International Journal of Communication Networks and Information Security
2024, 16(4)
ISSN: 2073-607X, 2076-0930
https://https://ijcnis.org/ **Research Article**

Deep ConvBi-LSTM: A Robust 3D Room Layout Estimation Model for Indoor Environment

Narendra Mohan[1], Manoj Kumar[2]*

[1]GLA University, Department of Computer Engineering and Applications, Mathura, India, 281406
[2*]GGV University, Department of Information Technology, Bilaspur, Chhattisgarh, India, 495009

Email: [1]narendra.mohan@gla.ac.in, [2]choubey.manoj@gmail.com

ARTICLE INFO

Received: 26 Apr 2024
Accepted: 02 Sep 2024

ABSTRACT

Room layout estimation is importance in recent times due to its extended application area. This process is highly challenging due to several factors affecting the room image such as clutter, occlusions, illuminations, etc. It is important to accurately identify the 3D layout of the room from a single 2D room image. The available techniques focused on determining the 3D layout but with limited number of features. It is important for a model to be fed with large number of features to result in successful predictions. To this extent, the proposed model introduced a robust 3D layout estimation framework for indoor environment. Initially, the input image is pre-processed and then subjected to layout estimation where our proposed model predicted both the edge maps and semantic labels for the image. For prediction, the proposed framework utilized the Deep ConvBi-LSTM model and a score function is defined and maximized by remora optimization algorithm (ROA) to obtain the optimal 2D layout from the candidate set. Finally, the 3D output is reconstructed from the 2D layout based on the layout coordinates and camera orientations. The experimental results of the proposed model proved the efficiency of the model in providing the desired performance.

Keywords: Indoor scene, layout estimation, 3D reconstruction, edge map, semantic label, deep ConvBi-LSTM, remora optimization algorithm.

INTRODUCTION

Room layout estimation is an important task that promotes the 3D indoor scene understanding. The problem is to identify the structure of the indoor room from a single

image. Alternatively, the layout estimation can be defined as the process of extracting the semantic boundaries of floor, walls and ceiling from a single image [1, 2]. This process of extracting the semantic boundaries from the image plays a major role in a wide range of applications such as scene reconstruction, indoor navigation, augmented reality (AR) and object detection. To enable effective object detection and recognition, it is highly important to understand the features related to the fixed background and movable objects [3, 4]. The 3D room layout can be depicted as a composition of orientation, wall and corner positions [5].

When there is an enough availability of images for scene classification, the complex 3D layouts can be estimated from the images by using a dense point cloud. But the single view compositions are far more difficult due to the presence of occlusions with low depth information [6, 7]. Most of the existing literatures focus on identifying the room layouts that comprise five major planes such as ceiling, right, left, front walls and ground [8]. The layout that is projected from the layout estimation task can be indicated as a projection of boundaries or corner positions or as a 3D mesh [9, 10]. The existing models that are based on deep learning (DL) are mostly used for estimating the room layouts from an image. Several assumptions are made about the room structures to generate better results [11].

The layout estimation problem has been extensively studied and several solutions are formulated in literatures that are based on single and multiple images [12-16]. Strong assumptions on the geometry of the indoor image are done in most of the methods i.e., Manhattan scenes and underfitting the richness of indoor spaces. Also, to deal with the ambiguities, wide fields of view are utilized by certain researchers such as 360° panoramas [17, 18]. The deep methods built are capable of learning the deep features for layout estimation. Scoring functions are followed in the models to rank the hypotheses generated [19]. But the major problem identified with these traditional methods is the vanishing points that require deep understanding of the scenes [20, 21].

It is important for any deep model to clearly understand the input features to generate optimal layouts. Most of the existing literature either extracted the edge maps or the semantic features for better labelling. To resolve the issue of providing inadequate information to the model, edge-semantic learning model [22] utilized both these features to train the network to generate desired outputs. But the drawback identified in the model is the problem of spatial redundancy thatlimited the information of neighboring pixels in the training process. Therefore, this research gap is addressed in the proposed model with effective combination of techniques. The sample images acquired from the LSUN dataset [23] is presented in Figure 1.

Figure 1: Sample images from the LSUN dataset for 11 different room types [23]

1.1 Motivation

Room layout estimation is a significant research area with a wide range of practical applications. The presence of clutter, large occlusions, low depth or missing information and limited visible range constitute the major challenges in room layout estimation. Low resolution due to the low illumination in the images makes it difficult to understand the layout of the room from a single image. Errors occur when the walls are far away from the camera and this leads to poor detection of geometrical points that are crucial for layout estimation. Another, serious issue is the accurate detection of edges and the problem arises when the edges are mixed with the cluttered indoor images. There are a lot of works introduced in literature to provide an accurate room layout in 3D with a 2D input image. In this work, a methodology for 3D layout estimation from 2D input image is presented. The major motivation of the work is to overcome the spatial redundancy in room layout estimation process as this problem is not adequately addressed in the existing literatures. The redundant pixels in the images disrupt the training process by limiting the amount of information for classification. To, achieve the desired performance, it is important to accurately discriminate every pixel of the input image. Apart from this, it is important to consider both the edge and semantic information present in an image to generate reliable output. Thus, the model considers both the information for layout refinement along with certain constraints to detail the refinement.

1.2 Contribution

The major contributions of the proposed work are as follows:

- Indoor room layout estimation is a challenging task and the estimation model requires intensive extraction capability to accurately estimate the layout of the room. This task processes a single image to obtain the layout but the problem identified is the spatial redundancy due to the huge amount of similar pixels. The proposed approach explores the Deep ConvBi-LSTM model for room layout estimation to deal with the spatial redundancy problem.
- Layout refinement is a crucial task in layout estimation problem that is subjected to several constraints. It is important to consider the major constraints for better refinement and to overcome the occlusions. The proposed model utilizes the Remora optimization algorithm (ROA) in the layout refinement and ranking phase subjected to surface smoothness, geometric and layout contour straightness constraints.
- 3D layouts provide clear visualization about the indoor layout of the room compared to a 2D output. Therefore, the proposed model reconstructs the 3D room layout from the optimal 2D layout with respect to Manhattan constraints.
- Extensive evaluations of the model are conducted to identify the performance of the model in layout estimation compared with the state-of-the-art models.

1.3 Paper organization

The remaining of the paper is structured as follows: Section 2 covers the literature review about the most recent techniques, section 3 presents our proposed methodology, section 4 presents the results and discussion and section 5 concludes the paper.

RELATED WORK

The generation of room layout using a single image is advantageous and popular in recent times with the advances in its application area. A simple approach named RoomNet presented by Lee et al. [24] estimated the layout keypoints initially and the locations of those key points in the room image was predicted by the auto-encoder framework. An alternative method for

layout estimation based on the semantic transfer features was designed by Zhao et al. [25]. Highly robust features from the indoor scene were initially extracted using the physics inspired optimization and the scene layout was predicted by the CNN. An approach named Pano2CAD to estimate the 3D layout of the room from a single 360° panorama image was introduced by Xu et al. [26]. The model considered the 2D and 3D objects for layout estimation and provided better results on the Sun360 dataset. Similar to RoomNet, an improve architecture named LayoutNet was developed by Zou et al. [27]. The approach aligned the images based on vanishing points and multiple layouts were predicted from which the optimal layout was identified by fitting the Manhattan layout into the estimated layout. An approach to automatically describe the floor plan from incomplete measurements gathered from autonomous robot was developed by Shariati et al. [28]. The free space regions and the room boundaries are exploited to provide guidance for motion planning systems.

To reconstruct a piece-wise planar depthmap from a single RGB image, Liu et al. [29] designed PlaneNet based on deep neural network (DNN). The model was end-to-end trainable and directly inferred the plane parameters and segmentation masks from the single input image. Unlike the above approaches, the estimation of geometric layout for indoor scenes based on latent variables was developed by Wang et al. [30]. In that approach, the location features were initially extracted and classified using the N-slack SSVM model. Then, the bag-of-words approach with cosine similarity and information divergence filtering was utilized to obtain the desired geometric layout. A model for layout estimation based on the deep refinement network was designed by Kruzhilov et al. [31]. This model replaced the traditional VGG-16 in auto-encoder with the ResNet50 where an iterative refinement structure was utilized to analyze the high and low-level features. An automatic indoor scene modeling by recovering the semantic contents, 3D geometry and relationship between objects was introduced by Nie et al. [32]. The approach was named Shallow2Deep that utilized the convolutional neural networks (CNNs) to extract the deep features. Another novel approach integrating scene grammar into the layout estimation task was reported by Purkait et al. [33]. On contrary to the existing grammar based approaches, the grammar was automatically constructed with the extraction of production rules based on the object co-occurrences.

An end-to-end approach to predict the 3D room layout using a single panoramic image was developed by Pintore and Gobbetti [34]. The AtlantaNet model projected the panoramic image into two horizontal planes and the 3D layout was predicted using an auto-encoder framework in which the long-range geometric patterns were captured. Another new approach using scanned scene for layout estimation was introduced by Avetisyan et al. [35] based on the inter-relationships between objects-to-layout and objects-to-objects. The object CAD models were used to obtain the geometric correspondences and a hierarchical layout prediction approach was implemented to obtain the layout. Unlike the normal literatures based on either semantic segmentation or edge/keypoint detection, Zhang et al. [36] introduced geometric reasoning into DL to accomplish the layout estimation task. The pixel-level surface parameters were predicted and depth maps were generated that were intersected to generate the layout of the scene. Normally, the layout is defined on a 2D image but Ren et al. [37] argued that impossible or invalid layouts were generated in those techniques. Therefore, an approach called layout via incremental movements (LIM) was devised to define the room layout on a 3D image based on spatial geometric representation. Another fully automatic solution for 3D layout estimation was rendered by Yan et al. [38] that estimated the layout from a single 2D RGB image. The structure lines of the room were estimated and the layout topology was identified in an automatic manner.

A framework called HoHoNet for holistic understanding of the indoor scene from a 360-degree panorama image was developed by Sun et al. [39] using latent horizontal feature (LHFeat). The model was fast in modelling dense modalities even with high resolution

panorama images. A DL based model called RackLay for real-time shelf layout estimation was introduced by Nigam et al. [40] from a monocular color image. Unlike the other estimation methods, RackLay provided the top and front view layout for every shelf to improve the overall accuracy. An end-to-end model for parametric layout estimation based on the input panorama image was developed by Zhao et al. [41]. The semantic maps were used as the intermediate domain and the implicit encoding strategy was utilized to embed the layouts into the latent space. An effective approach known as Deep3DLayout developed by Pintore et al. [42] exploited the crucial 3D properties from the room environment to reconstruct the 3D layout. The model used the graph convolutional network (GCN) to infer the room structure of spherical panoramic image. Unlike the other approaches, Zioulis et al. [43] developed a method to identify full room layout from a single-shot without the need for post-processing. The Manhattan-aligned outputs were directly inferred in that approach with the help of direct coordinate regression. Cluttered images are difficult to be processed and a mechanism of spatial layout estimation for those images was introduced by Dasgupta et al. [44]. The CNN with an optimization framework named DeLay was utilized by them to perform layout estimation.

