown in Figure 8, c_2 is [1], meaning that q_2 is only related to line 1. Accordingly, c_1 is [1, 1, 2]. Based on these information, the topology of the scene is determined basically. The next step is to further optimize the 2D positions of the topology anchor points.

For this optimization, we define a consistency objective for each topology anchor point.

$$CO = \sum_{k \in [1,num]} \sum_{f \in c_k} T^f(q_{kx}, q_{ky}) \tag{4}$$

Here T denotes the output of the network. f denotes the f-th channel in T. (q_{kx}, q_{ky}) denotes the coordinates of anchor point q_k. Our goal is to maximize CO by adjusting the position of q_k. To be specific, we calculate the gradient of a certain point and update the anchor point position accordingly until CO doesn't increase.

$$\frac{\partial CO}{\partial q_{kx}} \approx CO(q_{k(x+\Delta x)}) - CO(q_{k(x-\Delta x)}) \tag{5}$$

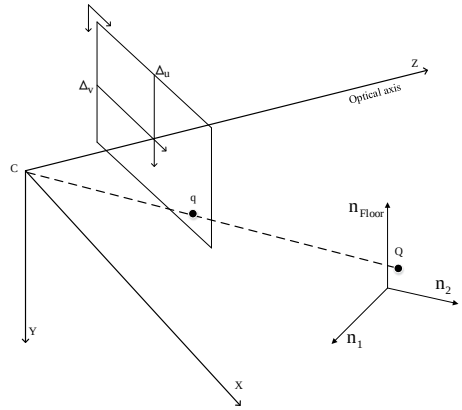

Fig. 9: Camera projection coordinate system used in our method. Q is a point in 3D world.

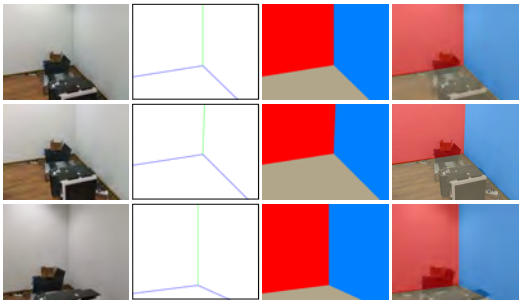

Fig. 10: Shooting the same scene from different direction to prove that the camera's optical axis direction doesn't need to be parallel to the floor plane strictly. In the three rows the camera in facing the horizontal direction, a little bit up and a little down. The first column shows the input images. The second column shows the 2D layouts of the scene. The third column shows the 3D layouts of the scene. The final column shows the blend results of the original image and the 3D layout.

$$\frac{\partial CO}{\partial q_{ky}} \approx CO(q_{k(y+\Delta y)}) - CO(q_{k(y-\Delta y)}) \tag{6}$$

$$\Delta q_k = \alpha \times (\frac{\partial CO}{\partial q_{kx}}, \frac{\partial CO}{\partial q_{ky}}) \tag{7}$$

α is a scaling factor. If an anchor point is in boundary, an additional constraint is imposed by setting corresponding component of Δq_k to zero. The operations mentioned in Sec III-B can be done within one second.

C. 3D Reasoning

We follow the algorithm in [30] to solve the 3D plane estimation problem and make some necessary assumptions:

Algorithm 1: Topology Anchor Points Optimization

Input: preliminary topology anchor points and T
Output: optimized topology anchor points
while CO *increases* **do**
  for $k = 1; k \leq num; k + 1$ **do**
    update q_k according to Equation 5 6;
  end
  calculate CO with updated topology anchor points;
end

1. The image is obtained by perspective projection. As shown in Figure 9, a 3D coordinate Q is projected to a pixel coordinate q.

$$Q = \lambda K^{-1} q \tag{8}$$

Details of K, q and Q are as follows: $K = \begin{bmatrix} f & 0 & \Delta_u \\ 0 & f & \Delta_v \\ 0 & 0 & 1 \end{bmatrix}, q = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, Q = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$. Here, λ denotes the depth of the scene point.
2. Each scene satisfies the Manhattan world assumption [17], [52]: adjacent planes are perpendicular to each other.
3. The camera's optical axis is parallel to the floor plane. Further explanation can be found in Figure 10.

