Structural Deep Metric Learning for Room Layout Estimation

Wenzhao Zheng[1,2,3], Jiwen Lu[1,2,3,⋆], and Jie Zhou[1,2,3,4]

[1] Department of Automation, Tsinghua University, China
[2] State Key Lab of Intelligent Technologies and Systems, China
[3] Beijing National Research Center for Information Science and Technology, China
[4] Tsinghua Shenzhen International Graduate School, Tsinghua University, China
zhengwz18@mails.tsinghua.edu.cn; lujiwen@tsinghua.edu.cn;
jzhou@tsinghua.edu.cn

Abstract. In this paper, we propose a structural deep metric learning (SDML) method for room layout estimation, which aims to recover the 3D spatial layout of a cluttered indoor scene from a monocular RGB image. Different from existing room layout estimation methods that solve a regression or per-pixel classification problem, we formulate the room layout estimation problem from a metric learning perspective where we explicitly model the structural relations across different images. We propose to learn a latent embedding space where the Euclidean distance can characterize the actual structural difference between the layouts of two rooms. We then minimize the discrepancy between an image and its ground-truth layout in the learned embedding space. We employ a metric model and a layout encoder to map the RGB images and the ground-truth layouts to the embedding space, respectively, and a layout decoder to map the embeddings to the corresponding layouts, where the whole framework is trained in an end-to-end manner. We perform experiments on the widely used Hedau and LSUN datasets and achieve state-of-the-art performance.

Keywords: Deep Metric Learning · Room Layout Estimation · Structured Prediction

1 Introduction

Room layout estimation has attracted great attention in recent years, since it serves as a basic step to provide strong priors for a variety of applications such as indoor navigation [1,30,55], augmented reality [26,27,46], and scene understanding [8, 9, 13]. The goal of room layout estimation is to find a projection of a 3D box onto the image which best fits the actual layout of the scene, as described in Fig. 1. A major challenge of this task is the presence of clutter, where different kinds of furniture like beds, sofas, and tables may totally or partially occlude the boundary between two surfaces of the box, making it difficult to recover the underlying layout of the room.

⋆ Corresponding author

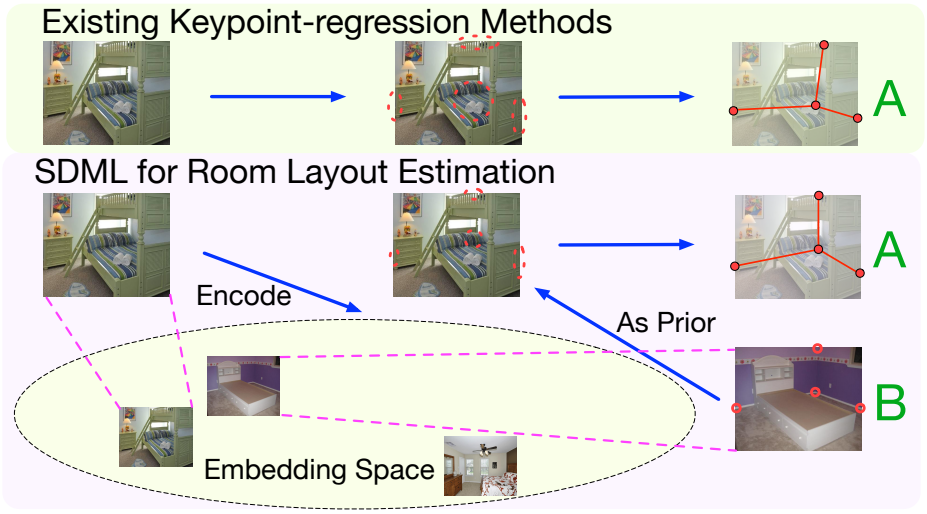


Fig. 1. Comparison of the proposed SDML method with existing methods for room layout estimation which simply regress the locations of keypoints (e.g. point of intersection among the left wall, the front wall, and the ceiling). The keypoints are often occluded by clutter like beds or sofas, making it hard to directly determine the exact locations of keypoints. Instead, we propose to learn a latent embedding space to explicitly model the structural relations across different images and use the globally similar images as prior to better infer the precise keypoints of the query image.

A main characteristic of room layout estimation is that the output is of a structured form, as indoor scenes typically satisfy the "Manhattan world assumption" [2]. Conventional methods on room layout estimation represent the room layout by a set of parameters and solve a structured prediction problem [16, 24, 36]. With the success of deep learning, recent works begin to employ deep convolutional neural networks (CNNs) [15,20,37,41] to extract features from images. The first category of works obtain a segmentation map for each image and then perform a search process for the best legitimate layout [3, 29, 49, 51]. The second category of works define a set of keypoints to describe the room layout and regress the locations of those keypoints [17, 23]. However, all these previous works fail to consider the structural correlations among different images and cannot well capture the global features of the layouts. On the other hand, being able to globally consider the whole indoor scene is important for room layout estimation due to the occlusion problem caused by clutter. For instance, as demonstrated by scene A in Fig. 1, it is difficult to directly find the point of intersection of the floor, the left wall, and the right wall, yet it is simpler to say it is similar to scene B in terms of their layouts. Using the predicted layout of scene B as prior, it is easier to determine the keypoints and layout of scene A.

Motivated by the above example, in this work, we formulate the room layout estimation problem from a metric learning perspective and propose a structural

deep metric learning (SDML) method which can be trained in an end-to-end manner. We explicitly model the structural relations across images by learning a mapping from the image space to a latent embedding space where the Euclidean distance can reflect the similarity of the underlying layouts of two images, i.e., a smaller distance between two images with more similar layouts and vice versa. We further propose a dense structural loss to enforce such a continuous constraint, which generalizes beyond previous triplet-based loss and enjoys the advantage of efficient sampling and faster convergence. We propose a layout antoencoder with a layout encoder to obtain a representation for each layout in the embedding space and a layout decoder to generate layouts from the embedding space. We then minimize the difference of the embeddings of an image and its corresponding ground-truth layout as well as the reconstruction cost between the original layout and the reconstructed layout. In the test phase, we simply connect the metric model and the layout decoder to obtain an estimated layout for an image. Extensive experiments on the widely used Hedau [16] and LSUN [50] datasets demonstrate the effectiveness of the proposed approach.