Upon reviewing the existing literatures, it is seen that there are several effective techniques proposed to generate accurate layouts from the room images. Still there are open research challenges that are required to be well addressed. Most of the models either relied on the geometric information of the image and some methods used either the edge or semantic information to produce the layout of the room. One approach utilized both the semantic and edge information and obtained better results than the other models. We found it reliable and we made an attempt to use both this information to generate the 3D layout of the room. One of the major research gap that is not yet addressed in the field of room layout estimation is the spatial redundancy problem due to the presence of similar neighboring pixels. Therefore, an attempt is made here to overcome this problem whereas, generating optimal 3D layout for the input room image.

PROPOSED METHODOLOGY

Room layout estimation is the process of estimating the layout of the indoor scene from a single monocular image. This process involves the estimation of floors, ceilings and individual walls to identify the exact layout. In this paper, a novel method is introduced to extract the 3D room layout from a single input room image. The methodology is proposed with four main modules as follows:

- Pre-processing
- Layout estimation
- Layout refinement and ranking
- 3D reconstruction

Initially, the input image is pre-processed to improve the quality of the image and make it suitable for processing and estimation. After pre-processing, the input image is passed to the estimation model to exactly differentiate the pixels and label them. Based on the output of the estimation model, layout hypotheses sets are defined and the optimal layout is chosen as the 2D layout. Finally, the 3D layout is reconstructed based on the camera orientations and layout coordinates identified in the 2D layout. The overall architecture of the proposed room layout estimation model is depicted in Figure 2.

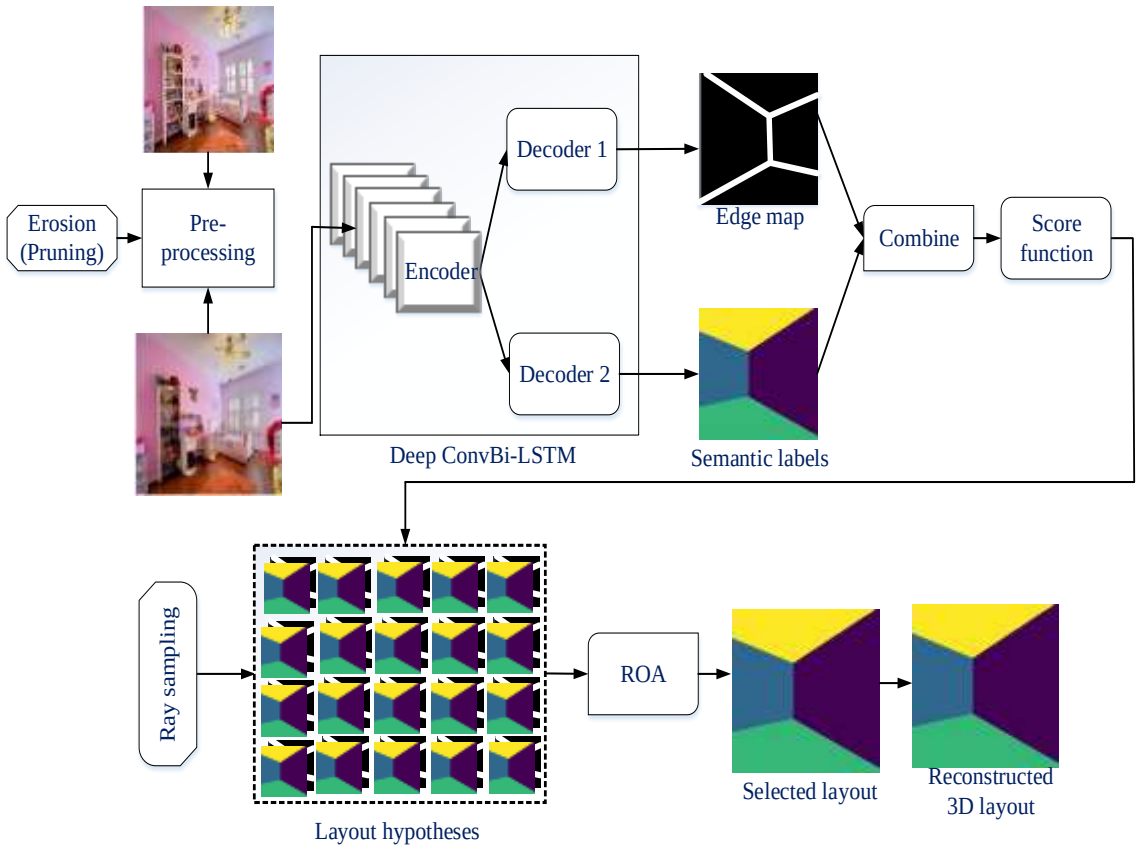


Figure 2: Architecture of the proposed 3D room layout estimation framework

The input image is initially acquired from the dataset and provided to the pre-processing module where the quality of the image is enhanced and made ready for prediction. The second module is the layout estimation module where a new auto-encoder is proposed to estimate the edge and semantic information simultaneously from the input image. This procedure is followed by the layout refinement and ranking module where an optimal layout is chosen using an optimization algorithm and a score function. The output of this module is referred as the 2D layout of the input image from which the 3D layout is reconstructed using the layout coordinates and camera orientations.

3.1 Pre-processing

The initial module of the proposed framework is pre-processing where the input image is pre-processed to remove the spurious regions and enhance the quality. The input to the proposed framework is a single 2D room image acquired from the dataset. Normally, the room image consists of several objects that disrupt the framework in appropriate detection of room layout. To enhance the layout estimation process, it is important to remove the objects from the image that make the layout more visible. At the pre-processing module, a popular morphological pruning operation known as erosion is carried out to detect and remove the unwanted object boundaries from the image. Using this operation, the edges of the objects are pruned that leads to the formation of holes. The mathematical formulation of erosion operation on the input room image can be expressed as follows:

$$P\Theta Q = \left\{ p \in P \middle| (Q)_p \subseteq P \right\} \quad (1)$$

where, P is the input image subjected to erosion, Q is the structuring element supporting erosion, $(Q)_p$ can be defined as the structuring element at pixel p. The structuring element is defined as a matrix to process the input image with one center pixel defined as its origin. The

pixel of interest i.e. the object boundary pixels from the room image are identified by this origin and then processed. The erosion is established as a looping operation where a single layer of pixels are eroded at each loop.
The erosion procedure leads to the formation of holes after the object boundaries being pruned. This disrupts the normal working of the proposed module as the holes might be wrongly recognized as a pixel value. To deal with this, the mean values of the neighboring pixels surrounding the holes are used to fill the pruned regions. By this step, all the pruned regions are filled and the image is made ready for layout estimation.

3.2 Layout estimation

Layout estimation is the most important module of the proposed framework where the edge and semantic information present in the input image are jointly utilized to obtain the edge and semantic labels. The joint learning of both the edge and semantic information helps the model to train better and to accurately define the layout. As mentioned earlier, one of the major motives of the proposed layout estimation step is to address the problem of spatial redundancy that arises due the presence of large amount of neighboring pixels.
The room image consists of walls that comprise similar pixels causing spatial redundancy. This problem limits the information available for training as well as lead to the generation of undesired layouts. Though the spatial information is important for 3D reconstruction of a scene, it is equally important to enhance the quality of training. The problem of spatial redundancy is addressed in this work through bilateral training. To this extent, this paper introduces the Deep ConvBi-LSTM auto-encoder that gets trained in both the forward and backward directions to increase the amount of information for training. This step enhances the accuracy in prediction of edge and semantic labels of the input image.
The pre-processed input image is provided to the proposed auto-encoder framework to obtain the edge and semantic labels. The encoder used in the proposed model is the ConvBi-LSTM [45] in which certain modifications in the convolutional layers are made to extract the deep features. The encoder model is made up of convolutional layers, flattening layer, Bi-LSTM layers, fully connected (FC) layer and classification layer with softmax at the end. The Bi-LSTM layers are responsible to get trained in both forward and backward directions to learn the pixels twice. This way of training improves the overall accuracy of training and helps in distinguishing the pixels of the input image. By this, the pixels can be labelled more accurately that helps in effective generation of 2D layout. The encoder model used in the proposed framework is depicted in Figure 3.

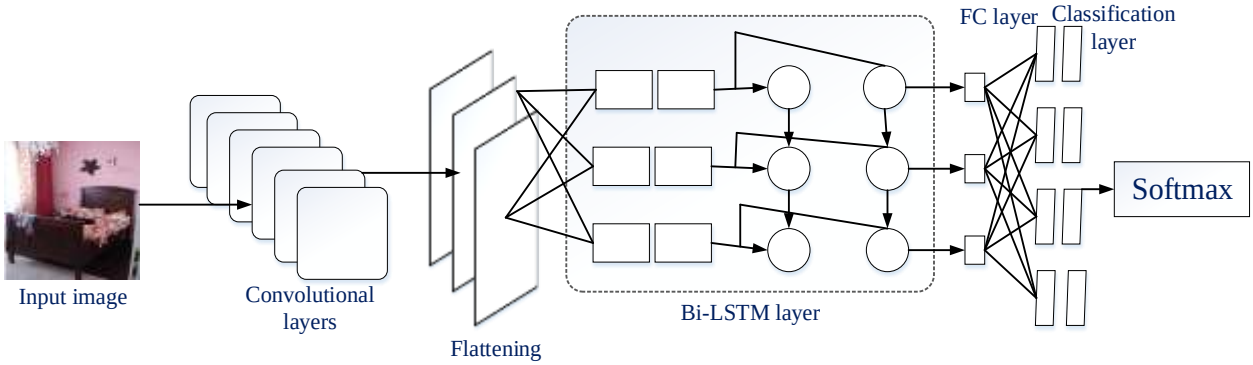


Figure 3: Encoder model utilized in the proposed work

Training the image pixels twice helps the model to gain more information about the input image. The output of the encoder is then passed to four convolutional layers that are followed by two decoders. The first decoder outputs the edge map of the input image and the second

decoder outputs five semantic labels (i.e. ceiling, floor, front wall, right wall and left wall) based on the input image. Each decoder is designed in a way as the reverse order of the encoder with de-convolutional layers to generate the desired output. In the training step, the input image is provided to the encoder where the major features are extracted for prediction.
The convolutional layers use the sliding window to extract the essential features from the input. The Bi-LSTM blocks in the encoder part extracts the features of the image through forward and backward training. The redundant pixels are trained twice that helps the model to identify the accurate labels for each pixel. The first decoder in the proposed model predicts the edge map for the input image. Five outputs are generated from the second decoder which is trained to produce five semantic heat maps for the room image. Based on the explanation in [46], the semantic labels of the input image are acquired. If only one wall is visible, then it is labelled as the front wall and if two walls are visible, then they are labelled as left and right wall.
The extraction process taking place in the convolution layer can be demonstrated using the following mathematical formulation:

$$\mathrm{X}_j^{\ell} = f\left(\sum \mathrm{X}_i^{\ell-1} \otimes k_j^{\ell} + b_j^{\ell}\right) \quad (2)$$

where, X is the input data, $\otimes$ indicates the convolution operation, k_j^{ℓ} is the convolution kernel of the ℓ^{th} layer and b_j^{ℓ} indicates the corresponding bias vector. The mathematical formulations for the forward and backward training taking place in the Bi-LSTM blocks can be demonstrated as follows:

$$\mathrm{I}_t = \sigma\left(\omega_{\mathrm{I}} \hat{\lambda}_{t-1} + \upsilon_{\mathrm{I}} k_t + b_{\mathrm{I}}\right) \quad (3)$$

where, I_t indicates the input gate, σ is the sigmoid activation, $\omega_{\mathrm{I}}\ and\ b_{\mathrm{I}}$ are the corresponding weight and bias vectors, k is the input to Bi-LSTM, $\hat{\lambda}_{t-1}$ is the output of forget gate and υ_{I} is the correlation coefficient.The forget gate computation can be given as follows:

$$F_t = \sigma\left(\omega_F \hat{\lambda}_{t-1} + \upsilon_F k_t + b_F\right) \quad (4)$$

where, F_t indicates the output of forget gate, $\omega_F\ and\ b_F$ are the weight and bias vectors of forget gate output and υ_F is the corresponding correlation coefficient. The overall output of the Bi-LSTM can be given as follows:

$$\mathrm{O}_t = \sigma\left(\omega_{\mathrm{O}} \hat{\lambda}_{t-1} + \upsilon_{\mathrm{O}} k_t + b_{\mathrm{O}}\right) \quad (5)$$

where, O_t is the output of output gate, $\omega_{\mathrm{O}}\ and\ b_{\mathrm{O}}$ are the weight and bias vectors belonging to output gate and υ_{O} is the corresponding correlation coefficient. These computations are made in both forward and backward directions to attain the output of Bi-LSTM blocks. The deep convolution layers utilized in the proposed model refines the output in a coarse-to-fine manner that results in increased robustness even in the presence of clutter. Thus, the redundant pixels in the input image get well-trained so that the layout estimation is more accurate.

3.3 Layout refinement and ranking

After the layout estimation process, the edge and estimated semantic labels are provided to the layout refinement and ranking module to obtain the optimal 2D layout of the input image. For this purpose, initially the semantic labels are combined into a single segmentation map using the following formulation:

$$\Psi(a,b) = \arg\max_{t} m^{t}(a,b); \quad \forall a,b \in [1,\ldots\varsigma] \quad (6)$$

where, $\Psi(a,b)$ is the segmentation map of the obtained semantic labels for a single input image, m^{t} are the five semantic labels obtained as output, (a,b) are the pixel coordinates of the image and ς indicates the size of the input image.

As per the definition of LSUN layout challenge [18], all the room image layouts can be covered with the defined 11 room layouts. All these 11 layouts are categorized as types and the room layouts are categorized based on the type and corner point coordinates. Thus, any parameterized room layout can be defined as $L=(\Gamma,\rho_1,\rho_2,....\rho_n)$ where, Γ indicates the type of layout and $\rho_1,\rho_2,....\rho_n$ are the corner points of the layout. Based on the type of layout, the importance of each corner point is determined. The functions that map the layout to the homogeneous edge map and segmentation map can be given as follows:

$$\Im(L)=\kappa(L),\quad \Psi(L)=\delta(L) \tag{7}$$

where, $\Im(L)$ indicates the edge map of L, $\Psi(L)$ indicates the segmentation map of L, $\kappa(L)$ is the function that maps the L to edge map and $\delta(L)$ is the function that maps the L to segmentation map.

The edge map predicted is for the complete room where the pixels are not considered. On the other hand, the segmentation map provides class labels for every pixel in the input image. In this case, the segmentation map is found to be advantageous as all the pixels can be distinguished from each other thereby improving the effectiveness of layout refinement. But the problem identified in the segmentation map is that it suffers from ambiguity issue. Thus, it becomes insufficient to estimate the layout with either one of the maps predicted. Thus, it is important to combine both these maps to generate accurate layouts. The combination can be done with the use of a scoring function as mentioned below:

$$Z\left(\Psi(L),\Im(L)\middle|\Psi,\Im\right)=Z_1\left(\Psi(L),\Psi\right)+\mu Z_2\left(\Im(L),\Im\right) \tag{8}$$

where, Z_1 is the pixel-wise accuracy of maximum bipartite matching taken for 2 segmentation maps, Z_2 indicates the negative Euclidean distance between the corners and walls and μ is a constant. Among the two segmentation maps taken, one is the predicted map and the other one is taken from the candidate layout. Here, cost is calculated by taking the label consistency of any two wall regions. Then, the bipartite matching procedure searched for the maximum cost function to obtain the pixel-wise accuracy. This reduces the ambiguity among the corners and wall regions [22].

3.3.1 Layout generation

To obtain the optimal 2D layout of the scene, initially the hypotheses set is generated in the proposed framework using the ray sampling method [47]. We adopt the methodology proposed in [48] to generate the layout hypotheses. For box layout representation, totally three vanishing points in three orthogonal directions are considered. On placing one vanishing point within the quadrilateral, the other two vanishing points are alone considered for candidate layout generation. The two vanishing points considered for sampling the rays**.** The mutually orthogonal vanishing points in the image are identified using the Rother's algorithm based on the robust voting and search schemes.

To obtain the candidate layouts through ray sampling, the two vanishing points are placed as two farthest points and the rays are sampled on either side of the center point. The intersections in these lines provide information about the corners of middle wall. The rest of the faces are generated with the connection of sampled rays. An example of ray sampling followed in the proposed work is depicted in Figure 4.

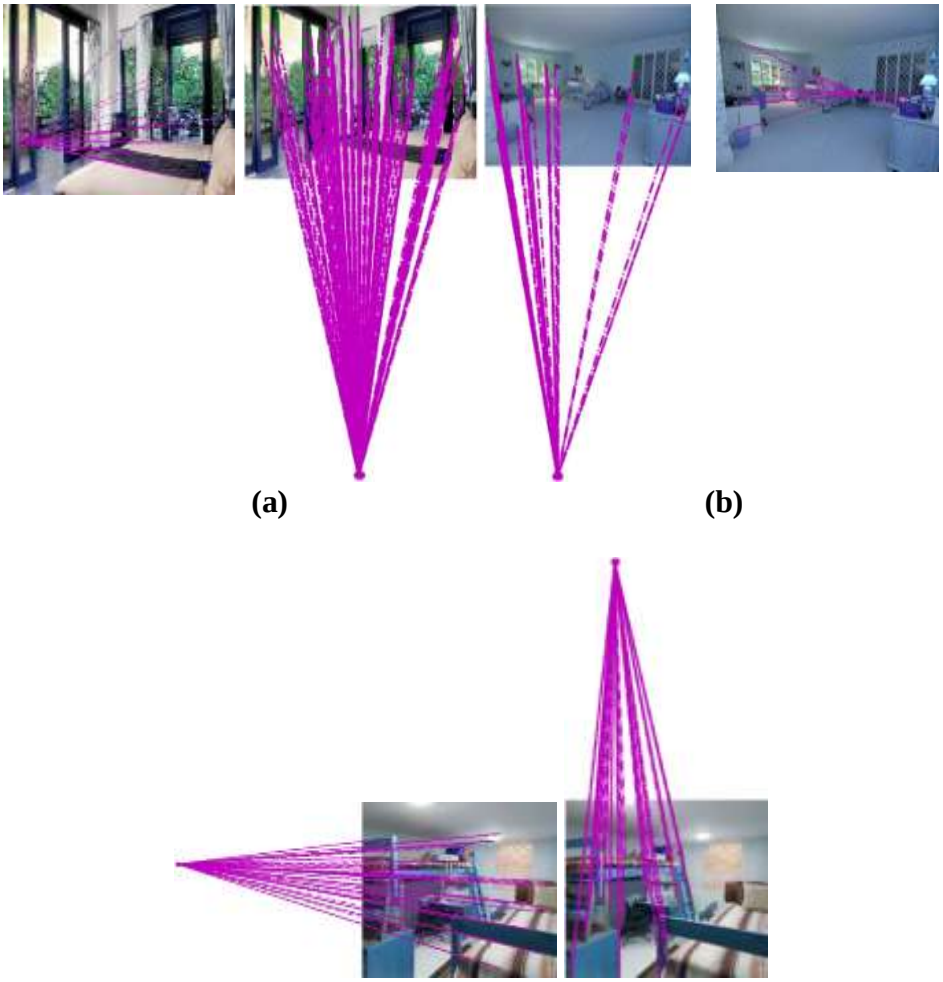


Figure 4: Examples of ray sampling for different indoor scenes

With this technique, numerous layouts can be generated from which the optimal one is yet to be identified. In our implementations, totally 20 layouts are generated for a single room image. To identify the optimal layout from the candidate layout set, the score function defined in equation (7) is required to be maximized.

3.3.2 Layout optimization

Let the candidate layout set obtained through ray sampling be defined as C and the problem here is to choose the best layout from this set. The best layout is identified by matching the score function secured by different layouts. The one that maximizes the score function is selected as the optimal layout. To strengthen the optimization process, it is subjected to the surface smoothness, geometric and layout contour straightness constraints to deal with the occluded boundaries. It is worthwhile to mention that the images with severe occlusions post challenges in accurate detection. So, to deal with the challenging images, the optimization process is subjected to the constraints to select the most appropriate layout.

3.3.2.1 Remora for layout optimization

To select the optimal layout by maximizing the score function, the ROA optimization [49] algorithm is utilized. This algorithm searches for the best candidate layout that accurately matches the layout of the room image by satisfying the respective constraints. This algorithm is inspired from the behaviors of remora, a marine fish belonging to the family Echeneidae. The intelligent behavior of these species in hunting and preying can be utilized in the proposed model to obtain the target layout with improved accuracy.