In the camera coordinate system, a 3D plane can be determined uniquely using normal vector n_p and the distance from the plane to the camera center d_p. Index p denotes plane p. Equation 9 describes the relationship of these variables.

$$n_p \cdot Q_k = \lambda_k \cdot n_p \cdot (K^{-1} q_k) = d_p \tag{9}$$

q_k is an anchor point we obtained previously. If q_k belongs to one planes p which has a normal vector n_p and a distance d_p from the camera center. Since q_k is an edge point which at least lies on two planes, we can get two λ using the information of the two planes. The values of the two λ should be the same exactly in the ideal situation. Therefore, we can solve the unknowns by minimizing the difference between two different λs as small as possible.

$$E = \sum_{k \in num} \sum_{p,p' \in B} \Delta_{k,p,p'} = \sum_{k \in num} \sum_{p,p' \in B} \|\lambda_{k,p} - \lambda_{k,p'}\|_2 \tag{10}$$

E is our energy function. B is the set of pairs (p, p') of planes that share a common anchor point q_k in the scene. $\Delta_{k,p,p'}$ denotes the depth difference of point q_k, which is calculated using the information of plane p and plane p'. Besides the constraints based on Equation 9, we further add the following constraints based on our assumptions:

1. The normals of adjacent planes are orthogonal.
2. The normal vectors are the unit vectors.
3. The distance of the plane from the camera center should be greater than a threshold, saying 10^{-5}. Without this constraint, planes with 0 distance to the camera center will be the optimal solution, but do not make any sense.

Now we construct a nonlinear programming problem. We solve this problem with active set method. We add a set of

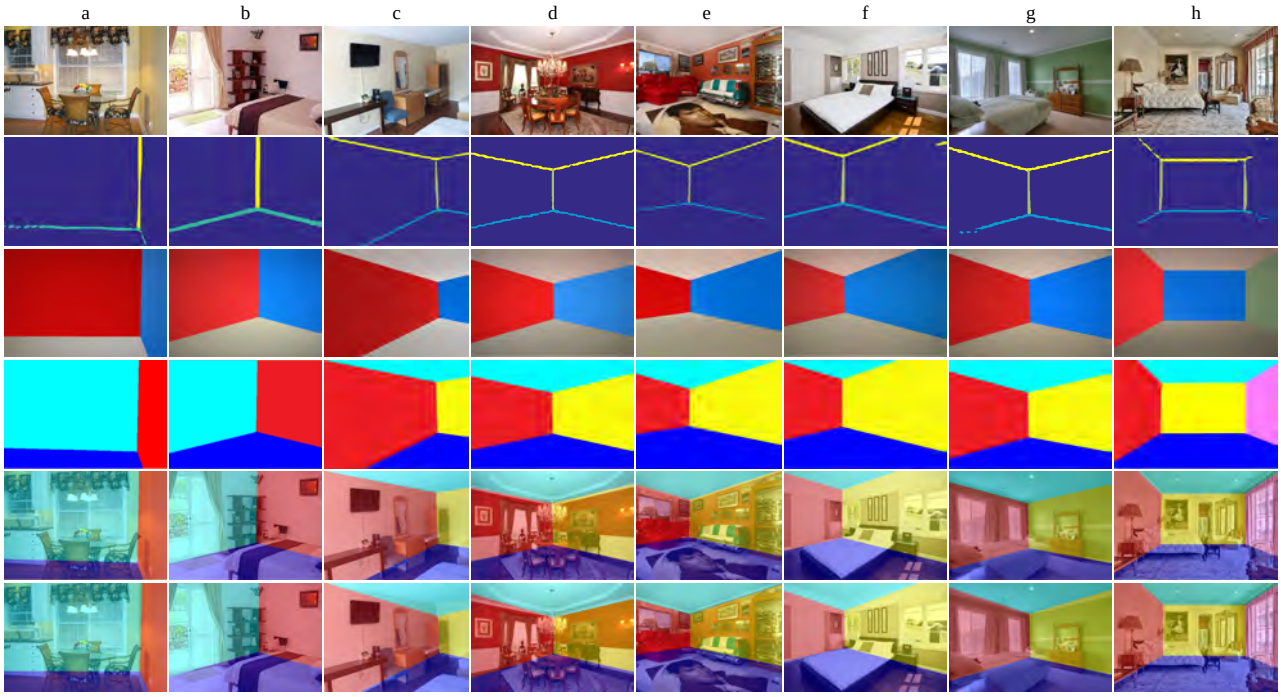

Fig. 11: Qualitative results on LSUN. The first row is original images; the second row is the outputs of neural network; the third row is our 3d rendering results (see supplementary videos for a clearer 3D exposition); the forth row is the 2D plane ground truth; the fifth row is the blend results of the original image and the ground truth. The sixth row is the blend results of the original image and the 3D layout that we reconstruct. In order to make the results more obvious, we give different colors for planes.

