LaLaLoc: Latent Layout Localisation in Dynamic, Unvisited Environments

Henry Howard-Jenkins
Active Vision Laboratory
University of Oxford
henryhj@robots.ox.ac.uk

Jose-Raul Ruiz-Sarmiento
Machine Perception and
Intelligent Robotics Group
University of Málaga
jotaraul@uma.es

Victor Adrian Prisacariu
Active Vision Laboratory
University of Oxford
victor@robots.ox.ac.uk

Abstract

We present LaLaLoc to localise in environments without the need for prior visitation, and in a manner that is robust to large changes in scene appearance, such as a full rearrangement of furniture. Specifically, LaLaLoc performs localisation through latent representations of room layout. LaLaLoc learns a rich embedding space shared between RGB panoramas and layouts inferred from a known floor plan that encodes the structural similarity between locations. Further, LaLaLoc introduces direct, cross-modal pose optimisation in its latent space. Thus, LaLaLoc enables fine-grained pose estimation in a scene without the need for prior visitation, as well as being robust to dynamics, such as a change in furniture configuration. We show that in a domestic environment LaLaLoc is able to accurately localise a single RGB panorama image to within 8.3cm, given only a floor plan as a prior.

1. Introduction

Camera relocalisation is a fundamental problem in computer vision. Image-based relocalisation represents the goal of estimating the camera pose of an unseen image, given some prior knowledge about the surrounding environment. In this paper, we tackle the task of localisation in an environment that has not been previously visited, and one in which there may be considerable scene dynamics – an area where significant scope for improvement has been identified [37].

To address this, we propose to localise with respect to a known floor plan and the layout visible at a location within the scene. Floor plan-based localisation is particularly suited for the long-term localisation setting as, while objects and furniture may have moved, items represented in a structural floor plan, such as walls, floors and ceilings, will remain static. Therefore, it enables localisation over a long period of time without requiring continual re-training or re-mapping. In addition, in this formulation we only need the floor plan as prior, thus removing the need for previous visitation of the target environment, *i.e.* without a training trajectory of images.

We present LaLaLoc, which performs floor plan-based localisation through latent representations of room layout. This layout latent space is cross-modal, shared between layouts inferred from the floor plan and the RGB panoramas queried at inference time. More specifically LaLaLoc performs localisation in two stages, depicted in Figure 1. The first stage provides a coarse estimate of pose through cross-modal retrieval. For the second stage, we propose a cross-modal direct optimisation of pose through differentiable rendering.

Differentiable rendering has been shown to be effective for object pose estimation [23, 12]. But these works typically rely on like-for-like rendering losses, such as the pixelwise error between the rendered and target images. However, since LaLaLoc operates across multiple modes of data between query and prior, the prediction of a common data mode would be required for the comparison losses. Instead, we propose to optimise for pose directly in the layout latent space. Through this formulation, LaLaLoc is able to accurately align the floor plan to a cluttered RGB panorama without ever explicitly predicting its layout.

The contributions of this paper can be summarised as:

- We propose LaLaLoc, a highly accurate localisation method that is robust to scene dynamics such as the configuration of furniture, and able to localise in a new scene without prior visitation.

- We introduce direct pose optimisation in the latent space. This allows for cross-modal pose optimisation, without the need for a decoder to traverse data modes for the computation of the matching cost.

- Through experimental evaluation, we demonstrate the accuracy of LaLaLoc as well as validate its formulation. This includes showing that the representation of room layout has significant influence on the efficacy of layout-based localisation and cannot be used interchangeably.

1. Cross-modal Retrieval
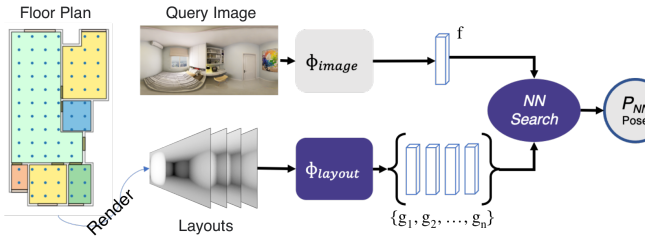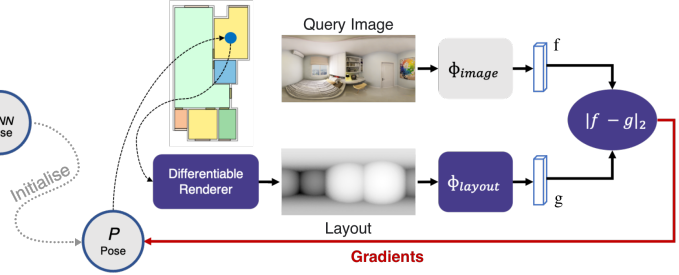
2. Latent Pose Optimisation

Figure 1. Overview of localisation using LaLaLoc. In the retrieval stage, the query image is mapped to the layout latent space by Φ_{image}. We then sample a grid of poses from a known floor plan, render their layouts, and compute their respective latent representation through Φ_{layout}. An initial pose estimate is found by a nearest neighbour search within this shared latent space. The nearest neighbour pose is then used as an initialisation for direct pose optimisation with our proposed latent pose optimisation. This is a gradient-based optimisation of pose conducted in the shared latent space, thus removing the need to decode into a common data mode. Our Vogel Disc resampling stage is ommitted from this viusalisation for clarity.