The main motive behind the selection of this algorithm for layout optimization is that it is dynamic in which there is a changing of host whenever a better host is found. This facilitates the algorithm to result in global optimal solution by completely surveying the search space. Apart from this, it is more effective than most of the existing optimizations in convergence rate. The major parameters in the traditional ROA algorithm are the remora (candidates), food (optimal), ships and marine species (tools). These parameters are exploited in our work to make it suitable for layout optimization. The remora is taken as the candidate layouts that are generated from the previous step, food indicates the optimal 2D layout and the tools indicate the boosting parameter to boost the optimization process. The fitness function is the score function defined in equation (8) under different constraints. The sequential steps are as follows:

Initially, a search environment is modelled with thresholds indicating the boundary of the environment. Then, the candidate layouts that are obtained from the last step is randomly placed in the search environment and those layouts that occur beyond the search space are discarded. The fitness function is defined for each candidate layout and the one that maximizes it will be selected as the optimal one. The proposed optimization model consists of two major phases such as the exploration and exploitation to identify the local and global layout.

3.3.2.1.1 Exploration

At this phase, the search environment is thoroughly scanned with the evaluation of fitness for each candidate layout to identify the better layout. The boosting parameter at this phase supports to identify the location of the layout in the search environment. After the identification of a layout in the environment, the fitness function is evaluated and the location is updated. The update formulation can be defined as follows:

$$C_i^{t+1} = C_{best}^t - \left(rnd[0,1] \times \left(\frac{C_{best}^t + C_{rnd}^t}{2} \right) - C_{rnd}^t \right) \quad (9)$$

where, t indicates the iteration number, C_{rnd} is the random location in the environment and C_{best} being the current best layout. After identifying the current best layout, the environment is analyzed to determine whether it is required to sort the candidate layouts. This is modelled mathematically as follows:

$$C_{srt} = C_i^t + \left(C_i^t - C_{pre} \right) \times rndn \quad (10)$$

where, C_{srt} indicates the sorted layout, C_{pre} is the previous location of the layout, $rndn$ is chosen at random and C_i^t is the current location of the layout.

3.3.2.1.2 Exploitation

After the sorting the candidates in the search environment, the complete environment is again scanned to determine the current optimal layout. The location update formulation for the sorted solution can be mathematically formulated as follows:

$$C_{i+1} = \Delta \times e^{\varphi} \times \cos(2\pi\varphi) + C_i \quad (11)$$

where, C_{i+1} is the new location, Δ is the distance between the previous and current optimal layouts in the search environment and φ is a random number between -1 and 1. After this step, the environment is reduced in size to improve the convergence rate in layout optimization. The mathematical formulation for this step is as follows:

$$C_i^t = C_i^t + \mathrm{A} \quad (12)$$

where, C_i^t is the current location of the layout and A is used to indicate a movement relevant to the volume space of candidates and host.

$$\mathrm{A} = \mathrm{B} \times \left(C_i^t - \mathrm{P} \times C_{best} \right) \quad (13)$$

where, P is the layout factor used to narrow the location space of the current layout and C_{best} Indicate the current best candidate.

$$B = 2 \times \mathrm{Y} \times rnd[0,1] - \mathrm{Y} \quad (14)$$

where, Y indicates a condition to determine the iteration and rnd indicates a random number between the range 0 and 1.

$$\mathrm{Y} = 2 \times \left(1 - \frac{t}{\mathrm{T}} \right) \quad (15)$$

where, t indicates the current iteration and T indicates the total number of iterations.

The above steps are repeated in an iterative fashion to identify the optimal layout from the candidate list. Finally, an optimal layout is obtained as the output of this step that constitutes the 2D layout of the input scene. The pseudo code of the proposed layout optimization is presented below:

Pseudo code for layout optimization
Initialize the candidate layout positions in the search environment
Initialize the optimal layout with its fitness function
While $t < Itr_{mx}$ **do**
Evaluate the fitness of every candidate layout in the search environment
Check if any candidate layout exists beyond the search environment and discard it
Update the values of φ *and* Y
For every layout in the search environment **do**
If $\Phi(i) = 0$ **then**
Update the position using equation (11)
Elseif $\Phi(i) = 1$ **then**
Update the position using equation (9)
Endif
Perform one-step prediction based on sorting using equation (10)
Identify the value of to judge whether it is required to change the host
If there is no need of changing the host, the environment size is reduced based on equation (12)
Endfor
End while

3.4 3D reconstruction

The optimal layout selected in the last phase is taken as the input in the 3D reconstruction step. For 3D reconstruction, the layout coordinates of the corners are computed to obtain the detailed information about the box layout. Spherical cameras capture scenes in a holistic manner and have been used for room layout estimation [50]. From the selected layout, the corner positions are obtained from which the camera positions are recovered. The box layout is incorporated into the estimated 2D layout for 3D reconstruction. Spherical cameras capture scenes in a holistic manner and have been used for room layout estimation.

An energy minimization function [51] is utilized in the proposed model to obtain the 3D layout of the indoor scene. The mathematical formulation for energy minimization can be given as follows:

$$E\left(L_{(\nu)},\Phi_{(c)}\right)=\min_{\Phi_{(c)},L_{(\nu)}}\sum_{(i,j)\in L_{(\nu)}}\left|\gamma\left(\Phi_i,\Phi_j\right)-\eta\left(\Phi_i,\Phi_j\right)\right| \quad (16)$$

where, is the 2D layout coordinate given as $L_{(\nu)}=\left\{\Phi_1=(0,0),\Phi_2=(q_1,r_1),.....\Phi_N=(q_N,r_N)\right\}$, $\Phi_{(c)}=\{q_c,r_c\}$ is the camera position, Φ_i,Φ_j are the neighboring vertices, η_{ij} indicates the pixel-wise distance between $\Phi_i \; and \; \Phi_j$ in a horizontal mannerdivided by the image length and γ_{ij} is the rotation angle given as $\gamma_{ij}=\arccos\frac{\Phi_i-\Phi_{(c)}\cdot\Phi_j-\Phi_{(c)}}{\left\|\Phi_i-\Phi_{(c)}\right\|\left\|\Phi_j-\Phi_{(c)}\right\|}$. The above equation (16) is solved efficiently as in [26].

Based on the layout coordinates such as the boundary and corner information, the 3D layout is reconstructed. Finally, the generated 3D layout is evaluated using the ceiling, floor and corner information gathered from the estimated layout. A score function is defined and then the obtained 3D layout is evaluated with the scoring function.

RESULTS AND DISCUSSION

Several experiments are carried out to prove the excellence of the proposed framework for 3D layout estimation. The working principle of the proposed framework can be summarized as follows: Initially, the room image is acquired from the dataset and subjected to pruning to remove the spurious regions. The erosion process is applied to remove the unwanted object boundaries so that the accuracy of training can be improved. After this process, the layout estimation step is carried out in which the Deep ConvBi-LSTM auto-encoder model predicted an edge map and five semantic labels for the input. Then, the hypotheses layout sets are constructed using the ray sampling method and optimal layout is searched and identified with the ROA algorithm. Finally, the 3D layout of the indoor scene is reconstructed from the 2D layout with the energy minimization function. The proposed framework is simulated and compared with the existing methods such as DeConvNet [47], DeLay [44], LayoutNet [27], joint learning [22] and 3D layout [38]. The simulation scenario, performance metrics and performance analysis of the proposed framework are illustrated in the upcoming sections.

4.1 Simulation scenario

The proposed framework is implemented in the python platform and tested using the large-scale scene understanding (LSUN) challenge dataset [23]. This dataset is recently introduced by the scene-centric large-scale challenges. It consists of 4000 training images, 1000 images for testing and 394 images for validation. There are a total of eight scene categories in this dataset such as bedroom, conference room, classroom, dining room, hotel room, dinette room, living room and office. All the images from the layout training samples are utilized to train the proposed framework to improve the training accuracy of the model.

The hyper-parameter setting followed in the proposed model is as follows: there are 4 convolutional layers, 4 de-convolutions, 1 fully connected layer, the batch size is set to 32×32, initial population is 20, maximum iterations are 200, maximum epochs are 300, and the total number of hosts are 2. The system configuration followed in the implementation process can be demonstrated as follows: The implementations are carried out in a system with Intel(R) Core (TM) i5-4670s processor running at 3.10 GHz on a 64-bit windows 10 operating system. The RAM installed in the system is 16 GB.

4.2 Performance metrics

The major performance metrics considered in the proposed work to evaluate the performance of the proposed model are pixel error, corner error, accuracy and error rate. The description and mathematical formulations are as follows:

4.2.1 Pixel error

Pixel error is referred to a single or numerous pixels in the image that are unable to display the required information. It is computed by determining the number of pixels that are unable to display the accurate room information. The value of pixel error is desired to be low to indicate better performance.

4.2.2 Corner error

Corner error can be defined as the deviations in the corner pixels in displaying the information compared to the ground truth. Lower values of corner error indicate that the system is effective in estimation.

4.2.3 Accuracy

Accuracy indicates the capability of the network model in exactly producing the edge and semantic labels for the image. The mathematical formulation for accuracy can be given as follows:

$$A = \frac{\mathrm{T}P + \mathrm{T}N}{\mathrm{T}P + \mathrm{T}N + FP + FN} \tag{17}$$

where, $\mathrm{T}P$ indicates true positives, $\mathrm{T}N$ indicates true negatives, FP is the false positive and FN is the false negative.

4.2.4 Error rate

The error rate is obtained by computing the mean squared error (MSE) of the network. This value is required to be low to indicate that the model is effective in prediction. The mathematical formulation for MSE can be given as follows:

$$\mathrm{M} = \frac{1}{n}\sum_{i=1}^{n}\left(\mathrm{Y}_i - \hat{\mathrm{Y}}_i\right)^2 \tag{18}$$

where, n is the total number of pixels for labeling, Y_i is the actual value and $\hat{\mathrm{Y}}_i$ is the predicted value.

4.3 Performance analysis

The performance of the proposed model is analyzed with the existing models that are implemented on the same dataset. The analysis has been conducted on different perspectives to identify the performance of the proposed framework in rendering the 3D layout of the room image. The performance analysis of the proposed framework on the basis of modules helps to understand the effectiveness of each step followed in 3D layout estimation. The deep analysis conducted and the performance comparison are detailed as follows:

4.3.1 Analysis based on performance metrics

The analysis of the proposed model based on different performance metrics is presented in this section. The major performance metrics considered are the pixel error and corner error that accurately determines the capability of the model is producing the edge and segmentation maps for the input image. The results of the proposed model is compared and analyzed with the existing models based on layout estimation. The performance comparison of pixel error for the proposed and existing models is presented in Table 1.