constraints onto the energy function. The constraints define a feasible region. x is a solution in the feasible region. Denoting the constraints as $\{g_1(x) \geq 0, \ldots, g_t(x) \geq 0, \ldots, g_m(x) \geq 0\}$ simply, it is called active at x if $g_t(x) = 0$ and inactive at x if $g_t(x) > 0$. There is always an active set in the feasible region. The goal of optimization is to find an x in the active set to make the energy function minimized. The output is the normal vector n_p and the distance d_p from the camera center to each plane, and λ corresponding to the depths of pixels. The solver needs initial values. And we set the values of n_p and d_p randomly to show the robustness of the optimization. The process estimating 3D layout from a 2D image consumes 3 to 6 seconds totally. To visualize the final output 3D layout of the input image, We render the 3D planes in 3D as shown in the third row of Figure 11.

IV. Experiments

In this section, we first show quantitative and qualitative results on the $LSUN$, $Hedau$ and $3DGP$ datasets to demonstrate the power of our technique. Then we compare our method with the state-of-the-art technique PIO on 2D layout estimation. Notice that we achieve similar quality as PIO but do not require specifying any topologies in advance. Finally, we discuss the limitations of our technique.

A. LSUN Results

LSUN is a diverse collection of indoor scenes with human-annotated room layouts for large-scale training and evaluation. It consists of 4000 training, 394 validation, and 1000 held-out testing samples. We validate the effectiveness of the proposed method with the validation set. While the resolution of some images is too high for a single 12GB modern GPU to process, we rescale them and keep the aspect ratio. The output of our method is the normal vector n_ps of the 3D planes in the scene and their distance d_ps to the origin which is the location of the camera center of a recorded image. For a better exposition, we render the planes by OpenGL (as illustrated in the third row of Figure 11 and supplementary videos). We also blend the rendered results with the original images for observing the consistency (as shown in the sixth row of Figure 11. We also quantitatively evaluate our results with pixel errors, which measure the ratio of mislabeled pixels to all pixels in our rendered results.

The inputs include common indoor scenes, such as bedrooms, meeting rooms and living rooms. Figure 11 shows a collection of scenes where our method produces 3D layout that closely matches the human-annotated ground truth. The viewport of Figure 11(a) is relatively narrow (only record some local regions of a scene). And there are many occlusions in the scene. The output 2D layout has some noise. However, our method can overcome these unfavorable factors. As we can see in the sixth row of Figure 11(a), the projections of 3D planes are close to the ground truth.

Figure 11(b), 11(c) and 11(d) show more cases in which edges are well located even though there are heavy occlu-

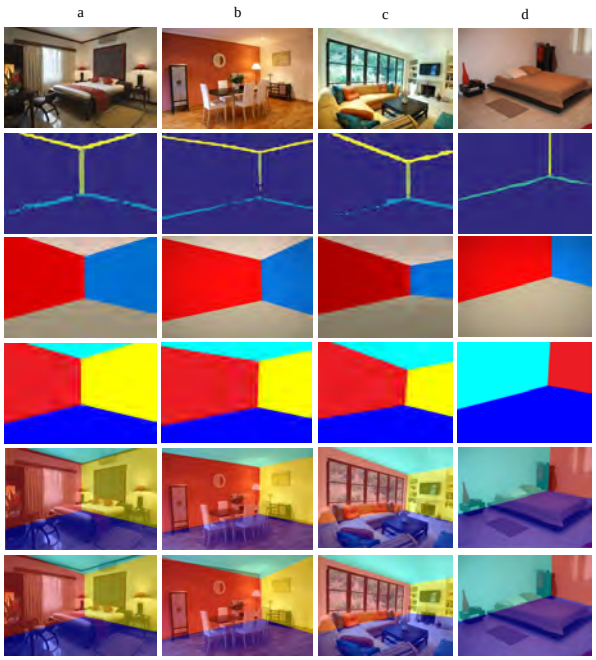

Fig. 12: Qualitative results on Hedau dataset. The image arrangement is the same as Figure 11.