2. Related Work

A wide variety of methods have been produced to tackle the task of camera localisation. There are scene-specific methods, which require fine-tuning to each individual scene. Pose-regression methods [18, 17, 10] train a deep network for each scene to directly predict the camera pose from an input, but these methods are limited in accuracy [27]. Instead of regressing camera pose, scene-coordinate regression methods [30, 7, 8, 9, 35] densely predict 2D-3D correspondences between the query and the environment, which then allows for solving for pose via PnP. On the other hand, some groups of methods are able to generalise across scenes without need for retraining. Image retrieval methods [28, 1, 2, 13] estimate pose through by using the pose of the most similar image within an image database. 3D structure-based methods [21, 20, 26] instead establish 2D-3D between the query image and points in a SfM model. However, the scene-specific and generalisable methods discussed do have a commonality in that they require prior visitation in the same data modality of the region in which localisation is performed: scene-specific methods require this for re-training, while the generalisable methods require it for map/database building. By contrast, LaLaLoc performs localisation without prior visitation, instead leveraging only a known floor plan as a prior.

Particularly within the field of robotics, there have been a few proposed approaches to localise with respect to a floor plan. These methods operate by aggregating depth [40], detecting suitable features such as layout corners [14], or extracting layout edges [6, 38, 34] within query RGB and/or depth images to estimate and compare with the inferred layout at a location within the floor plan to estimate the observation likelihood. When depth information is available, scan-matching techniques [24] find a rigid alignment between the scene and the scan through alignment methods such as Iterative Closest Point (ICP) [3], General-ICP (GICP) [29], or Normal Distributions Transform (NDT) [4]. ICP-based scan matching techniques have successfully been used to perform localisation within a floor plan [5, 39]. These methods generally rely on a flow of information over multiple sequential measurements, typically alongside motion priors, such as from wheel or camera odometry, and are fed as hypothesis weightings into a Monte Carlo Localisation (MCL) [33] framework. In this paper, however, we approach the task of localisation from an instantaneous observation with no motion or time-coherency cues, and without assuming a good initialisation for pose. Furthermore, unlike the scan-matching methods, we do not require depth information, and in fact we remove the need for any explicit prediction of layout geometry at query time entirely, instead solely leveraging latent representations of layout.

Kim *et al*. [19] first proposed the learning of a latent space which captures room layout similarity. Specifically, this is applied to the task of image retrieval, where images are embedded to reflect their underlying room layout and for a given query image the goal is to return other images with similar layouts. Zheng *et al*. [42] later employed a layout embedding space to aid in the prediction of room layouts. However, both of these works focused on room layouts that follow a box approximation, where the layout takes the form of the inside of a convex cuboid. This leads to much reduced variability in the space of possible layouts: there are 11 types of room layout which can be seen in an image, of which [19] only considered 1. We, instead, consider general room layouts, without imposing any assumptions about their structure, leading to more variety in potential layout. However, it is the same concept of a layout latent space on which LaLaLoc's localisation is enabled.

3. Task and Definitions

We perform camera pose localisation within a floor plan provided as prior knowledge. Specifically, we localise the 2

DoF camera pose, P, to a point in a 2D floor plan. For rendering of the reference layouts, we assume that the camera and ceiling height are given, and produce a 3D floor plan $\mathcal{M}$ of the scene through extrusion of the 2D floor plan. $\mathcal{M}$ consists of only walls, floors and ceilings.

Throughout the paper, we deal with three main data types that may be captured or rendered at a position within the floor plan, P_p: I_p is the RGB panorama captured; L_p is a rendered depth image corresponding to $\mathcal{M}$ being projected into the image; C_p is a pointcloud formed by back-projecting L_p, in effect a sub-sampling of $\mathcal{M}$.

4. Latent Layouts

LaLaLoc consists of a network with two parallel branches to form a quasi-Siamese network. One branch, Φ_{image}, computes a feature descriptor for an RGB panorama image, while the other branch, Φ_{layout}, computes a descriptor from layout renderings. The embedding for a RGB panorama image should be identical to the embedding computed for a layout rendering at the same camera pose. This means that LaLaLoc is tasked with computing a singular latent representation of the room layout as visible at a particular location within a floor plan, independent of the sensor data used to compute it.

To learn such an embedding space, we take an approach analogous to knowledge distillation [15]. Knowledge distillation generally first trains a complex model as the teacher, which is then used to improve the convergence of a less complex model. Instead, we train a model on the easier task: Φ_{layout} performing layout-to-layout matching, and use it as a "teacher" model for a "student" performing a more complicated task: Φ_{image} learning to encapsulate the layout information present in an RGB image. This approach allows us to maximise the richness of our layout latent space by only considering learning the relationships between layouts, before training Φ_{image} to also map images to this space. However, this learning formulation in LaLaLoc differs from more general knowledge distillation as we keep Φ_{layout} to use for matching with the outputs of Φ_{image}. Therefore, the task requires that Φ_{image} not only maintains the relative differences between embeddings, and with it their respective layout similarity, but that it is anchored to the corresponding embedding of Φ_{layout}.