Table 1: Performance comparison of pixel error for the proposed and existing methods

Methods	Pixel error (%)
DeConvNet [47]	12.49
DeLay [44]	10.63
LayoutNet [27]	9.69
Joint learning [22]	6.58
3D layout [38]	4.69
Deep ConvBi-LSTM	**2.36**

The results of pixel error in the table 1 show the effectiveness of the proposed model compared to the other existing models. Among the compared models, the 3D layout model is nearly close to the proposed model and is accurate. The joint learning model also provided better performance in the testing phase as it learned both the edge and semantic information together like the proposed model for layout estimation. The DeConvNet model provided the least performance compared to the other models. This is because the information available to model is limited that restricted the model to learn the required features for layout estimation. On the other hand, the bilateral training enabled the model to learn more features in the training phase. This helped the model to get trained with all the required features for layout estimation. Moreover, the joint training concept is followed here to obtain better outputs for layout estimation. The compared models such as DeLay and LayoutNet also resulted in poor performance due to improper training and missing information about the layout coordinates. Thus, it can be concluded that the proposed model provided clear and precise edge maps compared to the other existing models. The overall pixel error of the proposed model is 2.36% whereas, the compared models such as DeConvNet, DeLay, Layout/Net, joint learning and 3D layout provided 12.49%, 10.63%, 9.69%, 6.58% and 4.69% respectively.

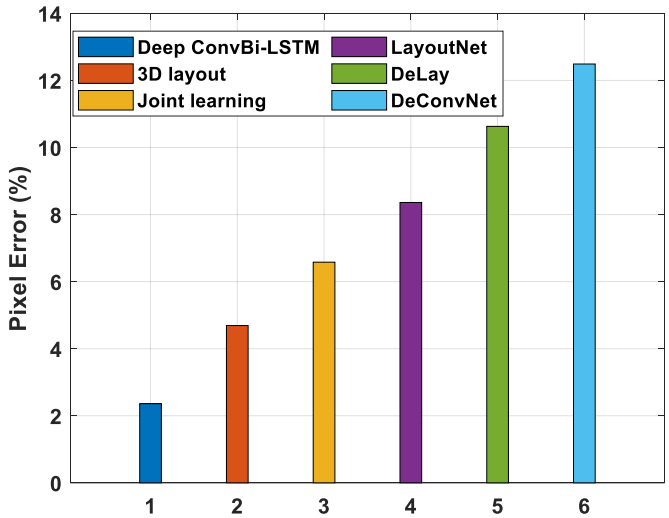


Figure 5: Pixel error comparison of the proposed and existing methods

The graphical depiction of the pixel error for the proposed and existing models is provided in Figure 5. From the figure, it is again clear that the edge maps and semantic labels produced by the proposed model is more accurate and effective than the output produced by the other models. The improved training procedure followed in the proposed model helped to achieve the desired performance.

Table 2: Performance comparison of corner error for the proposed and existing methods

Methods	Corner error (%)
DeConvNet [47]	8.70
DeLay [44]	8.20
LayoutNet [27]	7.89
Joint learning [22]	5.17
3D layout [38]	3.34
Deep ConvBi-LSTM	**1.23**

The performance comparison of corner error for the proposed and existing models is presented in Table 2. From the values, it is proved that the proposed model provided better prediction results compared to the existing models. While training the network for predictions, it is important to provide sufficient related features that pave way for accurate prediction. In the proposed model, the bilateral training concept examined each pixel twice and gained more features about the input image for training. This enhanced the model in accurately predicting the appropriate edge map and semantic labels for the image. The overall corner error of the proposed model is 1.23% and the corner error of DeConvNet, DeLay, LayoutNet, joint learning and 3D layout are 8.70%, 8.20%, 7.89%, 5.17% and 3.34% respectively.

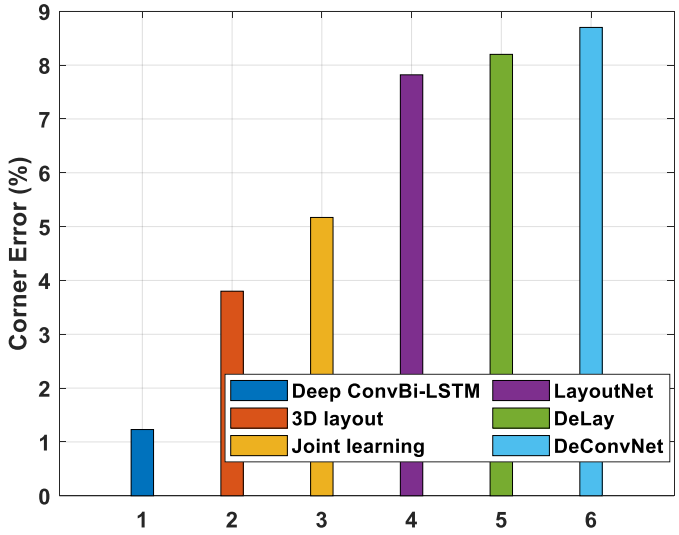


Figure 6: Corner error comparison of the proposed and existing methods

The graphical depiction of the corner error comparison for the proposed and existing models is provided in Figure 6. From the figure, it is clear that the proposed model is more optimal and can be applied to any kind of layout estimation tasks. The capability of the proposed model in discriminating each feature from the input image promoted the model to learn more number of features for every single image. Thus, the overall conclusion from the image and values is that the proposed model effectively learned the input features and performed successful prediction of edge map and semantic labels. The evaluation of pixel error and corner error also proves that the accuracy of the model is very high than the other models.

4.3.2 Model accuracy vs. model loss

The accuracy and loss of the model for the training and testing sets of the LSUN dataset is evaluated under this section. The accuracy of the model is one of the major concerns in any prediction process. It determines the capability of the prediction model in accurately labelling the output features. The loss value indicates that the model is not capable of accurately

identifying the edge map and semantic labels for the image. The evaluations of the proposed model in terms of model accuracy and model loss are detailed below:

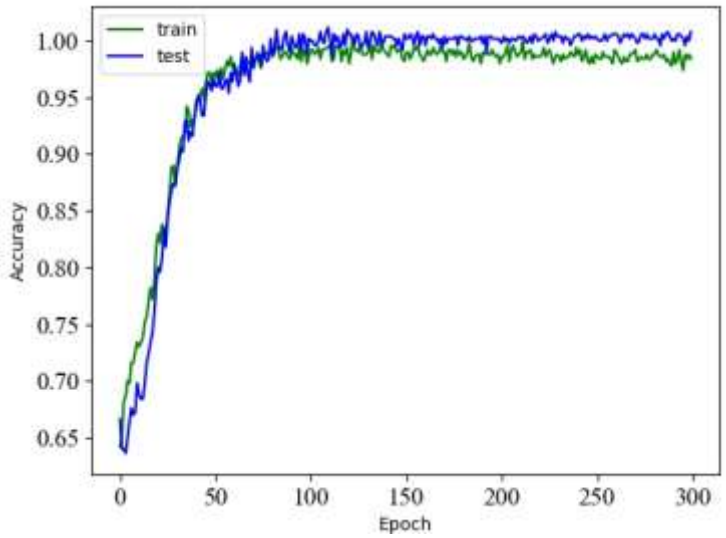


Figure 7: Model accuracy of the proposed layout estimation model

The model accuracy of the proposed layout estimation model is depicted in Figure 7. From the figure, it is clear that the proposed model has attained the highest accuracy rate in estimating the edge map and semantic labels of the image. At the initial stage, the accuracy level is low and gradually increased with the increase in epochs. In the figure, after 50 iterations, there is a increase in accuracy proving the model to be effective in training. Also, the graph shows almost equal accuracy level for both the training and testing curves. It is certain that there is no over-fitting issue identified in the training of the proposed model. Thus, it can be suggested that the model can be applied for layout estimation tasks to obtain better performance.

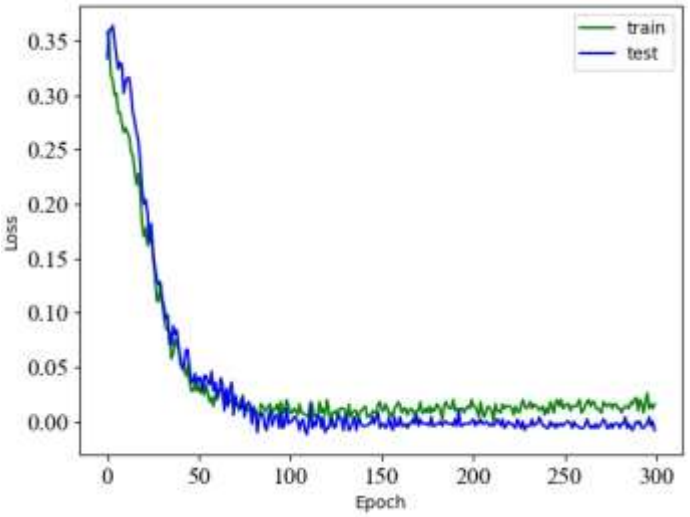


Figure 8: Model loss of the proposed layout estimation model

The loss curves of the proposed model on both the training and testing splits are graphically depicted in Figure 8. The values are plotted by varying the epoch size and the total number of epochs considered is 300. The graph shows a decreasing curve indicating that the loss value of the model is low. Initially, the loss value is high and when the epochs are increased, the

curve gradually decreased to a lower value. The distance between the two curves indicates that there is no over-fitting issue. Also, there is only a negligible error in the training phase indicating that the model is suitable for estimating the edge and segmentation maps from any kind of input image.

4.3.3 Module-wise evaluation

This section presents the results of the module-wise performance analysis compared with the existing models. In this section, the edge maps and layouts are compared with the existing models to identify the performance enhancement. The detailed evaluations are as follows:

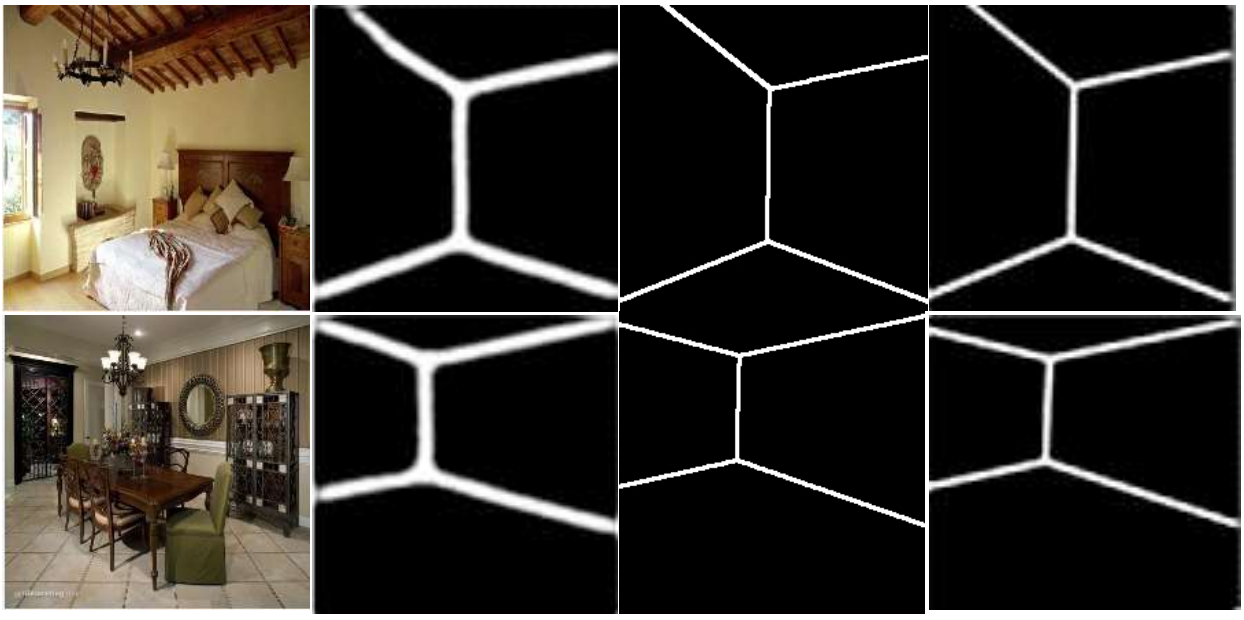
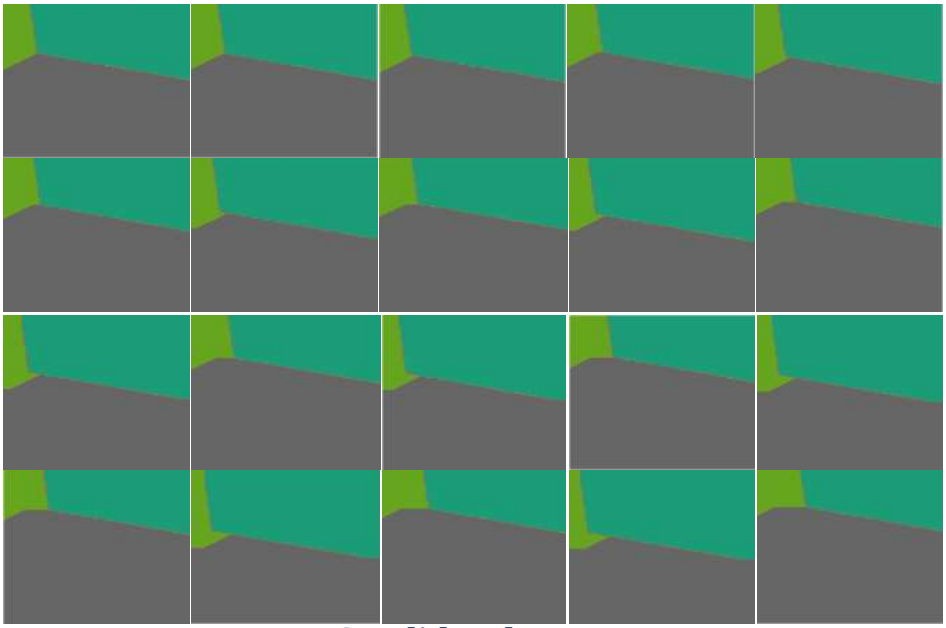


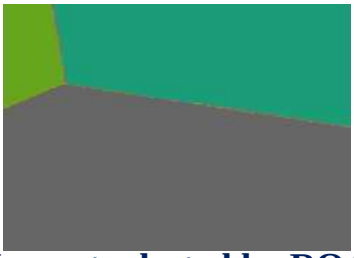
Figure 9: Comparison of edge map of the proposed and joint learning approach

In Figure 9, we have compared the edge map predicted by the proposed model with the native joint learning [22] approach. From the figure, it is seen that the edge map produced by the proposed model is more accurate without any distortions compared to the other model. The improved clarity of the edge map is due to the effective training carried out in the prediction phase. The forward and backward training of the auto-encoder model diminished the spatial redundancy and improved the amount of information available for learning. This enhanced the performance of the network in exactly labelling the output. Apart from this, the model also attained the desired performance in semantic labelling. This is because of the pixel wise training that trained each pixel in both forward and backward directions. The effective training enabled the network to accurately label each pixel based on its location in the image. Also, the deep convolutional layers helped the network to extract the relevant features for learning. This step reduced the pixel error in prediction and enabled the model to accurately obtain both the edge and semantic maps from the input image. The effectiveness of the model in generating accurate edge maps is more obvious in the second image in Figure 9, The edge boundary in the top-left corner of the predicted edge map is more accurate than the edge map predicted by the compared joint learning approach. Comparing to the ground truth, it is obvious that the predicted edge map of our approach is more accurate than the other model. It is also seen that the model is highly robust to occlusions and clutter than the compared approach. The pre-processing step helped the model to accurately determine the corners and boundaries of the image to predict the edge maps. This is proved in the second image where there are table and chairs hiding the lower edge boundary.

Input image



Candidate layouts



Layout selected by ROA

Figure 10: Visualization of layout selection by the proposed ROA layout optimization

The layout estimation step is enhanced with the optimization of score function that resulted in robust layout for the indoor image. The visualization results of layout selection by ROA is displayed in Figure 10. In the figure, totally 20 candidate layouts are generated through ray sampling and the optimal layout is selected by the ROA algorithm. The effective searching capability of the algorithm enabled the model to select the appropriate layout that well matches the input image. Also, the model is very fast in selecting the optimal layout due to the better convergence rate of the algorithm. This enhanced the efficiency of the proposed layout estimation framework though multiple steps are involved in it. Also, the algorithm is capable of differentiating the clutters and occlusions through proper scanning and fitness evaluations. The score matching concept helped the algorithm to match each layout with the other one in the search environment to find the optimal one. Apart from this, the geometric and spatial constraints established in the algorithm restricted the chance of selecting occluded layout as optimal.

It is seen in the image that the candidate layouts generated in the first row are almost equal that may result in lack of population diversity. However, there are significant deviations in the layouts present in the other rows. By this, it can be inferred that the model is more accurate in identifying even the smaller deviations among the layouts. The optimal layout selected by the proposed ROA model is more accurate and equivalent to the input image. Thus, the proposed layout estimation model can be recommended to be applied on layout optimization in real time applications.

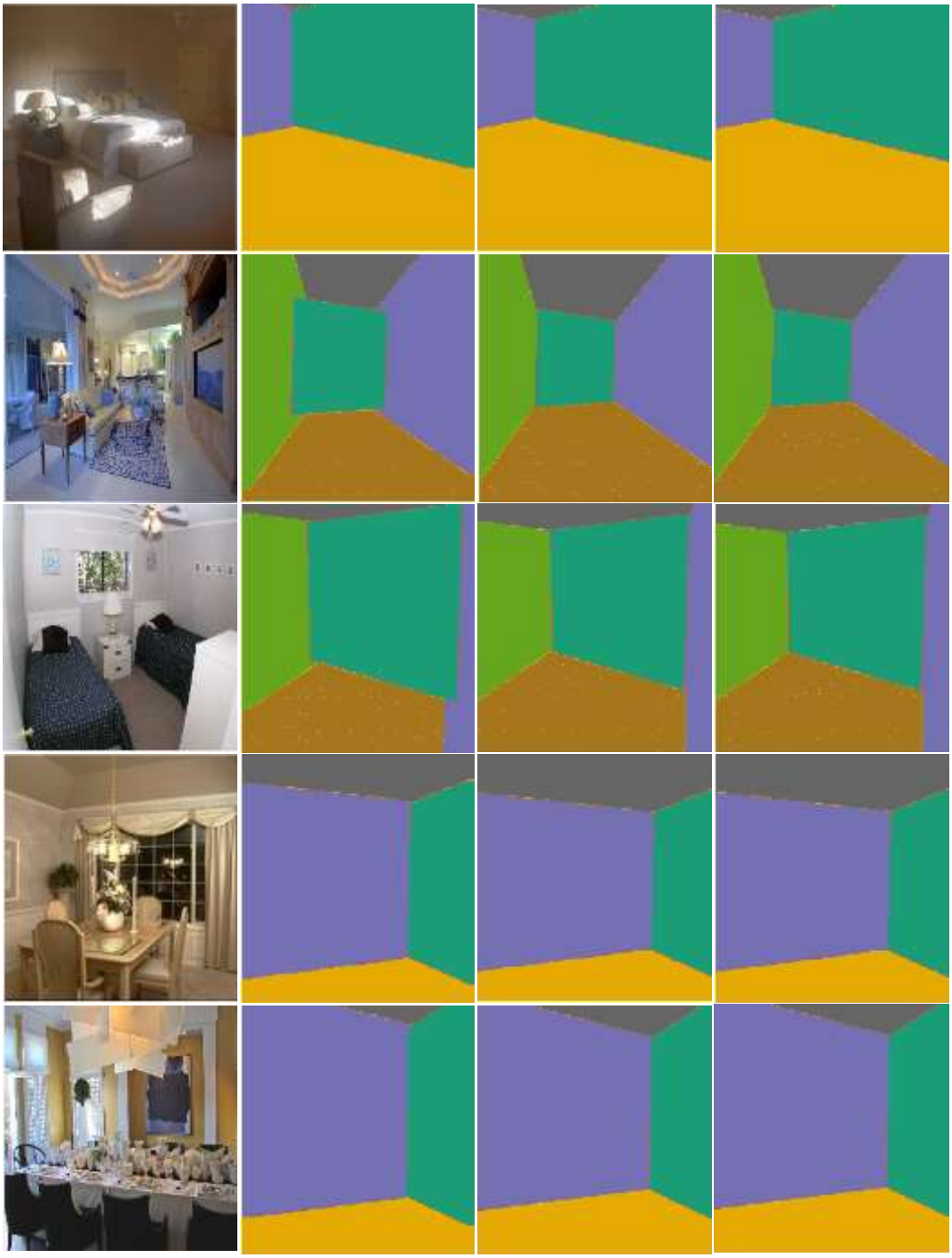


Figure 11: Comparison of the layout obtained by the proposed and existing method

The performance comparison of the layouts obtained by the proposed and existing method is displayed in Figure 11. From the figure, it is clear that the proposed model is more accurate in layout estimation than the other model. In the above figure, the first column indicates the input images, second column are the layouts produced by the joint learning approach, third column shows the results provided by our approach and the final column are the ground truth layouts. The results of five different images with clutter and illumination effects are presented in the figure to show the robustness of the model in layout estimation. In the first figure, there is a problem of illumination effect and the scene is not clearly visible. This is one of the major problem from which the existing models suffer in accurately determining the layout without any occlusions. The results displayed in the figure illustrate the performance of layout estimation for both the proposed and joint learning approaches. The output produced by the joint learning approach is almost similar to ground truth but there is a small deviation in the upper left corner. The output of the proposed model is more accurate and is able to determine the layouts even in the presence of illuminations. Compared to the ground truth of first image, the output rendered by the proposed model is closer to ground truth than the output of the other model.