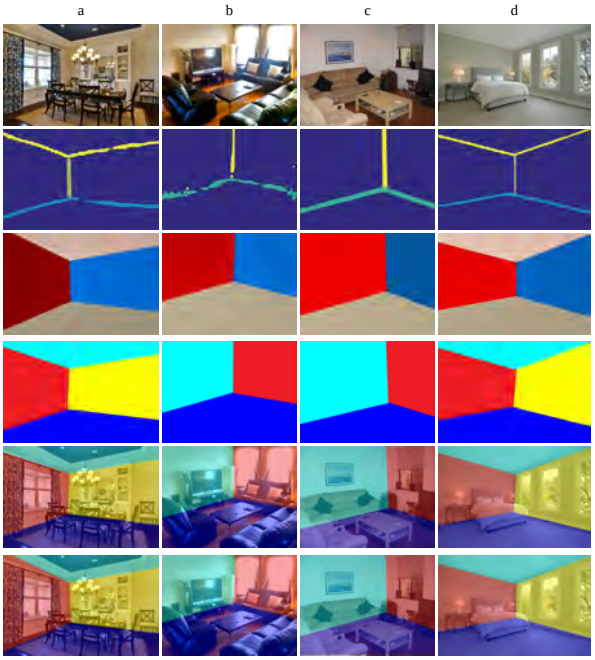

Fig. 13: Qualitative results on 3DGP dataset. The image arrangement is the same as Figure 11. For calculating pixel errors, we give different colors to the planes (floor, ceiling and walls) in the scene based on the original ground truth in the dataset.

sions in the scene. We automatically identify the topologies of the scenes and reconstruct them. The projection of the reconstructed 3D planes coincides with the ground truth.

Figure 11(e), 11(f) and 11(g) show the robustness of the second-stage of our pipeline ($TAPO$ and 3D reasoning). In these figures, the edge outputs of the network have some noticeable errors. In Figure 11(e) and 11(g), the edges between floors and walls are cracked. A worse situation appears in 11(f). The output of the network has obvious errors in the edge between the ceiling and the wall, because some textures that similar to edges in the scene may mislead the network. Given all these errors, our method is still able to handle these situations and calculate 3D plane models successfully.

Method	2D e_{pixel}	2D e_{corner}
Dasgupta [37]	10.63%	8.20%
Lee [39]	9.86%	6.30%
Zhao [7]	5.29%	3.84%
Ours	**5.07%**	**3.34%**

TABLE I: Performance on LSUN dataset. $*$ means we use 950 images in LSUN to do the testing, as there are some images do not satisfy the straight line assumption of our method.

Figure 11(h) shows the performance of our method when the scenes have a large depth of view. This scene also has a lot of occlusions. The edges between floor and walls are blocked by the bed and other furniture. The output of our network fails to locate edges well. Although edge predictions in label maps appear to be sparse, the heat maps still have values in the sparse corners and the corners can still be optimized. The result of 3D layout is still accurate. Notice that we can not handle all images in the test set because some images do not satisfy our assumptions. There are some failure cases in Figure 14.

Table I contains the final quantitative results on LSUN. The 2D pixel error and 2D corner error show that the performance of our TAPO is comparable with the state-of-the-art methods [7], [37], [39] or even better than them. Notice that since our method assumes the semantic edges are straight lines in image domain, we can not handle the distortion caused by wide angle cameras where the lines are recorded as curves. But on the other hand, our method automatically decides the topology, other than [7] which requires defining topology types in advance. Table I, IV show the performance of our method quantitatively.

B. Hedau Results

The Hedau dataset is presented by [16], being consisted of 209 training samples and 105 testing samples. As shown in Figure 12, our model extracts reasonable edges of the testing images. Based on the edges, we reconstruct the 3D layout topologies of the indoor scenes. The scenes in Figure 12(a) and 12(c) are relatively complex. There are a lot of objects in the scenes, which add difficulties to reconstruct the 3D topologies. We report better results than FCN-based methods like [7], [37], [39] (Table II, IV). According to the results and pixel errors, our method is reliable in dealing with these scenes. Notice that we have not trained our pipeline with the training set of Hedau dataset, so the good experimental results indicate the generality of our technique.

Method	2D e_{pixel}
Dasgupta [37]	9.73%
Lee [39]	8.36%
Zhao [7]	6.60%
Ours	**6.53%**

TABLE II: Performance on Hedau dataset.

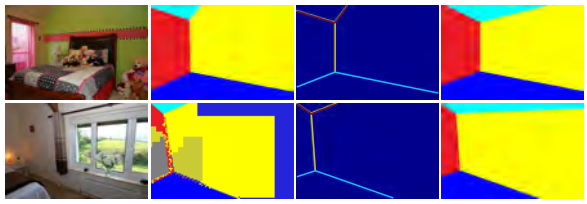

Fig. 14: Some failure cases on LSUN test set. The first column is the original images. The second column is the ground truth. The third column is the 2D layout topologies obtained by TAPO. The last is the projections of 3D layout results. The scenes in images do not satisfy our assumptions. The 2D layout topologies are qualitative. However, the 3D layout results have obvious errors.