In the following, we describe the learning of the latent space, as well the mapping of other modalities to it. An overview of this training procedure is shown at the top of Figure 2 and the architecture used is given in the supplementary materials.

4.1. Φ_{layout} : Layout-similarity Latent Space

We first learn a rich latent representation of room layouts by imposing a metric loss for training the layout branch Φ_{layout} alone. The embedding space is conditioned so that
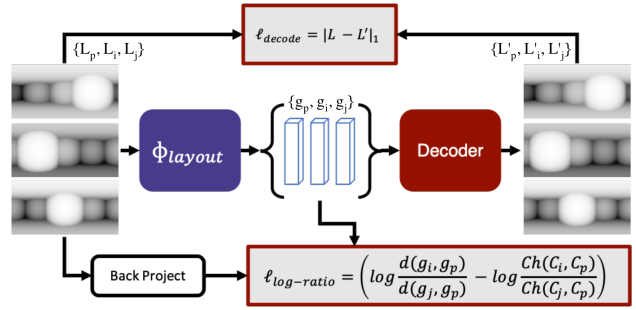
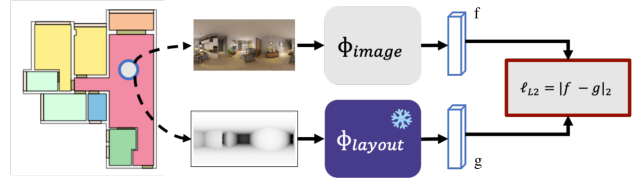**Layout Latent Space Training**

Image Branch Training

Figure 2. Overview of the LaLaLoc training process. *Top:* The learning of the layout latent space performed solely by Φ_{layout}. *Bottom:* The routine used to train Φ_{image} to map RGB panoramas to the latent space learned by Φ_{layout}.

the distance between the latent representations of layout renderings should reflect the structural difference in their respective layouts. This is achieved through a log-ratio loss formulation, as proposed by Kim *et al.* [19]:

$$\ell_{log\_ratio}(p,i,j) = \left(log\frac{D(g_i,g_p)}{D(g_j,g_p)} - log\frac{\text{Ch}(C_i,C_p)}{\text{Ch}(C_j,C_p)} \right)^2,$$
(1)

where (p,i,j) represents a triplet of consisting of an anchor p, given by the pose of the RGB panorama, with i and j as two neighbours of p; $g = \Phi_{layout}(L)$ represents the layout embedding computed from layout rendering L; C indicates the pointcloud generated from back-projection of the layout depths; $D(\cdot)$ is the Euclidean distance function and $\text{Ch}(\cdot)$ is the Chamfer distance.

In addition to the layout similarity loss, we also leverage a layout decoder for training. The decoder takes the latent vector $g = \Phi_{layout}(L)$ and decodes it to layout depth image, L' thus forming a layout auto-encoder during training. We apply a decoding loss in the form:

$$\ell_{decode} = |L' - L|_1,$$
(2)

with the L1-norm chosen for its favourable performance for depth estimation [11].

4.2. Φ_{image} : Learning to Match Images to Layouts

Once the layout latent space has been learned by the teacher branch, we train our student branch, Φ_{image}, to embed RGB images to the same layout embedding space. For

this training, we freeze Φ_{layout}, therefore the RGB branch is tasked only with mapping images to a fixed layout space and can be aided by the response of Φ_{layout}.

We do this simply by applying a loss on the Euclidean distance between the RGB embedding, f, and the layout embedding, g. This mirrors the localisation strategy where image-layout matching will be predicted through the Euclidean distance between the respective embeddings in the latent space. The loss is given simply by:

$$\ell_{L2}(p) = |f_p - g_p|_2. \tag{3}$$

5. Localisation

While it would be possible to fine-tune the latent representations computed by LaLaLoc when localising in new scenes, we instead propose the approach as a fixed network, which is able to generalise to new scenes without any fine-tuning or RGB training trajectories.

LaLaLoc is a 2-stage localisation approach: a coarse, global retrieval stage, followed by cross-modal pose optimisation through differentiable rendering. An overview of the localisation method is depicted in Figure 1. In the following, we describe each of these stages in more detail.

5.1. Coarse retrieval

The coarse retrieval stage operates by sampling candidate poses in a uniform grid across the known floor plan. At each sampled pose, we render the layout from the known floor plan geometry, which is then used to compute a latent vector. These latent vectors and their associated poses form the reference database for localisation. When performing localisation of a query image, we compute the latent vector of the query image and compute its distance with respect to each entry in the reference database. The coarse localisation estimate is given by the pose belonging to the nearest neighbour latent vector.

5.2. Pose refinement

The first, coarse stage of the LaLaLoc localisation method is limited by the density of the sampled poses within the floor plan. We, therefore, include pose refinement to enable more fine estimate of the camera location. We propose two methods for this: a retrieval-based approach which re-samples more densely around the coarse estimate and again localises by retrieval; or direct optimisation of pose by layout similarity in the latent space.