2 Related Work

Room Layout Estimation: The problem of room layout estimation was first formally introduced by Hedau *et al.* [16] and has attracted constant attention since then. Most conventional methods used structured prediction learning algorithms (e.g., structured SVMs [42]) for room layout estimation [16,24,35,36,43]. For example, Hedau *et al.* [16] proposed to iteratively localize visible objects and refit the box with structured SVMs in order to be more robust to clutter. Lee *et al.* [24] generated room hypotheses from line segments [25] and eliminated invalid hypotheses by volumetric reasoning before ranking them. Wang *et al.* [43] introduced latent variables to implicitly describe clutters and proposed to parameterize a layout by four factors inferred using structured SVMs.

Recent methods took advantage of deep networks and employed fully convolutional networks (FCNs) [28] to extract features, improving the performance of conventional methods dramatically. Mallya *et al.* [29] and Ren *et al.* [32] extracted per-pixel feature maps using FCNs and then ranked layout proposals based on them. Dasgupta *et al.* [3] and Zhao *et al.* [51] proposed to perform inference through optimization instead of proposal ranking, but they still require a two-step procedure to obtain the layout. Lee *et al.* [23] proposed an end-to-end framework for room layout estimation by simultaneously predicting the room type of a scene and regressing a set of pre-defined keypoints. However, existing methods lack a global understanding of the scene which is important to infer the underlying layout, especially for a cluttered room. Differently, the proposed SDML method explicitly models the structural relations among images which provide strong priors for more exact and robust room layout inference.

Metric Learning: Metric learning aims to learn a good distance function to measure the similarities between images with a common objective of minimizing intra-class distances and maximizing inter-class distances. Conventional metric

learning methods [4,11,44] usually learn a Mahalanobis distance as the distance metric. Recently a variety of deep metric learning methods [7,34,38,39,54] have been proposed and demonstrated promising results. They usually employ deep CNNs to compute an embedding for each image and then adopt the Euclidean distance between embeddings to model their similarities, where the loss function is carefully designed to enforce a certain relational constraint within a structured tuple. For example, Schroff *et al.* [34] proposed to separate the distances between the positive pair and the negative pair in a triplet by a fixed margin. Sohn [38] considered an (N+1)-tuple each time and required the metric to recognize one positive from $N-1$ negatives. Kim *et al.* [19] extended the triplet loss and required the distance ratio in the continuous label space to be preserved in the learned embedding space.

A straightforward application of metric learning is image retrieval as we only need to find the nearest neighbors of the query image under the learned metric. We can easily extend it to the classification problem by utilizing a K-nearest neighbor classifier [10]. However, it is still unclear how to use deep metric learning for classifying with structured output labels. We move a step forward and propose a SDML method for end-to-end structured room layout estimation.

3 Proposed Approach

In this section, we first present our method for structural layout distance learning and then describe the design of the layout autoencoder. Lastly, we detail the proposed SDML approach for relational room layout inference.

3.1 Structural Layout Distance Learning

Room layout estimation aims to obtain a boxy representation of the underlying layout of an indoor scene. Formally, let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$ be a set of indoor scene images and $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_N]$ be their corresponding ground-truth underlying layouts. Our objective is to assign a label (left wall, front wall, right wall, ceiling, and floor) to each pixel of the image indicating the surface it belongs to. For an image $\mathbf{x}$ of size $[h, w, 3]$, we estimate its layout on the resolution of $[\frac{h}{8}, \frac{w}{8}]$ following the protocol in previous work [23].

Existing room layout estimation methods fail to consider the structural relations across different images, which are actually of great value to obtain the estimated room layout, especially for a cluttered scene. Motivated by this, we propose to explore the correlations among different scenes by learning an embedding space where the Euclidean distance can reflect the actual structural distance between layouts.

To achieve this, we employ a CNN network to construct the mapping $\mathbf{f}(\mathbf{x}; \theta_f) = \mathbf{y}$ from the image space to the latent embedding space, where θ_f is the parameters of the metric network. We first extract a tensor $\mathbf{M} \in \mathbb{R}^{\frac{h}{8} \times \frac{w}{8} \times 2048}$ from the last convolutional layer of the CNN. We then use a 1×1 convolution to obtain

a feature map and flatten it to an n-dimension embedding that globally represents the layout. The learned distance is then defined as the Euclidean distance between the corresponding embeddings of two images:

$$D(\mathbf{x}_i, \mathbf{x}_j; \theta_f) = ||\mathbf{f}(\mathbf{x}_i; \theta_f) - \mathbf{f}(\mathbf{x}_j; \theta_f)||_2, \tag{1}$$

where $||\cdot||_2$ denotes the L2-norm.

We use the pixelwise surface label difference between two layouts as the layout distance [16], where we first employ the Hungarian algorithm [21] to find the matching surfaces. We then learn the distance metric to approximate the layout distance $d(\mathbf{z}_i, \mathbf{z}_j)$ with the Euclidean distance $D(\mathbf{x}_i, \mathbf{x}_j)$ so that images with similar room layouts are clustered together. The objective function of the metric learning problem can be formulated as:

$$L(\mathbf{x}_i, \mathbf{x}_j) = (D(\mathbf{x}_i, \mathbf{x}_j) - d(\mathbf{z}_i, \mathbf{z}_j))^2, \tag{2}$$

where $\mathbf{z}_i$ and $\mathbf{z}_j$ are the underlying room layouts of $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively.