C. 3DGP Results

In this subsection, we evaluate our method on 3DGP dataset which is used in [42]. It includes 963 images (622 images for training, 423 images for testing) of three types of scenes: living rooms, bedrooms and dining rooms. The 3DGP dataset has been used to evaluate layout estimation, object detection [53] and scene classification [54] simultaneously. For calculating pixel errors, we give different colors to the planes (floor, ceiling and walls) in the scene based on the original ground truth in the dataset. Again, we have not trained our model with its training set. There are some qualitative results in Figure 13. The scene in Figure 13(a) has some edges that are similar to wall-ceiling edges. The output of our network still has reliable performance. The 3D layout in Figure 13(a) is also accurate. Figure 13(b) shows the ability of our method to deal with a cluttered scene. Most of edges in the scene are blocked by sofas, tables and other objects. The output of our network has some noise. However, the $TAPO$ and 3D reasoning module of our pipeline still generate a fairly accurate result. The projections of the reconstructed 3D layout in the camera coordinate system have high consistency with the ground truth. Our performance on 3DGP are also summarized in Table III, IV. The results show that our method has the state-of-the-art performance. The error $3D\ e_{pixel}$ in Table IV is formulated as:

Method	2D e_{pixel}
Choi [42]	17.4%
Ours	**5.81%**

TABLE III: Performance comparison with [42] on 3DGP dataset.

$$3D\ e_{pixel} = \frac{\sum_{i\in[1,h],j\in[1,w]}(gt_{i,j}\ \&\ layout_{i,j})}{h \times w} \tag{11}$$

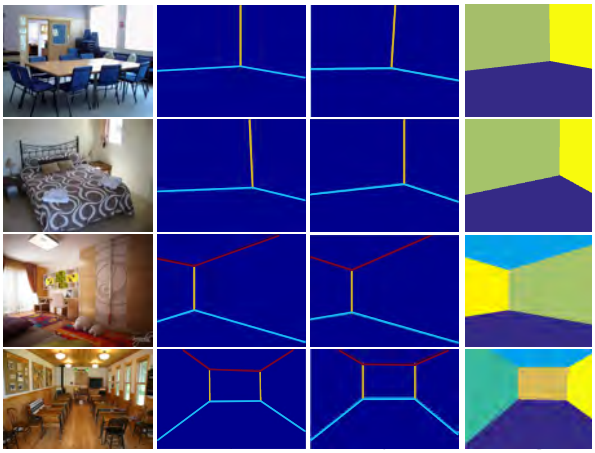

Fig. 15: Some results obtained with PIO and our method. The first column is the input images. The second column is our results. The third column is the results of PIO. The last column is the ground truth. In general the results of two methods are close.

where the gt denotes the 2D ground truth, & denotes the "and" operation. The layout denotes the projection of 3D layout.

Method	2D e_{pixel}	3D e_{pixel}
LSUN	4.69%	8.80%
Hedau	6.43%	8.21%
3DGP	5.36%	9.19%

TABLE IV: Average 2D and 3D pixel error on the test sets of LSUN, Hedau and 3DGP. 2D e_{pixel} is the pixel errors between the pixel labeling results and the ground truth. 3D e_{pixel} is the pixel errors between the final 3D layout and the ground truth, whose mathematical formula is shown in Equation 11.

D. Comparison

In [7], Zhao et. al proposed a method named Semantic Transfer to extract edge features under various circumstances and a method named Physics Inspired Optimization (PIO) to optimize the room layout with the edge output of a neural network. Because our proposed method also extract edge features, we do a comparison in this subsection. Notice that we reconstruct 3D layout while [7] only reconstructs 2D layout.

Figure 6 has qualitatively shown that our model has a better performance in edge features extraction. To quantitatively perform a comparison, we use the result of the two edge feature extraction methods as the input of our TAPO method, and the final results on the three datasets are shown in Table V. The 2D layout results using feature maps produced by our model have smaller average pixel errors because our feature maps have less noise and more precise than the feature maps of semantic transfer [7]. Table V indicates that the feature maps produced by our model are beneficial to layout estimation.

Topology anchor points are the intermediate output of our algorithm and are used for 3D layout reconstruction. PIO [7] needs to define 11 different possible room layout topologies

in advance. Then PIO uses each layout topology to fit feature maps and gets an optimal solution. Finally, PIO compares the energy of the 11 different optimal solutions and chooses the one with the lowest energy. Compared with PIO, the main advantage of our proposed method is that our method does not need to set any topologies in advance. It can automatically identify topology. As shown in Figure 15, the results of the two methods are very close, but our method does not define topologies in advance. There are some results obtained by PIO and also our method shown in Figure 15.