To more densely explore the region around the nearest neighbour from coarse retrieval, we sample poses in a Vogel Disc [36] centred around the nearest neighbour. This produces an (approximately) even sampling in a circular region around the neighbour. The refined pose is then returned as the most similar from these newly sampled poses.

We will describe the formulation of the direct optimisation more thoroughly below.

5.2.1 Latent Optimisation of Pose

We propose a direct pose optimisation through differentiable rendering. While differentiable rendering-based approaches have been shown to be effective for pose estimation [23, 12], these works rely on homogeneous data to compute losses between the prediction and the target, often employing pixelwise losses based on photo-metric or depth reconstruction errors. However, in our application we again must tackle the challenge of the asymmetry of our query (RGB) and reference (layouts) data types.

Instead, we optimise for pose with latent losses, using distance between embeddings computed by our network to model the matching energy between RGB images and layouts. In doing, so we are able to bridge the gap between data modalities, without the need for explicit prediction. This is achieved by employing a differentiable renderer to compute the layout at a pose estimate, P_r, we can ensure that the chain of operations to compute a layout embedding at P_r are differentiable: $g_r = \Phi_{layout}(\Omega(P_r, \mathcal{M}))$, where $\Omega(P_r, \mathcal{M})$ is the layout rendering. We therefore can refine the pose using a gradient-based optimisation with objective:

$$\min_{P_r} D(\Phi_{layout}(\Omega(P_r, \mathcal{M})), f_p). \tag{4}$$

6. Dataset

Training and evaluation is performed on the Structured3D dataset [41]. The dataset consists of 3,500 synthetic indoor scenes. Each scene comprises of multiple rooms, leading to 21,835 rooms in total. Importantly for our task, a 3D floor plan is provided, as well as a photorealistic panorama image rendered for each room in the scene.

We follow the predefined split of scenes, with 3000 used for training, 250 used for validation and 250 used for testing. We found that some scenes had corrupted data, which were excluded resulting in 2979/246/249 scenes for training/validation/testing. Due to the split being conducted over scenes, it means that all evaluation within this paper is conducted on an unvisited room, where we define unvisited as a scene with no prior image capture *e.g.* a training trajectory captured in the target scene.

For each image, there are three furniture configurations: empty, simple and full. Each of these are rendered in three lighting conditions: warm, cool and raw. Unless otherwise stated, a configuration is selected randomly at each iteration during training and evaluation is performed in the "full" and "warm" furniture and lighting settings, respectively – with full being the most difficult due to the presence of more distractors when inferring layout.

7. Experiments

We detail the localisation performance of LaLaLoc for our main task of localisation in an unseen environment, as

Query Type	Model	Layout R@1	Pose R@1	Localisation Accuracy				
				Median (cm)	<1cm	<5cm	<10cm	<1m
	Oracle	*100%*	*91.1%*	*20.4*	*0.3%*	*3.61%*	*13.5%*	*92.9%*
Layout	2D ICP	-	-	0.5	**87.9%**	**99.8%**	**99.8%**	**99.9%**
	LaLaLoc	90.1%	87.4%	**0.2**	69.1%	82.4%	91.4%	94.9%
RGB/Furnished	2D ICP	-	-	21.8	**9.5%**	26.4%	35.6%	68.5%
	HorizonNet [32] + Loc	72.4%	68.0%	9.1	3.3%	29.3%	53.4%	77.4%
	LaLaLoc	72.4%	70.6%	**8.3**	3.6%	**32.0%**	**58.0%**	**87.5%**

Table 1. Layout localisation accuracy for our trained models and baselines on the Structured3D test set.
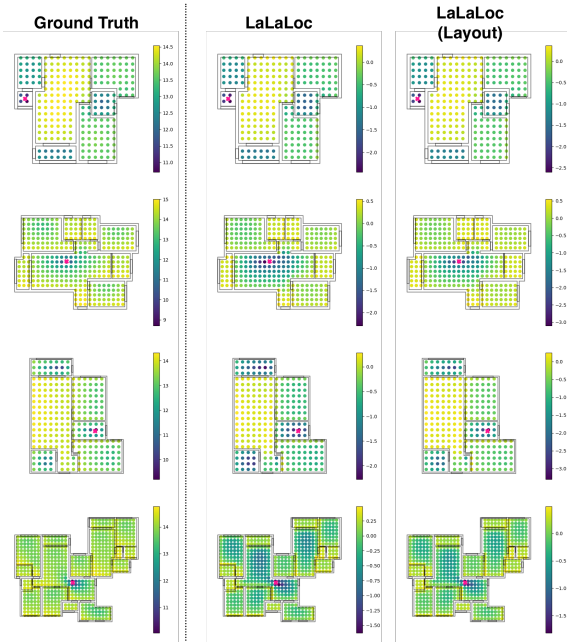

Figure 3. Qualitative depiction of the layout distances between a query pose, marked with a pink "X", and the grid of sampled poses across the scene. For each sampled pose, we colour it by the log of its respective distance from the query. *Left:* we plot the ground-truth layout distances. *Middle:* the layout distance is that predicted by LaLaLoc. *Right:* predicted by LaLaLoc in its layout-to-layout configuration.

well as analyse LaLaLoc's components and their contribution to the final accuracy. Unless otherwise stated, we sample a 2-dimensional grid of locations at a resolution of 0.5m × 0.5m. We then either take the nearest neighbour pose as our prediction, or use it to initialise the refinement stage. Our Vogel Disc re-sampling is set to a radius of twice the grid resolution. It is worth noting that all evaluations are performed with only the top 1 retrieval, which is the hardest test setting as there is limited scope to recover from a bad retrieval.