The second input image displayed in the figure also involves complexity as the layouts in the image are not clearly defined. Moreover, the objects in the image are more than the first image. This post challenges for the model to accurately determine the layout boundaries and corners. On viewing the output rendered by the joint learning approach, there is a slight deviation in the recognition of front wall. This is because of the lack of proper training of the model in learning the pixels of the input image. The major advantage of comparing the proposed and joint learning approaches is that both use the edge and semantic information together for layout estimation. The significance of the proposed model from the existing model is with the training process. The training of the proposed model is enhanced with the forward and backward training procedure. This helped the model to learn the pixels twice that resulted in higher discriminative capability. Thus accurate predictions of labels are achieved by the proposed model. Also, it is viewed in the output of the second image produced by the proposed model. Compared to the existing model's output the proposed model's output better matches the ground truth.

In the third image, the layout definition available in the image is much better than the last two images. The output of the joint learning approach still involved occlusion at the lower right corner of the layout. This is because of the presence of object in the image that disturbed the layout output. The proposed model provided optimal output in this case due to the effectiveness of training. Similarly, the fourth image also provides almost equal layout definition as in third image. The outputs in this case are better for both the models as the input image is much better with high clarity. Only minor deviations are provided by the existing model whereas, the proposed model provided the exact layout as in ground truth for this image. The final input image is more complex with more objects and the ceiling is not entirely visible. This image requires effective training to generate high quality edge and semantic labels. The layout generated by the existing method is slightly deviated in the ceiling part due to the occlusions. The proposed model resulted in almost similar output as in ground truth proving its efficiency in layout estimation. The overall comparison of the proposed and existing approaches on layout estimation proved the efficacy of the proposed model. Also, the refinement strategy utilized in the proposed model improved the quality of layout produced as output.

Figure 12: Visualization results of edge map and layout obtained by the proposed approach for different room types

The visualization of the results obtained from different modules on different types of room images for the proposed model is displayed in Figure 12. The images considered in this figure are provided with different amount of layout definitions. This way of comparison enables to understand the efficiency of the proposed model in generating the desired layouts. The first column in the figure are the input images, second column presents the predicted edge maps for each image, third column presents the layout estimated, fourth column is the ground truth

and fifth column is the comparison between the original and generated layout. The images chosen in this visualization are of different complexities.

The predicted edge maps in all the images are more accurate and precise. In all the images, the edge maps predicted are effective without any form of occlusions. This is achieved with the effective training process carried out in the proposed model. The learning of the each pixel in the image twice enhanced the model in accurate labelling of the images. Also, the layout generated is more optimal and equally matches the ground truth layout without any occlusions. This is because of the ROA model utilized in the proposed approach to choose the optimal layout. The refinement step followed in the model helped to achieve the desired performance. Almost all the images are well trained with the auto-encoder model and the ROA model has chosen the appropriate layout for all the images.

The enhancement in the training phase with the addition of convolutional layers helped the model to extract the relevant features that best describe the layout of the room. Also, the searching procedure as a refinement step provided better outcomes. The higher convergence rate of the algorithm provided fast results in layout refinement. In the final column, the output obtained is compared with the original input image. For all the images considered, the proposed layout matched with the existing layouts of the scenes. Thus, the model can be inferred optimal and can be recommended to be applied in any form of cluttered images to obtain the optimal layout. Moreover, the complexity of the proposed approach is lower due to the collaboration of intelligent approaches that reduced the computational time taken even for complex processes.

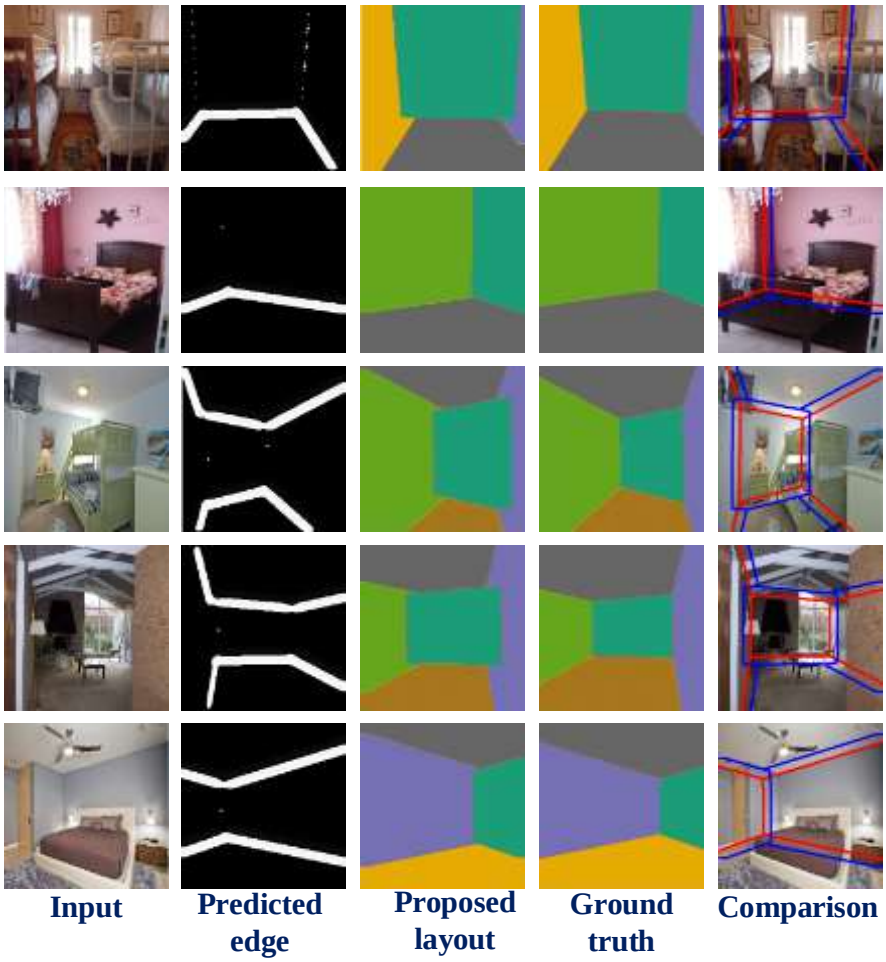


Figure 13: Some of the disturbed outputs provided by the proposed framework

The visualization of some of the disturbed outputs rendered by the proposed model is displayed in Figure 13. Different occluded and cluttered scenes are considered in this visualization to analyze the worst-case scenarios of the model. The first column is the input image, second column is the predicted edge map, third column is the proposed layout, fourth column is the ground truth and fifth column is the overlay comparison between the layout and original image. The predicted edge maps in these images are slightly occluded. There are some form of edge and semantic errors in the prediction phase. Because of this, the layouts generated are also occluded leading to some deviated outcomes. When compared to the input image, it is identified that there are minor errors in predictions. The differences in the overlaid lines indicate the edge and semantic errors in prediction. Upon further investigations, it is identified that the model produced only minor deviations that are mostly negligible.

From the overall simulations, it is identified that the proposed model provided optimal outcomes and results in better outputs than the existing models. The pixel and corner errors of the proposed model are compared with the existing models and the results suggested that the model is more suitable for 3D layout estimation than the other models. Deep analysis of the model shows the effectiveness and efficiency of the model in layout estimation. The comparison of the edge maps and layouts generated with the existing scheme inferred the performance improvement attained by the propose model. The effectiveness of training is proved through experiments. Also, the intelligent behavior adopted in the layout refinement strategy enhanced the model in generating the optimal layout as output. It is important to generate the layout hypotheses for different applications to better illustrate the layout of the room. Thus, the refinement strategy has been improved with highly converging ROA algorithm. Moreover, the proposed prediction model provided better predictions even on cluttered indoor scenes. Thus, the overall simulations suggest that the model can be applied to any form of layout estimation task in practical applications.

CONCLUSION

In this manuscript, a new methodology for room layout estimation has been presented. Initially, the dataset image is acquired and subjected to pre-processing to remove the spurious regions. The morphological operation known as erosion is put forth to remove the unwanted object boundaries. After this step, the edge and semantic labels for the input image are predicted using the deep ConvBi-LSTM auto-encoder. Then, the hypotheses layouts are generated through ray sampling from which the optimal layout is selected using the ROA algorithm. Finally, the 3D layout is reconstructed from the selected 2D layout by determining the layout coordinates and camera orientations. The performance of the model is tested using the LSUN dataset and the entire implementations are carried out in **on** the python platform. The results of the proposed model proved the efficiency of the model in optimally obtaining the 3D layout for the input image. The average pixel-error and corner error of the proposed model are 2.36% and 1.23%. The combination of edge and semantic information for layout estimation provided benefits in obtaining the 2D layout of the scene. Also, the spatial images are learned twice to deal with the spatial redundancies. This enhanced the amount of information available to the network in the training phase.

In future, it is aimed to conduct several examinations regarding the importance of combining both the edge and semantic information for layout estimation. Also, the research can be extended for the panorama images with depth information to produce more reliable 3D layouts.

REFERENCES

[1] Mathew, Bincy P., and SmithaDharan. "Review on room layout estimation from a single image." *Int. J. Eng. Res. Technol.* 9, no. 6 (2020): 1068-1073.

[2] Chen, Junming, Jie Shao, Dongyang Zhang, and Xuehui Wu. "A Fast End-to-End Method with Style Transfer for Room Layout Estimation." In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 964-969. IEEE, 2019.

[3] Hirzer, Martin, Vincent Lepetit, and PETER ROTH. "Smart hypothesis generation for efficient and robust room layout estimation." In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2912-2920. 2020.

[4] Yao, Hui, Jun Miao, Guoxiang Zhang, and Jun Chu. "3D layout estimation of general rooms based on ordinal semantic segmentation." *IET Computer Vision* (2023).

[5] Choi, Dongho. "3D Room Layout Estimation Beyond the Manhattan World Assumption." *arXiv preprint arXiv:2009.02857* (2020).

[6] Stekovic, Sinisa, ShreyasHampali, Mahdi Rad, Sayan Deb Sarkar, Friedrich Fraundorfer, and Vincent Lepetit. "General 3D Room Layout from a Single View by Render-and-Compare." In *European Conference on Computer Vision*, pp. 187-203. Springer, Cham, 2020.

[7] Chen, Xiaowei, and Guoliang Fan. "Indoor Camera Pose Estimation from Room Layouts and Image Outer Corners." *IEEE Transactions on Multimedia* (2023).

[8] Hsiao, Chi-Wei, Cheng Sun, Min Sun, and Hwann-Tzong Chen. "Flat2layout: Flat representation for estimating layout of general room types." *arXiv preprint arXiv:1905.12571* (2019).