Dataset	Ours	Semantic Transfer [7]
LSUN	**4.69%**	6.87%
Hedau	**6.43%**	7.45%
3DGP	**5.36%**	6.75%

TABLE V: Average 2D pixel error obtained with our method using feature maps that are produced by our model and semantic transfer [7]. We use the same images for 3D layout reconstruction in Table IV. The result shows that our proposed CNN-based scheme has a better performance.

E. Limitation

The 3D layout that we produced has a scale ambiguity. The reason is that the distance from the camera center to the floor is unknown in our mathematical formulation. Let us first assume that the ratio of d_p and $d_{p'}$ is r. When d_p becomes x times of its original value, λ_k also changes x times according to Equation 12. So d_p and $d_{p'}$ still satisfy the ratio r, and thus Equation 13 holds at the same time. The 3D plane model can be determined uniquely if the distance from the camera center to the floor in the world coordinate system is known. If this assumption is correct in certain cases (e.g. setting the camera height to that of a human user or a field robot), our method can solve this ambiguity and reconstruct the absolute 3D layout of a scene based on a single RGB image.

$$\lambda_k \cdot n_p \cdot (K^{-1} q_k) = d_p \tag{12}$$

$$\lambda_k \cdot n_{p'} \cdot (K^{-1} q_k) = d_{p'} \tag{13}$$

Another limitation is that our method is based on the Manhattan assumption and we can not handle rooms with non-orthogonal walls, which happens for some model buildings. This limitation is also shown in Figure 14. Also, we can only reconstruct the 3D information of walls, floors and ceilings, but not objects in the scene, which we believe is a future direction to further explore the 3D information of a scene. The future work is to get rid of the Manhattan assumption.

V. Conclusion

In this paper, we propose a fully automatic solution to solve the problem of estimating the 3D layout of an indoor scene from a single 2D image. We first propose a method that directly regresses the room structure lines with a single stage, which outperforms the state-of-the-art three-stage method. Then we propose a novel *topology anchor point optimization* technique to automatically identify the layout topology of an indoor scene, followed by a nonlinear optimization with equality constraints to estimate the final 3D layout. We evaluate our method on three public datasets. Our extensive evaluations show that the proposed method works well and gives accurate 3D layout results with high efficiency and robustness. Despite the scale ambiguity of the output 3D layout, our method can be potentially combined with other modern technologies to enable more applications.

VI. Acknowledgment

This work was supported by the National Key R&D Program of China 2018YFA0704000, the NSFC (No.61822111, 61727808, 61671268, 61931008, 61671196), Beijing Natural Science Foundation (JQ19015, L182052), Zhejiang Province Nature Science Foundation of China (LR17F030006, Q19F010030) and the National Natural Science Major Foundation of the Research Instrumentation of PR China under Grants 61427808, 111 Project, No. D17019.

References

[1] H. Saito, S. Baba, and T. Kanade, "Appearance-based virtual view generation from multicamera videos captured in the 3-d room," *IEEE Transactions on Multimedia*, vol. 5, no. 3, pp. 303–316, 2003.

[2] E. Marder-Eppstein, "Project tango," in *ACM SIGGRAPH 2016 Real-Time Live!* ACM, 2016, p. 40.

[3] A. Sankar and S. M. Seitz, "Interactive room capture on 3d-aware mobile devices," in *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, 2017, pp. 415–426.

[4] G.-J. Qi, "Hierarchically gated deep networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2267–2275.

[5] C. Yan, H. Xie, J. Chen, Z. Zha, X. Hao, Y. Zhang, and Q. Dai, "A fast uyghur text detector for complex background images," *IEEE Transactions on Multimedia*, vol. 20, no. 12, pp. 3389–3398, 2018.

[6] H. Xie, S. Fang, Z.-J. Zha, Y. Yang, Y. Li, and Y. Zhang, "Convolutional attention networks for scene text recognition," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 15, no. 1s, p. 3, 2019.

[7] H. Zhao, M. Lu, A. Yao, Y. Guo, Y. Chen, and L. Zhang, "Physics inspired optimization on semantic transfer features: An alternative method for room layout estimation," *arXiv preprint arXiv:1707.00383*, 2017.

[8] C. Yan, Y. Tu, X. Wang, Y. Zhang, X. Hao, Y. Zhang, and Q. Dai, "Stat: spatial-temporal attention mechanism for video captioning," *IEEE Transactions on Multimedia*, 2019.

[9] J. Huang, Z. Liu, and Y. Wang, "Joint scene classification and segmentation based on hidden markov model," *IEEE Transactions on Multimedia*, vol. 7, no. 3, pp. 538–550, 2005.

[10] Z. Zhou, F. Farhat, and J. Z. Wang, "Detecting dominant vanishing points in natural scenes with application to composition-sensitive image retrieval," *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2651–2665, 2017.