We include results for the two main modes of operation for LaLaLoc. The first assumes that the ground truth room layout is known for the query image. In this configuration, the layout branch, Φ_{layout}, is used alone. This setting is ideal for determining the expressiveness of the learned layout latent space as there are no discrepancies caused by translation from image to the layout latent space. The other configuration, however, is the more realistic scenario, where RGB panoramas are used as the query image, with Φ_{image} mapping them to LaLaLoc's embedding space.

For a baseline comparison, we include results for an ICP-based method. Similar methods are widely used in fields like robotics for motion estimation [22], scene reconstruction, map building and localisation [25]. The implementation details of this method are provided in the supplementary materials. In addition, we compare against a baseline formed from HorizonNet [32]. Given a query image, we use HorizonNet to predict its layout explicitly. We then retrieve the nearest neighbour from the sampled grid via L1 distances between depths. We perform VDR initialised at the nearest neighbour. Finally, we perform a gradient-based optimisation between the predicted layout and layouts rendered from the floor plan. It is worth noting that this is a significant extension to HorizonNet to perform localisation.

Various performance metrics are compiled across the following experiments that pertain to retrieval, and the final estimation of pose. The retrieval metrics provide insight into the performance of the first localisation stage: *Layout Recall @1* measures the proportion of the time the predicted nearest neighbour layout is the most similar layout; *Pose Recall @1* measures the fraction of predicted nearest neighbours that are the neighbour with the nearest pose to the query. Localisation accuracy is given by: *Median Pose Error* as the median distance between query and predicted pose; *Accuracy$< \tau$* lists the fraction of frames localised to within the threshold, τ.

7.1. Localisation with a Floor Plan Prior

Through the full localisation procedure outlined in Section 5, we detail LaLaLoc's performance in Table 1. Worth considering first is the retrieval accuracy. The layout-to-layout retrieval performance deteriorates only slightly from the layout oracle, suggesting that the latent space is able to
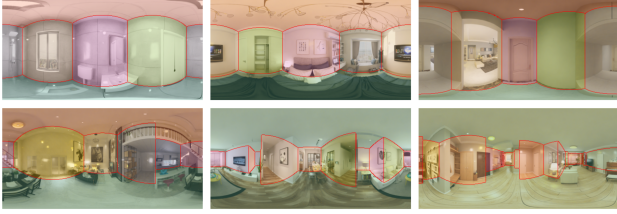
Figure 4. Example failures. *Top:* Cases where the wrong room has been retrieved due to ambiguity of layout, but the alignment is generally coherent with the layout seen in the query image. *Bottom:* The correct room is retrieved, but the alignment is incorrect. In these cases, it is often seen that room corners are incorrectly aligned to the edges of objects such as tables and counter tops.

capture layout extremely well. For qualitative confirmation of this, we plot ground truth and inferred layout distances in Figure 3. There it can be seen that the inferred difference or similarity between layouts is highly representative of the ground truth. Given the increased difficulty of inferring layout from RGB images, the cross modal retrieval performance of LaLaLoc, as seen qualitatively in Figure 3, is very good. This is emphasised by the retrieval accuracy where 87.5% of retrievals were within 1m of the true pose and thus within the radius of the Vogel Disc refinement.

In the layout-to-layout configuration, LaLaLoc is able to perform competitively against our ICP baseline. This is the ideal case for ICP, since it computes alignment on the point-clouds directly, both of which are known to the exact scale, whereas LaLaLoc introduces a layer of abstraction between the reference and the query. In the panorama-to-layout setting we see that LaLaLoc outperforms the baselines tested, including the method involving explicit layout prediction with HorizonNet [32] and aligning it to the known floorplan geometry.

We visualise LaLaLoc's two main types of failure in Figure 4. The most common is that an incorrect room with similar geometry is retrieved. In this scenario, the pose is still generally refined to a location that produces a plausible alignment. In the other scenario, the correct room is retrieved but the alignment is incorrect. Often in this scenario we see that floor or wall edges are aligned to objects in the room, suggesting that the refinement may have been caught in a local minimum.

In Table 2, we provide further results for localisation performance with a lower resolution grid of poses used for sampling. Specifically we sample at a resolution of 1m × 1m, as opposed to the original 0.5m × 0.5m. Despite the retrieval error being more than double that seen with the 0.5m grid, LaLaLoc is still able to recover from this during its refinement stage with final accuracy only seeing a small degradation in accuracy.

| | | | Localisation Error | | |
	Method	Med.	<1cm	<10cm	<1m
Lay.	*Retr. Oracle*	*42.7*	*0.0%*	*2.9%*	*75.7%*
	2D ICP	0.7	**76.9%**	**99.4%**	**99.6%**
	LaLaLoc	**0.5**	56.1%	70.5%	84.3%
Pano.	2D ICP	21.7	**7.3%**	34.7%	69.4%
	LaLaLoc	**11.5**	2.6%	**46.1%**	**79.5%**

Table 2. Localisation performance when sampling on a lower resolution, 1m × 1m grid.