Directly imposing such a constraint lacks flexibility and leads to inferior performance, and Kim *et al.* [19] instead minimizes the difference of log distance ratios in the two spaces based on triplets:

$$L(\mathbf{x}_a, \mathbf{x}_i, \mathbf{x}_j) = (\log \frac{D(\mathbf{x}_a, \mathbf{x}_i)}{d(\mathbf{z}_a, \mathbf{z}_i)} - \log \frac{D(\mathbf{x}_a, \mathbf{x}_j)}{d(\mathbf{z}_a, \mathbf{z}_j)})^2. \tag{3}$$

The goal of the structural layout distance learning is to obtain a metric to accurately represent the structural distance between layouts. Exploiting more relations imposes tighter constraints on the metric and is expected to perform better. Kim *et al.* [19] shows that densely sampling more triplets in one minibatch improves the performance. However, there exist $O(b^3)$ triplets that can be sampled from one minibatch where b is the batch size. Naively sampling all triplets and directly applying (3) will greatly increase the time complexity. To move a step forward, we propose a dense structural loss which includes not only all the triplets but also all the quadruplets in the minibatch:

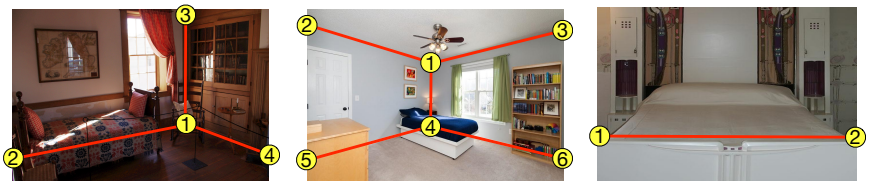
$$L_{dense} = \frac{1}{2} \sum_{\{i,j\} \neq \{k,l\}} (\log \frac{D(\mathbf{x}_i, \mathbf{x}_j)}{d(\mathbf{z}_i, \mathbf{z}_j)} - \log \frac{D(\mathbf{x}_k, \mathbf{x}_l)}{d(\mathbf{z}_k, \mathbf{z}_l)})^2, \tag{4}$$

where $i, j, k, l \in \mathbf{B}$. Note that the summands in (4) contain all the triplets when $i = k$. The proposed loss actually exploits all the triplet-wise relations and can be seen as a generalization of (3).

Still, directly computing (4) is computationally infeasible. Using Lagrange's identity [45], we can rewrite (4) as:

$$L_{dense} = \frac{b(b-1)}{2} \sum_{i<j \in \mathbf{B}} (\log \frac{D(\mathbf{x}_i, \mathbf{x}_j)}{d(\mathbf{z}_i, \mathbf{z}_j)})^2 - (\sum_{i<j \in \mathbf{B}} \log \frac{D(\mathbf{x}_i, \mathbf{x}_j)}{d(\mathbf{z}_i, \mathbf{z}_j)})^2, \tag{5}$$

where $\mathbf{B}$ denotes a minibatch of indices of training images and b is the batch size. We can efficiently compute the pairwise squared distance matrix $\mathbf{D}^2$ by



(a) Type 4 (b) Type 5 (c) Type 8

Fig. 2. Examples of the keypoint-based parameterization of room layouts. The LSUN dataset [50] defines 11 room types with a total of 48 keypoints, where each room type may contain different numbers of keypoints. We can generate a boxy layout by linking the keypoints with straight lines based on a predefined rule specific to each room type.

matrix operations $\mathbf{D}^2 = \widetilde{\mathbf{y}}\mathbf{1}^{\mathrm{T}} + \mathbf{1}^{\mathrm{T}}\widetilde{\mathbf{y}} - 2\mathbf{Y}\mathbf{Y}^{\mathrm{T}}$, where $D_{ij} = D(\mathbf{x}_i, \mathbf{x}_j)$, $\widetilde{\mathbf{y}} = [\{||\mathbf{y}_i||_2^2\}_{i\in\mathbf{B}}]^{\mathrm{T}} \in \mathbb{R}^{b\times 1}$, $\mathbf{Y} = [\{\mathbf{y}_i\}_{i\in\mathbf{B}}] \in \mathbb{R}^{b\times m}$, and m is the embedding size.

We see that the proposed dense structural loss (5) takes $O(b^2)$ time and can take full advantage of the minibatch. It explicitly constrains each pair to have the same ratio of distances with every other pair in one minibatch, which generalizes (3) and can exploit more information without substantially increasing the computing overhead.

3.2 Layout Autoencoder

Having obtained the latent embedding space, we can effectively measure the structural distance between two scenes. We minimize the discrepancies between the images and ground-truth labels by calculating L2 loss in this space. To achieve this, we propose to learn a layout autoencoder composed of a layout encoder $\mathbf{g}(\mathbf{z};\theta_g)$ to map a layout $\mathbf{z}$ to the embedding space and a decoder $\mathbf{h}(\mathbf{y};\theta_h)$ to map an embedding $\mathbf{y}$ back to the layout space.

We use the keypoint-based parameterization to describe a layout as specified in the LSUN dataset [50]. See Fig. 2 for a demonstration. They define 11 types of layouts with a total of 48 keypoints which include most pictures taken with standard cameras in a cuboid room. Each room type contains a sequence of different keypoints defined in a specific order. Connecting these keypoints following predefined rules will generate the final layout. The location of each keypoint can be expressed by a 2D Gaussian distribution heatmap centered at the keypoint, which is a more effective form as the input and output of CNNs [23]. Therefore, we can represent all of the 48 keypoints by a 3D tensor $\mathbf{K} \in \mathbb{R}^{\frac{h}{8}\times\frac{w}{8}\times 48}$ called keypoint tensor with each channel as the corresponding keypoint heatmap. A room layout is then a combination of room type and the keypoint tensor $\mathbf{z} = \{l, \mathbf{K}\}$, where $l \in \{0, 1, \cdots, 10\}$.

The layout encoder $\mathbf{g}$ is a two-layer convolutional network followed by a 1×1 convolutional layer, which shares the parameters with that in the metric model. The encoder then flattens it to obtain the layout embedding. The input of the encoder is a ground-truth keypoint tensor $\mathbf{K}^g$, where channels corresponding to keypoints that do not appear in the ground-truth room type are set to zeros.