[11] C. Yan, L. Li, C. Zhang, B. Liu, Y. Zhang, and Q. Dai, "Cross-modality bridging and knowledge transferring for image understanding," *IEEE Transactions on Multimedia*, 2019.

[12] J. Tang, L. Jin, Z. Li, and S. Gao, "Rgb-d object recognition via incorporating latent data structure and prior knowledge," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1899–1908, 2015.

[13] A. Wang, J. Lu, J. Cai, T.-J. Cham, and G. Wang, "Large-margin multi-modal deep learning for rgb-d object recognition," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1887–1898, 2015.

[14] C. Huang, Z. He, G. Cao, and W. Cao, "Task-driven progressive part localization for fine-grained object recognition," *IEEE Transactions on Multimedia*, vol. 18, no. 12, pp. 2372–2383, 2016.

[15] D. Hoiem, A. A. Efros, and M. Hebert, "Recovering surface layout from an image," *International Journal of Computer Vision*, vol. 75, no. 1, pp. 151–172, 2007.

[16] V. Hedau, D. Hoiem, and D. Forsyth, "Recovering the spatial layout of cluttered rooms," in *Computer vision, 2009 IEEE 12th international conference on*. IEEE, 2009, pp. 1849–1856.

[17] J. M. Coughlan and A. L. Yuille, "Manhattan world: Compass direction from a single image by bayesian inference," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2. IEEE, 1999, pp. 941–947.

[18] A. G. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun, "Efficient structured prediction for 3d indoor scene understanding," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2815–2822.

[19] L. Del Pero, J. Bowdish, D. Fried, B. Kermgard, E. Hartley, and K. Barnard, "Bayesian geometric modeling of indoor scenes," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2719–2726.

[20] H. Wang, S. Gould, and D. Roller, "Discriminative learning with latent variables for cluttered indoor scene understanding," *Communications of the ACM*, vol. 56, no. 4, pp. 92–99, 2013.

[21] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: a factorization method," *International Journal of Computer Vision*, vol. 9, no. 2, pp. 137–154, 1992.

[22] P. Favaro and S. Soatto, "Shape and radiance estimation from the information divergence of blurred images," in *European Conference on Computer Vision*. Springer, 2000, pp. 755–768.

[23] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, "Shape-from-shading: a survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 21, no. 8, pp. 690–706, 1999.

[24] A. Criminisi, I. Reid, and A. Zisserman, "Single view metrology," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 123–148, 2000.

[25] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 1996, pp. 11–20.

[26] H.-Y. Shum, M. Han, and R. Szeliski, "Interactive construction of 3d models from panoramic mosaics," in *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*. IEEE, 1998, pp. 427–433.

[27] P. Sturm and S. Maybank, "A method for interactive 3d reconstruction of piecewise planar objects from single images," in *The 10th British machine vision conference (BMVC'99)*. The British Machine Vision Association (BMVA), 1999, pp. 265–274.

[28] A. Kosaka and A. C. Kak, "Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties," *CVGIP: Image understanding*, vol. 56, no. 3, pp. 271–329, 1992.

[29] F. Han and S.-C. Zhu, "Bayesian reconstruction of 3d shapes and scenes from a single image," in *Higher-Level Knowledge in 3D Modeling and Motion Analysis, 2003. HLK 2003. First IEEE International Workshop on*. IEEE, 2003, pp. 12–20.

[30] E. Delage, H. Lee, and A. Y. Ng, "Automatic single-image 3d reconstructions of indoor manhattan world scenes," in *Robotics Research*. Springer, 2007, pp. 305–321.

[31] L. Del Pero, J. Bowdish, B. Kermgard, E. Hartley, and K. Barnard, "Understanding bayesian rooms using composite 3d object models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 153–160.

[32] Y. Zhang, S. Song, P. Tan, and J. Xiao, "Panocontext: A whole-room 3d context model for panoramic scene understanding," in *European conference on computer vision*. Springer, 2014, pp. 668–686.

[33] G. Schindler and F. Dellaert, "Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1. IEEE, 2004, pp. I–I.

[34] H. Xie, Z. Mao, Y. Zhang, H. Deng, C. Yan, and Z. Chen, "Double-bit quantization and index hashing for nearest neighbor search," *IEEE Transactions on Multimedia*, 2019.

[35] C. Yan, G. Biao, Y. Wei, and Y. Gao, "Deep multi-view enhancement hashing for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[36] A. Mallya and S. Lazebnik, "Learning informative edge maps for indoor scene layout prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 936–944.