Furniture Level	Method	Layout R@1	Pose R@1	Median Error (cm)
Empty	2D ICP	-	-	**2.4**
	LaLaLoc	76.4%	74.0%	5.7
Simple	2D ICP	-	-	10.0
	LaLaLoc	74.8%	73.3%	**6.7**
Full	2D ICP	-	-	23.0
	LaLaLoc	72.4%	70.6%	**8.3**

Table 3. Impact of furniture level on localisation performance. LaLaLoc is run with a full refinement scheme (VDR + LPO) for computation of localisation error.

7.1.1 Robustness to Level of Furniture

With the introduction of furniture in the second test setting, the effectiveness of LaLaLoc becomes readily apparent. LaLaLoc now significantly outperforms the ICP baseline in nearly all the metrics evaluated. In Table 3, we further analyse the impact of the furniture configuration on the final localisation accuracy by evaluating performance across those provided in the Structured3D dataset. Here, it becomes apparent that, although the ICP baseline performs very well when operating on the ground-truth layouts, when the query type becomes increasingly furnished its performance drops remarkably. On the other hand, the results show that LaLaLoc sees very little deterioration as the number of distractors in the scene increase from empty to full. This shows that Φ_{image} within LaLaLoc is able to accurately infer room layout from the RGB images, even when the room is cluttered.

7.1.2 Pose Refinement

To evaluate the performance of LaLaLoc's pose refinement, here we compare various strategies. Specifically, we investigate the combination of the Vogel Disc re-sampling and latent pose optimisation to improve upon the coarse pose as predicted by retrieval. We also include an explicit prediction and alignment strategy, "VDR + Decode". This is implemented by keeping the decoder from training and using it to predict the query layout explicitly. Pose is then pre-

Refine	Localisation Error			
Method	Med.	<1cm	<5cm	<10cm
Retrieval	22.5	0.3%	3.19%	11.8%
VDR-only	11.0	0.8%	15.1%	45.3%
LPO-only	10.5	2.0%	27.4%	48.4%
VDR + Decode	12.2	0.5%	13.2%	39.7%
VDR + LPO	**8.3**	**3.6%**	**32.0%**	**58.0%**

Table 4. Comparison of latent pose optimisation against a render-and-compare method, "Decode", as well as against no optimisation. All refinement methods are initialised at the pose returned from our Vogel disc re-sampling.

dicted via a gradient-based optimisation on the L1 distance between the predicted layout and rendered layouts from the floor plan. Results are listed in Table 4. The VDR and LPO refinement stages of LaLaLoc are both shown to be effective, with each able to improve on the retrieval alone. Our latent pose optimisation applied alone outperforms VDR, but the best localisation performance is given by using the re-sampling to initialise the latent pose optimisation.

In comparison to the decode baseline, it is clear from the results that the optimisation within the latent space leads to superior results. In fact, the refined poses through decoding performed worse than the retrieval poses used to initialise them. We believe the discrepancy in refinement accuracy can be explained by layouts being easier to capture layout in the latent space, than they are to articulate, *i.e.* to explicitly predict. The simple depth decoder may not be sufficiently intricate to predict layouts suitable for this refinement. After all, prediction of general room layouts is an activate area of research in itself [16, 32, 31].

As a further evaluation of the latent pose optimisation, we perform refinement with a varying number of more densely sampled poses in the naive Vogel Disc refinement strategy. As can be seen in Figure 5, the benefit from the re-sampling decreases quickly as the number of samples decreases. This experiment also demonstrates the ability of latent pose optimisation to recover from increasingly poor initialisation provided by the Vogel Disc re-sampling. In all tested configurations, the latent pose optimisation was able to improve upon the re-sampled nearest neighbour, and the decrease in accuracy was far less severe.

7.2. Ablation

In this section, we validate various components of LaLaLoc's design, from the demonstration that layout localisation is feasible and the metric used for layout similarity, to the training objectives for the network. Unless otherwise stated, the results are all validation accuracy and correspond to retrieval only, with no pose refinement performed. In the experiments we mark our chosen design with an *.
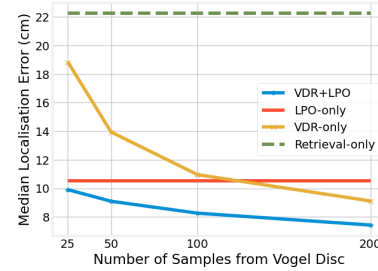

Figure 5. Investigation of the impact of sampling density from the Vogel Disc before performing latent pose optimisation. We include results for performing latent pose optimisation without the Vogel Disc re-sampling beforehand, as well as reference to the initial retrieval accuracy used to initialise these refinement schemes.