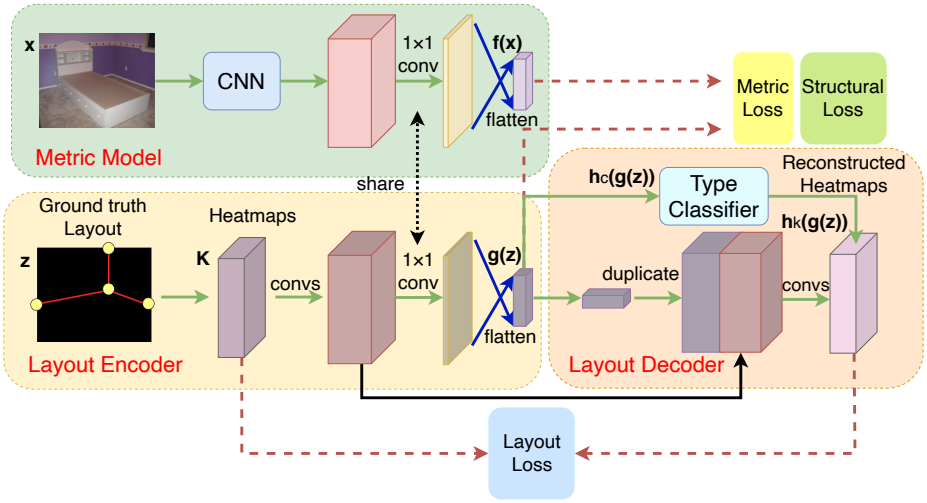


Fig. 3. An illustration of the proposed SDML method for room layout estimation. The metric model employs a CNN backbone network and a 1×1 convolutional layer to obtain a feature map of the query image and flattens it to obtain the embedding. The layout encoder first represents a ground-truth layout by the keypoint tensor and then uses two convolutional layers and one 1×1 convolutional layer to obtain the feature map. The 1×1 convolutional layer shares parameters with that in the metric model and we similarly flatten it to obtain the layout embedding. The layout decoder duplicates the layout embedding and concatenates them with the tensor from the encoder to aggregate global and local information. We then employ two convolutional layers to reconstruct the keypoint tensor and use a softmax classifier to select correct heatmaps.

The layout decoder $\mathbf{h}$ is composed of two parts: the type classifier $\mathbf{h}_c$ and the keypoint estimator $\mathbf{h}_k$. The type classifier $\mathbf{h}_c$ is a fully connected layer which takes as input an embedding $\mathbf{y}$ and outputs an 11-dimension vector $\mathbf{h}_c(\mathbf{y})$ representing the predicted room type possibilities. The keypoint estimator $\mathbf{h}_k$ first duplicates the embedding to obtain a tensor of size $\mathbb{R}^{\frac{h}{8} \times \frac{w}{8} \times d}$ and concatenates them to the feature maps outputted by the CNN in the layout encoder to aggregate both global and local information. We then use a two-layer convolutional network to obtain the estimated keypoint tensor $\mathbf{h}_k(\mathbf{y})$.

We use the reconstruction cost between the original and reconstructed ground-truth label as the objective to train the layout autoencoder. Similar to RoomNet [23], we measure the discrepancy of two keypoint tensors by the L2 loss of only the channels corresponding to keypoints that appear in the *original* room type $l(\mathbf{z})$. We use the softmax loss to train the type classifier to reconstruct the original room type. The layout reconstruction loss is formulated as:

$$L_{layout} = \sum_{i=0}^{47} I_{l(\mathbf{z})}(i) ||\mathbf{h}_k(\mathbf{g}(\mathbf{z}))_i - \mathbf{K}(\mathbf{z})_i||_F + \lambda_s L_{softmax}(\mathbf{h}_c(\mathbf{g}(\mathbf{z})), l(\mathbf{z})), \quad (6)$$

where $l(\mathbf{z})$ denotes the type of layout $\mathbf{z}$, $I_{l(\mathbf{z})}$ is an indicator function which equals 1 if the ith keypoint appears in the room type $l(\mathbf{z})$, $||\cdot||_F$ denotes the Frobenius norm, $\mathbf{K}(\mathbf{z})_i$ denotes the ith feature map of the keypoint tensor of layout $\mathbf{z}$, λ_s is a pre-defined parameter, and $L_{softmax}$ indicates the softmax loss function.

3.3 Relational Room Layout Inference

Keypoints are often occluded by clutter like beds and sofas, making it difficult to directly recover the room layout of an indoor scene. We think it is simpler to estimate the similarities of layouts of different scenes, and utilizing layouts of other images as prior will help determine the exact layout of the query image.

However, existing methods on room layout estimation fail to correctly model the relations between layouts and thus implicitly use a biased prior. For example, RoomNet [23] treats room layout estimation as a regression problem and directly uses the L2 distance to measure the discrepancy between layouts, which does not consider the structural nature of room layouts and is not necessarily consistent with the actually proper metric like the pixel error [16]. In this case, nearest neighbors may not have the most similar underlying layouts and the correlations cannot be used as a good prior. The core problem is that such a structural metric is usually non-differential. We address this by learning an embedding space where the L2 distance reflects the structural distance. The correlation between images now is an unbiased prior and the nearest samples in the embedding space can be correctly decoded to similar layouts.

To achieve this, we first review an image $\mathbf{x}$ from a global perspective and consider its structural relation with other layouts to obtain the corresponding embedding $\mathbf{f}(\mathbf{x})$ in the latent embedding space. Then we employ the layout decoder $\mathbf{h}$ to map the embedding to the final estimated layout $\mathbf{h}(\mathbf{f}(\mathbf{x}))$. The overall framework of the proposed SDML is illustrated in Fig. 3.

To constrain the metric model $\mathbf{f}$ to map an image to the embedding corresponding to its ground-truth layout, we minimize the discrepancy between the predicted layout and the ground-truth layout using a simple L2 loss in the embedding space. The structural prediction loss is formulated as:

$$L_{str} = ||\mathbf{f}(\mathbf{x}) - \mathbf{g}(\mathbf{z})||^2. \tag{7}$$

We can also regard (7) as a distribution matching loss [18] to minimize the shift between the distributions of the image embeddings and layout embeddings.