[37] S. Dasgupta, K. Fang, K. Chen, and S. Savarese, "Delay: Robust spatial layout estimation for cluttered indoor scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 616–624.

[38] W. Zhang, W. Zhang, K. Liu, and J. Gu, "Learning to predict high-quality edge maps for room layout estimation," *IEEE Transactions on Multimedia*, vol. 19, no. 5, pp. 935–943, 2016.

[39] C.-Y. Lee, V. Badrinarayanan, T. Malisiewicz, and A. Rabinovich, "Roomnet: End-to-end room layout estimation," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4875–4884.

[40] 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 *European conference on computer vision*. Springer, 2016, pp. 628–644.

[41] C. Zou, A. Colburn, Q. Shan, and D. Hoiem, "Layoutnet: Reconstructing the 3d room layout from a single rgb image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2051–2059.

[42] W. Choi, Y.-W. Chao, C. Pantofaru, and S. Savarese, "Understanding indoor scenes using 3d geometric phrases," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 33–40.

[43] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2017.

[44] Y. Zhang, S. Song, E. Yumer, M. Savva, J.-Y. Lee, H. Jin, and T. Funkhouser, "Physically-based rendering for indoor scene understanding using convolutional neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 5057–5065.

[45] H. Xie, D. Yang, N. Sun, Z. Chen, and Y. Zhang, "Automated pulmonary nodule detection in ct images using deep convolutional neural networks," *Pattern Recognition*, vol. 85, pp. 109–119, 2019.

[46] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[47] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," in *CVPR*, 2017.

[48] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2881–2890.

[49] S. Song, S. P. Lichtenberg, and J. Xiao, "Sun rgb-d: A rgb-d scene understanding benchmark suite." in *CVPR*, vol. 5, 2015, p. 6.

[50] Y. Zhang, F. Yu, S. Song, P. Xu, A. Seff, and J. Xiao, "Large-scale scene understanding challenge: Room layout estimation," *accessed on Sep*, vol. 15, 2015.

[51] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[52] J. M. Coughlan and A. L. Yuille, "Manhattan world: Orientation and outlier detection by bayesian inference," *Neural computation*, vol. 15, no. 5, pp. 1063–1088, 2003.

[53] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[54] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 413–420.

Chenggang Yan received the B.S. degree in Computer Science from Shandong University in 2008 and Ph.D. degree in Computer Science from the Institute of Computing Technology, Chinese Academy of Sciences in 2013. Now he is the Director of Intelligent Information Processing Lab in Hanzhou Dianzi Univeristy. Before that, he was an assis-tant research fellow in Tsinghua University. His research interests include intelligent information processing, machine learning, image processing, computational biology and computational photography.

He has authored or co-authored over 50 refereed journal and conference papers. As a co-author, he got the Best Paper Awards in Pacific Rim Conference on Multimedia 2018, International Conference on Internet Multimedia Computing and Service 2018, and International Conference on Game Theory for Networks 2014, the Best Student Paper in International Conference on Multimedia and Expo 2019.

Biyao Shao received the Master degree in Controll Science and Engineer from Hangzhou Dianzi University in 2019. During his research, he worked on Semantic Segmentation and Reconstruction of indoor scenes. He has published articles on NEUROINFORMATICS and IEEE GlobalSIP 2018.

Hao Zhao received the B.S. degree in electronic engineering from Tsinghua University in 2013. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Tsinghua University. His research interest focuses on 3-D scene understanding with multi-modal sensory inputs.

Ruixin Ning received the B.S. degree in Automation from Northeastern University in 2017. He is currently pursuing the Master degree at Hangzhou Dianzi Universityw. His research interest focus on low-drift visual odometry in SLAM research field.

Yongdong Zhang(M'08-SM'13) received the Ph.D. degree in electronic engineering from Tianjin University, Tianjin, China, in 2002.He is currently a Professor with the School of Information Science and Technology, University of Science and Technology of China. His current research interests are in the fields of multimedia content analysis and understanding, multimedia content security, video encoding and streaming media technology. He has authored over 100 refereed journal and conference papers. He was a recipient of the Best Paper Awards in PCM 2013, ICIMCS 2013, and ICME 2010, the Best Paper Candidate in ICME 2011.He serves as an Associate Editor of IEEE Trans. on Multimedia and an Editorial Board Member of Multimedia Systems Journal.

Feng Xu is currently an associate professor in School of Software at Tsinghua University. He received a B.S. degree in physics from Tsinghua University, Beijing, China in 2007, and Ph.D. degree in automation from Tsinghua University, Beijing, China in 2012. His research interests include 3D reconstruction, performance capture and face animation.