Similarity Metric	Recall @1	Pose Error Median (cm)	Correct Room
Pose	*100%*	*19.8*	*100%*
Edges	79.3%	20.3	83.6%
Depth	80.3%	20.1	84.0%
Rel. Depth	77.1%	20.1	85.4%
Chamfer*	**90.8%**	**20.0**	**92.1%**

Table 5. Evaluation of layout similarity metrics for the localisation task on the Structured3D validation split. *Pose* refers to picking the nearest location in the sampled grid to the query.

7.2.1 Layout-similarity Metric

When discussing localisation from a floor plan, it is not unreasonable to think of room shapes identical to one another, thus leading to significant ambiguity and rendering localisation ineffective without other cues. To some degree this ambiguity cannot be completely removed when localising with respect to layout alone. However, we hypothesise that there are significant differences between layout representations and the respective measures of similarity in their susceptibility to this ambiguity. This means that, if the wrong metric is chosen, layout similarity can become a much less expressive signal for localisation. Therefore, we explore which representation of an arbitrary room layout gives the best performance for localisation.

Specifically, we evaluate four representation and layout similarity metric pairs: *Edges*, the 2D Chamfer distance between sets of edge pixels in layouts represented as an edge segmentation, used in [6] for floor plan localisation; *Depth*, defined as the L1 distance between layouts rendered as depth images; *Relative Depth*, the L1 distance between relative depth images (max depth value = 1 in all images); and *Chamfer distance*, where the depth images are back-projected and the Chamfer distance is computed between the resulting pointclouds.

We list the results in Table 5. As can be seen in the table,

Query Type	Training Routine	Layout R@1	Pose R@1	Median Error (cm)
Lay.	End-to-End	81.1%	80.4%	20.9
	2-stage*	**89.3%**	**87.6%**	**19.9**
RGB	End-to-End	24.4%	23.7%	62.7
	2-stage*	**70.7%**	**70.5%**	**21.7**

Table 6. Retrieval comparison with an end-to-end training on the Structured3D validation split.

Losses	Layout R@1	Pose R@1	Median Error (cm)
Oracle	*100%*	*90.8%*	*20.0*
ℓ_{log_ratio}	78.8%	77.3%	20.5
ℓ_{decode}	**90.9%**	84.6%	20.1
$\ell_{log\_ratio} + \ell_{decode}$*	89.3%	**87.6%**	**19.9**

Table 7. Layout-to-layout teacher model ablation on the Structured3D validation set. We evaluate the contribution of each of the losses used to train our layout branch.

Backbone	Predictor	Layout R@1	Pose R@1	Median Error (cm)
ResNet18*	FC*	**89.3%**	**87.6%**	**19.9**
ResNet50	FC	87.5%	85.3%	20.0
ResNet18	MLP	87.2%	86.3%	**19.9**

Table 8. Comparison of network architecture for Φ_{layout} on the Structured3D validation split.

Losses	Layout R@1	Pose R@1	Median Error (cm)
Oracle	*100%*	*90.8%*	*20.0*
ℓ_{L2}*	**70.7%**	**70.5%**	**21.7**
ℓ'_{log_ratio}	59.6%	56.9%	25.1
$\ell_{L2} + \ell'_{log\_ratio}$	60.7%	59.9%	24.6
$\ell_{L2} + \ell_{kd\_lr}$	58.9%	56.9%	25.0

Table 9. Image-to-layout model ablation on the Structured3D validation set. We evaluate the contribution of each of the losses used to train Φ_{image} from the frozen layout branch.

representing layout as a pointcloud with the Chamfer distance provides the best formulation of those tested, significantly out-performing the image-based similarity metrics tested.

7.2.2 End-to-End Training

We compare our proposed 2-stage training to an end-to-end formulation. We use the same losses as described previously, however we found it best to cut the gradient of g when applied in ℓ_{L2}. Therefore, Φ_{layout} is only optimised to minimise ℓ_{log_ratio} and ℓ_{decode}, mirroring the 2-stage approach. In Table 6, we list the results of this training strategy. Φ_{layout} is still able to learn a representative latent space, albeit significantly less so than when trained alone. However, Φ_{image} performs very poorly when trained in this manner, despite trying a number of different loss scaling factors. We hypothesise that the other losses pollute the gradients to Φ_{image}, similar to the introduction of additional losses for training Φ_{image}, as explored later. With this, it appears that our 2-stage training is the more effective, or at least significantly more forgiving.

7.2.3 Layout Latent Space Learning

Here we evaluate multiple objective functions to learn LaLaLoc's latent layout embedding space, *i.e.* for training Φ_{layout}. Results are listed in Table 7, where we see that the combination of the log-ratio loss and decoder loss provides the most effective latent space for layout-similarity matching and localisation. Notably, it appears that the autoencoder formulation is more representative than training with the relational similarity constraints alone.

In Table 8 we evaluate some alternative architectures for Φ_{layout}. Specifically, we test replacing the ResNet18 backbone with ResNet50, and replacing the single fully connected layer with a 2-layer MLP after pooling. Although all options perform similarly, we find that the simplest architecture produced the best retrieval results.