Since we obtain an estimated layout by reconstruction from the embedding space, the embedding of an image should be as close to the corresponding layout embedding as possible. A shift from the correct layout embedding will cause an error to the estimated layout and thus harms the accuracy of layout estimation. Therefore, we tailor the proposed dense structural loss to further strengthen the connections between image embeddings and layout embeddings. We propose to constrain the log distance ratio between images and labels to be equal in the

embedding space and the layout space:

$$L_{metric} = b(b-1) \sum_{i \neq j \in \mathbf{B}} (\log \frac{D(\mathbf{x}_i, \mathbf{z}_j)}{d(\mathbf{z}_i, \mathbf{z}_j)})^2 - (\sum_{i \neq j \in \mathbf{B}} \log \frac{D(\mathbf{x}_i, \mathbf{z}_j)}{d(\mathbf{z}_i, \mathbf{z}_j)})^2, \quad (8)$$

where $D(\mathbf{x}_i, \mathbf{z}_j) = ||\mathbf{f}(\mathbf{x}_i) - \mathbf{g}(\mathbf{z}_j)||_2$ is the Euclidean distance between the mapped image $\mathbf{f}(\mathbf{x}_i)$ and the encoded layout $\mathbf{g}(\mathbf{z}_j)$ in the embedding space. Note that (8) has the same computation complexity as the original dense structural loss since we can similarly compute an asymmetric distance matrix. Moreover, the proposed metric loss integrates the constraints on the relations of image embeddings and layout embeddings and thus is more adaptive to the proposed SDML method. We only use (8) to update the metric function $\mathbf{f}$ in order to make the training more stable.

The entire framework of the proposed SDML approach is composed of three parts, a metric model $\mathbf{f}$ to map a scene to its embedding, a layout encoder $\mathbf{g}$ to map a layout to its embedding and a layout decoder $\mathbf{h}$ to map an embedding to its corresponding layout. Our SDML method can be trained end-to-end where we simultaneously learn the three parts $\mathbf{f}$, $\mathbf{g}$, and $\mathbf{h}$. The overall objective of the proposed approach can be formulated as:

$$\min_{\theta_f, \theta_g, \theta_h} L = \min_{\theta_f} L_{metric} + \lambda_1 \min_{\theta_f, \theta_g} L_{str} + \lambda_2 \min_{\theta_g, \theta_h} L_{layout}, \quad (9)$$

where λ_1 and λ_2 are parameters to balance the contributions of different losses.

L_{metric} constructs an embedding space where the Euclidean distance can approximate the actual layout distance. L_{str} minimizes the difference between the predicted layout and the ground-truth layout by the L2 loss in the embedding space. L_{layout} learns a layout autoencoder to connect corresponding points in the embedding space and structured layout space.

In the test phase, we can directly estimate the room layout of an image $\mathbf{x}$ by $\mathbf{h}(\mathbf{f}(\mathbf{x}))$. Having obtained the keypoint tensor, we can find the corresponding keypoint heatmaps based on the estimated room type and extract the locations of each keypoint by an argmax operation on the heatmap. We then link the keypoints following the pre-defined protocol to obtain the final boxy representation of the estimated room layout.

4 Experiments

In this section, we conducted a variety of experiments to evaluate the performance of the proposed SDML for room layout estimation. Our experiments demonstrate the superiority of the proposed dense structural loss on the room layout retrieval task and analyze the origins of performance improvement through the ablation study. For room layout estimation, we employ the pixel error (PE) and the keypoint error (KPE) as performance metrics. The pixel error measures the average classification error of the predicted surface label of each pixel. The keypoint error computes the average Euclidean distance between the estimated keypoint locations and the ground-truth keypoint locations normalized by the image diagonal length.

4.1 Datasets

We followed existing methods [3,23,29] and evaluated our method on the widely used Hedau [16] and Large-scale Scene Understanding Challenge (LSUN) [50] room layout benchmark datasets. The Hedau dataset [16] consists of 367 images collected from the Internet using LabelMe [33], including 209 training images, 53 validation images, and 105 test images. The LSUN dataset [50] consists of 5,394 images sampled from the SUN database [47], including 4000 training images, 394 validation images, and 1000 test images.

For all the experiments, we trained our model from scratch on the training split of the LSUN dataset. In the training phase, we resized each image to a scale of 320×320 and estimated its room layout at a scale of 40×40. In the test phase, we obtained the estimated layouts at the scale of 40×40, but rescale it to the original image scale.

4.2 Implementation Details

We conducted all the experiments using the PyTorch package. We instantiated the metric function $\mathbf{f}$ by a ResNet-50 [15] backbone model with the dilated network strategy [48] similar to that used by PSPNet [52], which takes as input an 320×320 image and outputs a $40 \times 40 \times 2048$ tensor. The following 1×1 convolutional layer further maps it to a feature map of size $40 \times 40 \times 1$, rendering 1600 as the dimension of the embedding space. The layout encoder takes as input a keypoint tensor of size $40 \times 40 \times 48$ and employs two convolutional layers with 512 1×1 kernels and 2048 3×3 kernels to obtain a $40 \times 40 \times 2048$ tensor. The following 1×1 convolutional layer shares parameters with that in the metric model. The layout decoder is composed of two convolutional layers with 512 3×3 kernels and 48 1×1 kernels. The classifier is a softmax layer with the input of a 1600-dimension embedding and the output of an 11-dimension vector indicating the estimated probabilities of the room type. We performed a random horizontal mirror of images during training for data augmentation. We fixed the batch size to 15 due to the limited physical memory of the GPU card. We used the Adam optimizer and set the learning rate to 10^{-4}. We set λ_1, λ_2, and λ_s to 5, 1, and 0.3, respectively, to balance the effect of different losses.