[9] Zou, Chuhang, Jheng-Wei Su, Chi-Han Peng, Alex Colburn, Qi Shan, Peter Wonka, Hung-Kuo Chu, and Derek Hoiem. "3d manhattan room layout reconstruction from a single 360 image."CoRR, abs/1910.04099, (2019).

[10] Chen, Chen, and Ziwen Liu. "A Fast Method for Identifying Room Configurations from Unit Boundaries in Existing Residential Buildings." *Buildings* 13, no. 2 (2023): 357.

[11] Sun, Cheng, Chi-Wei Hsiao, Min Sun, and Hwann-Tzong Chen. "Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1047-1056. 2019.

[12] Huang, Rong-Ze, Yin-Bo Liu, Meh Jabeen, and Qing-HaoMeng. "Indoor Layout Estimation by Fusing Monocular RGB Image Features Extracted with HRNet." In *2020 39th Chinese Control Conference (CCC)*, pp. 7412-7417. IEEE, 2020.

[13] Zhang, Weidong, Qian Zhang, Wei Zhang, JianjunGu, and Yibin Li. "From Edge to Keypoint: An End-to-End Framework for Indoor Layout Estimation." *IEEE Transactions on Multimedia* (2020).

[14] Yang, Shang-Ta, Fu-En Wang, Chi-Han Peng, Peter Wonka, Min Sun, and Hung-Kuo Chu. "Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3363-3372. 2019.

[15] Huang, Jiahui, Zheng-FeiKuang, Fang-Lue Zhang, and Tai-Jiang Mu. "WallNet: Reconstructing General Room Layouts from RGB Images." *Graphical Models* 111 (2020): 101076.

[16] Li, Mingyang, Yi Zhou, Ming Meng, Yuehua Wang, and Zhong Zhou. "3d room reconstruction from a single fisheye image." In *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8. IEEE, 2019.

[17] Fernandez-Labrador, Clara, Jose M. Facil, Alejandro Perez-Yus, CédricDemonceaux, Javier Civera, and Jose J. Guerrero. "Corners for layout: End-to-end layout recovery from 360 images." *IEEE Robotics and Automation Letters* 5, no. 2 (2020): 1255-1262.

[18] Albanis, Georgios, Vasileios Gkitsas, Nikolaos Zioulis, Stefanie Onsori-Wechtitsch, Richard Whitehand, Per Ström, and Dimitrios Zarpalas. "An AI-Based System Offering Automatic DR-Enhanced AR for Indoor Scenes." In *Advanced Intelligent Virtual Reality Technologies: Proceedings of 6th International Conference on Artificial Intelligence and Virtual Reality (AIVR 2022)*, pp. 187-199. Singapore: Springer Nature Singapore, 2023.

[19] Li, Jieyu, and Robert L. Stevenson. "Indoor layout estimation by 2d lidar and camera fusion." *Electronic Imaging* 2020, no. 14 (2020): 391-1.

[20] Boniardi, Federico, AbhinavValada, Rohit Mohan, Tim Caselitz, and Wolfram Burgard. "Robot localization in floor plans using a room layout edge extraction network." In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5291-5297. IEEE, 2019.

[21] Han, Jiali, Yuzhou Liu, Mengqi Rong, Xianwei Zheng, and Shuhan Shen. "FloorUSG: Indoor floorplan reconstruction by unifying 2D semantics and 3D geometry." *ISPRS Journal of Photogrammetry and Remote Sensing* 196 (2023): 490-501.

[22] Zhang, Weidong, Wei Zhang, and Jason Gu. "Edge-semantic learning strategy for layout estimation in indoor environment." *IEEE transactions on cybernetics* 50, no. 6 (2019): 2730-2739.

[23] Zhang, Yinda, Fisher Yu, Shuran Song, PingmeiXu, Ari Seff, and Jianxiong Xiao. "Large-scale scene understanding challenge: Room layout estimation." In *CVPR Workshop*. 2015.

[24] Lee, Chen-Yu, Vijay Badrinarayanan, Tomasz Malisiewicz, and Andrew Rabinovich. "Roomnet: End-to-end room layout estimation." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4865-4874. 2017.

[25] Zhao, Hao, Ming Lu, Anbang Yao, YiwenGuo, Yurong Chen, and Li Zhang. "Physics inspired optimization on semantic transfer features: An alternative method for room layout estimation." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 10-18. 2017.

[26] Xu, Jiu, BjörnStenger, TommiKerola, and Tony Tung. "Pano2cad: Room layout from a single panorama image." In *2017 IEEE winter conference on applications of computer vision (WACV)*, pp. 354-362. IEEE, 2017.

[27] Zou, Chuhang, Alex Colburn, Qi Shan, and Derek Hoiem. "Layoutnet: Reconstructing the 3d room layout from a single rgb image." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2051-2059. 2018.

[28] Shariati, Armon, Bernd Pfrommer, and Camillo J. Taylor. "Predictive and semantic layout estimation for robotic applications in manhattan worlds." *arXiv preprint arXiv:1811.07442* (2018).

[29] Liu, Chen, Jimei Yang, DuyguCeylan, ErsinYumer, and Yasutaka Furukawa. "Planenet: Piece-wise planar reconstruction from a single rgb image." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2579-2588. 2018.

[30] Wang, Yuxiao, Yaochen Li, Ming Zeng, Zikun Dong, Jian Yuan, and Ziwei Wang. "Bottom-up Estimation of Geometric Layout for Indoor Images." In *2019 IEEE International Conference on Unmanned Systems (ICUS)*, pp. 848-853. IEEE, 2019.

[31] Kruzhilov, Ivan, Mikhail Romanov, and Anton Konushin. "Double refinement network for room layout estimation." (2019).

[32] Nie, Yinyu, ShihuiGuo, Jian Chang, Xiaoguang Han, Jiahui Huang, Shi-Min Hu, and Jian Jun Zhang. "Shallow2Deep: Indoor scene modeling by single image understanding." *Pattern Recognition* 103 (2020): 107271.

[33] Purkait, Pulak, Christopher Zach, and Ian Reid. "SG-VAE: Scene Grammar Variational Autoencoder to generate new indoor scenes." In *European Conference on Computer Vision*, pp. 155-171. Springer, Cham, 2020.

[34] Pintore, Giovanni, Marco Agus, and Enrico Gobbetti. "AtlantaNet: Inferring the 3D Indoor Layout from a Single 360^0 Image Beyond the Manhattan World Assumption." In *European Conference on Computer Vision*, pp. 432-448. Springer, Cham, 2020.

[35] Avetisyan, Armen, Tatiana Khanova, Christopher Choy, Denver Dash, Angela Dai, and Matthias Nießner. "Scenecad: Predicting object alignments and layouts in rgb-d scans." In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pp. 596-612. Springer International Publishing, 2020.

[36] Zhang, Weidong, Wei Zhang, and Yinda Zhang. "GeoLayout: Geometry driven room layout estimation based on depth maps of planes." In *European Conference on Computer Vision*, pp. 632-648. Springer, Cham, 2020.

[37] Ren, Liangliang, Yangyang Song, Jiwen Lu, and Jie Zhou. "Spatial Geometric Reasoning for Room Layout Estimation via Deep Reinforcement Learning." In *European Conference on Computer Vision*, pp. 550-565. Springer, Cham, 2020.

[38] Yan, Chenggang, Biyao Shao, Hao Zhao, RuixinNing, Yongdong Zhang, and Feng Xu. "3D room layout estimation from a single RGB image." *IEEE Transactions on Multimedia* 22, no. 11 (2020): 3014-3024.

[39] Sun, Cheng, Min Sun, and Hwann-Tzong Chen. "Hohonet: 360 indoor holistic understanding with latent horizontal features." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2573-2582 (2021).

[40] Nigam, MeherShashwat, AvinashPrabhu, AnuragSahu, Puru Gupta, TanviKarandikar, N. Sai Shankar, Ravi KiranSarvadevabhatla, and K. Madhava Krishna. "RackLay: Multi-Layer Layout Estimation for Warehouse Racks." *arXiv preprint arXiv:2103.09174* (2021).

[41] Zhao, Hao, Rene Ranftl, Yurong Chen, and HongbinZha. "Transferable End-to-end Room Layout Estimation via Implicit Encoding." *arXiv preprint arXiv:2112.11340* (2021).

[42] Pintore, Giovanni, Eva Almansa, Marco Agus, and Enrico Gobbetti. "Deep3DLayout: 3D reconstruction of an indoor layout from a spherical panoramic image." *ACM Transactions on Graphics (TOG)* 40, no. 6 (2021): 1-12.

[43] Zioulis, Nikolaos, Federico Alvarez, DimitriosZarpalas, and PetrosDaras. "Single-shot cuboids: Geodesics-based end-to-end Manhattan aligned layout estimation from spherical panoramas." *Image and Vision Computing* 110 (2021): 104160.

[44] Dasgupta, Saumitro, Kuan Fang, Kevin Chen, and Silvio Savarese. "Delay: Robust spatial layout estimation for cluttered indoor scenes." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 616-624. 2016.

[45] Amin, Javaria, Muhammad Almas Anjum, Muhammad Sharif, SeifedineKadry, Yunyoung Nam, and ShuiHua Wang. "Convolutional Bi-LSTM Based Human Gait Recognition Using Video Sequences." *CMC-COMPUTERS MATERIALS & CONTINUA* 68, no. 2 (2021): 2693-2709.

[46] Naveen, P., and P. Sivakumar. "Adaptive morphological and bilateral filtering with ensemble convolutional neural network for pose-invariant face recognition." *Journal of Ambient Intelligence and Humanized Computing* 12, no. 11 (2021): 10023-10033.

[47] Zhang, Weidong, Wei Zhang, Kan Liu, and Jason Gu. "Learning to predict high-quality edge maps for room layout estimation." *IEEE Transactions on Multimedia* 19, no. 5 (2016): 935-943.

[48] Hedau, Varsha, Derek Hoiem, and David Forsyth. "Recovering the spatial layout of cluttered rooms." In *2009 IEEE 12th international conference on computer vision*, pp. 1849-1856. IEEE, 2009.

[49] Jia, Heming, XiaoxuPeng, and Chunbo Lang. "Remora optimization algorithm." *Expert Systems with Applications* 185 (2021): 115665.

[50] Zioulis, Nikolaos, Federico Alvarez, Dimitrios Zarpalas, and Petros Daras. "Monocular spherical depth estimation with explicitly connected weak layout cues." *ISPRS Journal of Photogrammetry and Remote Sensing, Elsevier* 183 (2022): 269-285

[51] Farin, Dirk, Wolfgang Effelsberg, and Peter HN de With. "Floor-plan reconstruction from panoramic images." In *Proceedings of the 15th ACM international conference on Multimedia*, pp. 823-826. 2007.