4.3 Results and Analysis

Evaluation of the dense structural loss: We first conducted an experiment on the task of room layout retrieval to verify the effectiveness of the proposed dense structural loss. The goal is to retrieve a set of images with similar underlying layouts given a query image. We measure the distance between layouts by the pixelwise surface label difference. We obtain the retrieved images by searching for the nearest neighbors in the learned latent embedding space, which requires the learned metric to accurately reflect the actual layout similarity. We adopted the mean label distance (smaller is better) and the modified version of the normalized discounted cumulative gain (nDCG) [19, 22] (larger is better) as the

Table 1. Room layout retrieval results (%) on the LSUN dataset compared with other methods. Red and bold numbers denote the best and second-best results, respectively.

Method	Mean label distance					nDCG				
Number of retrievals	1	2	4	8	16	1	2	4	8	16
Contrastive [14]	31.2	31.6	32.7	33.0	33.4	92.0	93.2	94.7	95.2	96.9
Triplet [34]	27.2	27.4	27.7	28.3	29.5	94.1	94.7	95.6	96.3	97.1
N-pair [38]	29.5	29.9	30.8	31.1	32.1	93.1	93.9	95.7	98.5	97.3
RoomNet [23]	27.9	28.2	29.7	30.9	32.5	93.3	93.8	94.9	96.0	96.5
Log-ratio (Random) [19]	25.2	26.4	27.1	28.7	30.5	95.3	96.0	96.9	97.4	97.8
Log-ratio (Dense) [19]	23.3	24.1	25.3	26.5	27.9	96.1	96.6	97.6	98.3	98.9
Dense structural	**20.7**	**21.5**	**22.9**	**24.1**	**25.3**	**97.2**	**97.5**	**98.6**	**98.8**	**99.1**
Dense structural (Layout)	**20.9**	**21.6**	**23.0**	**24.0**	**25.1**	**97.4**	**97.5**	**98.5**	**98.7**	**99.2**

Table 2. Performance of the proposed SDML method using different losses on the LSUN dataset.

Setting	PE (%)	KPE (%)
SDML without L_{metric}	9.93	6.42
SDML without L_{str}	15.36	10.23
SDML with L_{dense}	8.56	5.86
SDML	**6.95**	**5.29**

Table 3. Performance of the proposed SDML method under different model settings on the LSUN dataset.

Setting	PE (%)	KPE (%)
RoomNet [23]	10.46	6.95
RoomNet + L_{dense}	9.23	6.27
Keypoints regression	9.97	6.60
SDML	**6.95**	**5.29**

evaluation metrics. We employed 60 images as queries in the validation split of the LSUN dataset. We refer readers to previous work [19] for more details.

We compared our dense structural loss with several baseline methods including the contrastive loss [14], the triplet loss [34], the N-pair loss [38] and the state-of-the-art method log-ratio loss with dense sampling [19]. The first three methods aim to pull closer samples from the same class and push away samples from different classes. For each sample, we chose its 30 nearest neighbors as positive samples, and others as negative samples to perform training. We evaluated all the losses using the same metric model as described in Section 3.1. We also tested the framework of RoomNet [23] and used the 512-dimension vector in the room type prediction module as the embedding for room layout retrieval. For our method, Dense structural denotes using the proposed dense structural loss (5) and Dense structural (Layout) denotes using the proposed metric loss (8) with a layout encoder.

Table 1 shows the results of room layout retrieval on the LSUN dataset. We see that the proposed loss outperforms the other baseline methods by a large margin. In particular, our method performs better than the state-of-the-art log-ratio loss with dense sampling. This is because our loss exploits full information from the batch and imposes more structural constraints on the metric.

Table 4. Experimental results of the proposed SDML method compared with existing methods. Red and bold numbers denote the best and second-best results, respectively.

Method	Hedau dataset	LSUN dataset		Time (s/image)
	PE (%)	PE (%)	KPE (%)	
Hedau *et al.* [16]	21.20	24.23	15.48	-
Del Pero *et al.* [5]	16.30	-	-	720
Gupta *et al.* [12]	16.20	-	-	-
Zhao *et al.* [53]	14.50	-	-	-
Ramalingam *et al.* [31]	13.34	-	-	6
Schwing *et al.* [35]	12.8	-	-	0.15
Del Pero *et al.* [6]	12.7	-	-	900
Mallya *et al.* [29]	12.83	16.71	11.02	-
DeLay [3]	9.73	10.63	8.20	30
CFILE [32]	8.67	9.31	7.95	-
RoomNet [23]	12.19	10.46	6.95	0.052
RoomNet recurrent 3-iter [23]	8.36	9.86	6.30	0.17
LayoutNet [56]	9.69	-	-	**0.039**
Hirzer *et al.* [17]	7.44	7.79	5.84	0.086
Zhang *et al.* [49]	**7.36**	**6.58**	**5.17**	150.18
SDML	**7.21**	**6.95**	**5.29**	**0.017**

Ablation study: We first evaluated the effect of the three losses in (9). Table 2 shows the performance of the proposed SDML method using different losses. SDML without L_{metric} indicates training the SDML framework without explicitly modeling the structural relations across different scenes. The degraded performance verifies the advantage of considering the correlations among room layouts. SDML without L_{str} denotes training our framework without the structural loss, which constrains the metric model and the label encoder to map an image to the same embedding corresponding to its ground-truth label. It achieves inferior results since the label decoder might decode an inconsistent layout embedding leading to mistaken estimation. However, the reduction of the performance is not too large because L_{metric} still has an effect of decreasing the distribution shift. In addition, replacing the metric loss L_{metric} with the dense structural loss L_{dense} also decreases the performance of the proposed SDML method. This demonstrates L_{metric} is more adaptive with our method since it reinforces the connections between image embeddings and layout embeddings.

Moreover, we evaluated the proposed SDML method under different model settings, as shown in Table 3. We modified RoomNet [23] by further applying the dense structural loss L_{dense} to the 512-dimension vector in the room type classifier in RoomNet. We see that RoomNet + L_{dense} outperforms the original RoomNet. This is because the dense structural loss encourages the decoder to consider the structural relations between images which further constrain it to encode more global information relative to the underlying layout. We also tested the performance of our model when removing the layout decoder and simply

Fig. 4. Qualitative results of our SDML in comparison with RoomNet [23].

regress the positions of keypoints similar to RoomNet. We observe that the performance is only slightly better than RoomNet. This shows the performance improvement of our method mainly results from the relation modeling.

Quantitative results: We compared the proposed SDML framework with existing methods on room layout estimation, where we summarize the results on the Hedau dataset and LSUN dataset in Table 4. We see that our method is the fastest with comparable performance with Zhang *et al.* [49]. In particular, while Zhang *et al.* [49] achieves slightly better results on the LSUN dataset than the proposed framework, we want to emphasize that it is a two-step method which is orders of magnitude slower than our method. Note that our method was only trained on the training split of the LSUN dataset without using external data. Zhao *et al.* [51] achieved better results (5.29% pixel error and 3.84% keypoint error on the LSUN dataset), yet they exploited more information by additionally training a model on the SUBRGBD dataset [40] on a 37-class semantic segmentation task to better describe clutter.

The proposed SDML framework exploits the learned structural relations as global prior to infer the underlying room layout of a scene, leading to superior performance. Our method has the advantage of balanced performance and cost and has the potential to be further applied to other structural prediction tasks.

Qualitative results: Fig. 4 shows the visualization of the room layout estimation results of our method. We provide a comparison with the current state-of-the-art method RoomNet [23], which directly regresses the locations of keypoints. We observe that our method can more robustly estimate the locations of the keypoints even when they are occluded by clutter like beds or tables, which intuitively demonstrates the effectiveness of the proposed approach.

Fig. 5 demonstrates some ambiguous scenes where the predictions of the proposed SDML method and RoomNet both fail to match the ground-truth annotations. Still, we see that our method produces better estimations than RoomNet

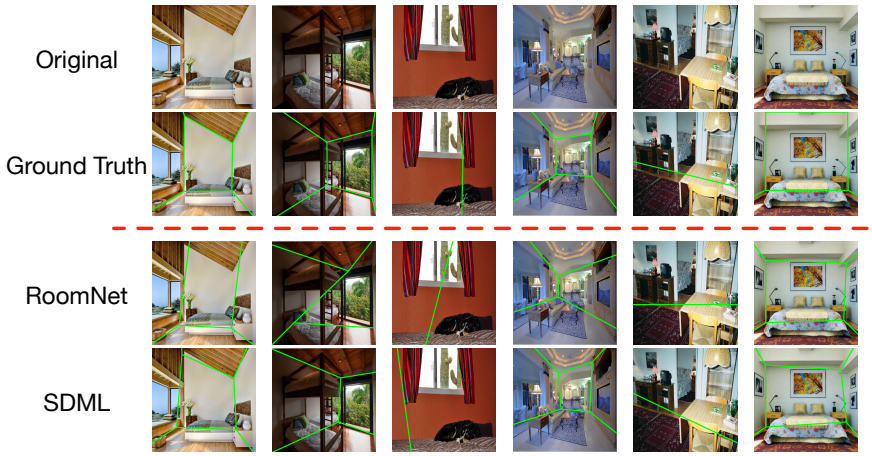


Fig. 5. Some ambiguous cases where SDML and RoomNet [23] both predict incorrectly.

since we explicitly model the structural relations among different scenes and use them as prior to assist the room layout estimation process.

5 Conclusion

In this paper, we have presented a structural deep metric learning framework (SDML) for room layout estimation, which formulates the problem from a metric learning perspective. We learn a latent embedding space to explicitly model the relations across different indoor scenes and utilize a layout autoencoder to connects the embedding space and the underlying layout in order to perform relational room layout inference. We have performed experimental evaluations on two widely used datasets which have verified the effectiveness of our method. In the future, it is interesting to further extend our method to the general structural prediction problem and apply SDML to more tasks such as human pose estimation and hand pose estimation.

Acknowledgements

The authors would like to thank Yangyang Song for his kind support and helpful discussions. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61822603, Grant U1813218, Grant U1713214, and Grant 61672306, in part by Beijing Natural Science Foundation under Grant No. L172051, in part by Beijing Academy of Artificial Intelligence (BAAI), in part by a grant from the Institute for Guo Qiang, Tsinghua University, in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564, and in part by Tsinghua University Initiative Scientific Research Program.

References

1. Boniardi, F., Valada, A., Mohan, R., Caselitz, T., Burgard, W.: Robot localization in floor plans using a room layout edge extraction network. In: IROS. pp. 5291–5297 (2019)
2. Coughlan, J.M., Yuille, A.L.: The manhattan world assumption: Regularities in scene statistics which enable bayesian inference. In: NIPS. pp. 845–851 (2001)
3. Dasgupta, S., Fang, K., Chen, K., Savarese, S.: Delay: Robust spatial layout estimation for cluttered indoor scenes. In: CVPR. pp. 616–624 (2016)
4. Davis, J.V., Kulis, B., Jain, P., Sra, S., Dhillon, I.S.: Information-theoretic metric learning. In: ICML. pp. 209–216 (2007)
5. Del Pero, L., Bowdish, J., Fried, D., Kermgard, B., Hartley, E., Barnard, K.: Bayesian geometric modeling of indoor scenes. In: CVPR. pp. 2719–2726 (2012)
6. Del Pero, L., Bowdish, J., Kermgard, B., Hartley, E., Barnard, K.: Understanding bayesian rooms using composite 3d object models. In: CVPR. pp. 153–160 (2013)
7. Duan, Y., Zheng, W., Lin, X., Lu, J., Zhou, J.: Deep adversarial metric learning. In: CVPR. pp. 2780–2789 (2018)
8. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: ICML. pp. 2650–2658 (2015)
9. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NIPS. pp. 2366–2374 (2014)
10. Fix, E., Hodges Jr, J.L.: Discriminatory analysis-nonparametric discrimination: consistency properties. Tech. rep., California Univ Berkeley (1951)
11. Globerson, A., Roweis, S.T.: Metric learning by collapsing classes. In: NIPS. pp. 451–458 (2006)
12. Gupta, A., Hebert, M., Kanade, T., Blei, D.M.: Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In: NIPS. pp. 1288–1296 (2010)
13. Gupta, S., Arbeláez, P., Girshick, R., Malik, J.: Aligning 3d models to rgb-d images of cluttered scenes. In: CVPR. pp. 4731–4740 (2015)
14. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: CVPR. pp. 1735–1742 (2006)
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
16. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: ICCV. pp. 1849–1856 (2009)
17. Hirzer, M., Roth, P.M., Lepetit, V.: Smart hypothesis generation for efficient and robust room layout estimation. In: WACV. pp. 2912–2920 (2020)
18. Huang, J., Gretton, A., Borgwardt, K., Schölkopf, B., Smola, A.J.: Correcting sample selection bias by unlabeled data. In: NIPS. pp. 601–608 (2007)
19. Kim, S., Seo, M., Laptev, I., Cho, M., Kwak, S.: Deep metric learning beyond binary supervision. In: CVPR. pp. 2288–2297 (2019)
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. pp. 1097–1105 (2012)
21. Kuhn, H.W.: The hungarian method for the assignment problem. Naval Research Logistics Quarterly **2**(1-2), 83–97 (1955)
22. Kwak, S., Cho, M., Laptev, I.: Thin-slicing for pose: Learning to understand pose without explicit pose estimation. In: CVPR. pp. 4938–4947 (2016)
23. Lee, C.Y., Badrinarayanan, V., Malisiewicz, T., Rabinovich, A.: Roomnet: End-to-end room layout estimation. In: ICCV. pp. 4865–4874 (2017)

24. Lee, D.C., Gupta, A., Hebert, M., Kanade, T., Blei, D.M.: Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In: NIPS. pp. 1288–1296 (2010)
25. Lee, D.C., Hebert, M., Kanade, T.: Geometric reasoning for single image structure recovery. In: CVPR. pp. 2136–2143 (2009)
26. Lin, C., Li, C., Furukawa, Y., Wang, W.: Floorplan priors for joint camera pose and room layout estimation. arXiv **abs/1812.06677** (2018)
27. Liu, C., Schwing, A.G., Kundu, K., Urtasun, R., Fidler, S.: Rent3d: Floor-plan priors for monocular layout estimation. In: CVPR. pp. 3413–3421 (2015)
28. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. pp. 3431–3440 (2015)
29. Mallya, A., Lazebnik, S.: Learning informative edge maps for indoor scene layout prediction. In: ICCV. pp. 936–944 (2015)
30. Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A.J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., et al.: Learning to navigate in complex environments. In: ICLR (2017)
31. Ramalingam, S., Pillai, J.K., Jain, A., Taguchi, Y.: Manhattan junction catalogue for spatial reasoning of indoor scenes. In: CVPR. pp. 3065–3072 (2013)
32. Ren, Y., Li, S., Chen, C., Kuo, C.C.J.: A coarse-to-fine indoor layout estimation (cfile) method. In: ACCV. pp. 36–51 (2016)
33. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. IJCV **77**(1-3), 157–173 (2008)
34. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: CVPR. pp. 815–823 (2015)
35. Schwing, A.G., Hazan, T., Pollefeys, M., Urtasun, R.: Efficient structured prediction for 3d indoor scene understanding. In: CVPR. pp. 2815–2822 (2012)
36. Schwing, A.G., Urtasun, R.: Efficient exact inference for 3d indoor scene understanding. In: ECCV. pp. 299–313 (2012)
37. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
38. Sohn, K.: Improved deep metric learning with multi-class n-pair loss objective. In: NIPS. pp. 1857–1865 (2016)
39. Song, H.O., Xiang, Y., Jegelka, S., Savarese, S.: Deep metric learning via lifted structured feature embedding. In: CVPR. pp. 4004–4012 (2016)
40. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. In: CVPR. pp. 567–576 (2015)
41. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR. pp. 1–9 (2015)
42. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables. JMLR **6**(Sep), 1453–1484 (2005)
43. Wang, H., Gould, S., Koller, D.: Discriminative learning with latent variables for cluttered indoor scene understanding. In: ECCV, pp. 497–510 (2010)
44. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. JMLR **10**(2), 207–244 (2009)
45. Weisstein, E.W.: CRC concise encyclopedia of mathematics. Chapman and Hall/CRC (2002)
46. Xiao, J., Furukawa, Y.: Reconstructing the world's museums. IJCV **110**(3), 243–258 (2014)
47. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: CVPR. pp. 3485–3492 (2010)

48. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: ICLR (2016)
49. Zhang, W., Zhang, W., Gu, J.: Edge-semantic learning strategy for layout estimation in indoor environment. TCYB (2019)
50. Zhang, Y., Yu, F., Song, S., Xu, P., Seff, A., Xiao, J.: Large-scale scene understanding challenge: Room layout estimation. In: CVPR Workshop (2015)
51. Zhao, H., Lu, M., Yao, A., Guo, Y., Chen, Y., Zhang, L.: Physics inspired optimization on semantic transfer features: An alternative method for room layout estimation. In: CVPR. pp. 10–18 (2017)
52. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR. pp. 2881–2890 (2017)
53. Zhao, Y., Zhu, S.C.: Scene parsing by integrating function, geometry and appearance models. In: CVPR. pp. 3119–3126 (2013)
54. Zheng, W., Chen, Z., Lu, J., Zhou, J.: Hardness-aware deep metric learning. In: CVPR. pp. 72–81 (2019)
55. Zhu, F., Zhu, L., Yang, Y.: Sim-real joint reinforcement transfer for 3d indoor navigation. In: CVPR. pp. 11388–11397 (2019)
56. Zou, C., Colburn, A., Shan, Q., Hoiem, D.: Layoutnet: Reconstructing the 3d room layout from a single rgb image. In: CVPR. pp. 2051–2059 (2018)