7.2.4 Mapping Images to the Latent Space

When training Φ_{image}, there are many possibilities for how to best learn a mapping of RGB images to the existing latent space. In Table 9, we list results of various training objectives considered in the design of LaLaLoc. In this table: ℓ'_{log_ratio} refers to a cross-modal adaptation of the original loss, where g_p is replaced by f_p; ℓ_{kd_lr} is a knowledge distillation variant of the log-ratio loss, where the ground-truth layout similarity is replaced by the embedding distances as computed by Φ_{layout}. Equations for these losses are provided in the supplementary materials. Interestingly, the most simple objective, reducing the Euclidean distance between the image embedding and its corresponding layout embedding, resulted in the best results.

8. Conclusion

In this paper we have presented LaLaLoc, a localisation method which localises RGB queries to a known floor plan by matching in a latent space that encodes layout similarity. We further leverage this expressive latent space for direct pose optimisation through differentiable rendering. We show that LaLaLoc is able to localise with considerable accuracy in unseen or highly dynamic environments.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016. 2

[2] Vassileios Balntas, Shuda Li, and Victor Prisacariu. Relocnet: Continuous metric learning relocalisation using neural nets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 751–767, 2018. 2

[3] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. 2

[4] P. Biber and W. Strasser. The normal distributions transform: a new approach to laser scan matching. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, volume 3, pages 2743–2748 vol.3, 2003. 2

[5] Hermann Blum, Julian Stiefel, Cesar Cadena, Roland Siegwart, and Abel Gawel. Precise robot localization in architectural 3d plans. *arXiv preprint arXiv:2006.05137*, 2020. 2

[6] Federico Boniardi, Abhinav Valada, Rohit Mohan, Tim Caselitz, and Wolfram Burgard. Robot localization in floor plans using a room layout edge extraction network. *arXiv preprint arXiv:1903.01804*, 2019. 2, 7

[7] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6684–6692, 2017. 2

[8] Eric Brachmann and Carsten Rother. Learning less is more-6d camera localization via 3d surface regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4654–4662, 2018. 2

[9] Eric Brachmann and Carsten Rother. Expert sample consensus applied to camera re-localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7525–7534, 2019. 2

[10] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2616–2625, 2018. 2

[11] Marcela Carvalho, Bertrand Le Saux, Pauline Trouvé-Peloux, Andrés Almansa, and Frédéric Champagnat. On regression losses for deep depth estimation. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2915–2919. IEEE, 2018. 3

[12] Wenzheng Chen, Jun Gao, Huan Ling, Edward J Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. *arXiv preprint arXiv:1908.01210*, 2019. 1, 4

[13] Mingyu Ding, Zhe Wang, Jiankai Sun, Jianping Shi, and Ping Luo. Camnet: Coarse-to-fine retrieval for camera re-localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2871–2880, 2019. 2

[14] Harlan Hile and Gaetano Borriello. Positioning and orientation in indoor environments using camera phones. *IEEE Computer Graphics and Applications*, 28(4):32–39, 2008. 2

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[16] Henry Howard-Jenkins, Shuda Li, and Victor Prisacariu. Thinking outside the box: generation of unconstrained 3d room layouts. In *Asian Conference on Computer Vision*, pages 432–448. Springer, 2018. 7

[17] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5974–5983, 2017. 2

[18] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938–2946, 2015. 2

[19] Sungyeon Kim, Minkyo Seo, Ivan Laptev, Minsu Cho, and Suha Kwak. Deep metric learning beyond binary supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2288–2297, 2019. 2, 3

[20] Hyon Lim, Sudipta N Sinha, Michael F Cohen, and Matthew Uyttendaele. Real-time image-based 6-dof localization in large-scale environments. In *2012 IEEE conference on computer vision and pattern recognition*, pages 1043–1050. IEEE, 2012. 2

[21] Liu Liu, Hongdong Li, and Yuchao Dai. Efficient global 2d-3d matching for camera localization in a large-scale 3d map. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2372–2381, 2017. 2

[22] Jorge L Martínez, Javier González, Jesús Morales, Anthony Mandow, and Alfonso J García-Cerezo. Mobile robot motion estimation by 2d scan matching with genetic and iterative closest point algorithms. *Journal of Field Robotics*, 23(1):21–34, 2006. 5

[23] Andrea Palazzi, Luca Bergamini, Simone Calderara, and Rita Cucchiara. End-to-end 6-dof object pose estimation through differentiable rasterization. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 1, 4

[24] François Pomerleau, Francis Colas, and Roland Siegwart. A review of point cloud registration algorithms for mobile robotics. *Foundations and Trends in Robotics*, 4(1):1–104, 2015. 2

[25] Jose-Raul Ruiz-Sarmiento, Cipriano Galindo, and Javier Gonzalez-Jimenez. Building multiversal semantic maps for mobile robot operation. *Knowledge-Based Systems*, 119:257–272, 2017. 5

[26] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12716–12725, 2019. 2

[27] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. Understanding the limitations of cnn-based absolute camera pose regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3302–3312, 2019. 2

[28] Grant Schindler, Matthew Brown, and Richard Szeliski. City-scale location recognition. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7. IEEE, 2007. 2

[29] Aleksandr Segal, Dirk Haehnel, and Sebastian Thrun. Generalized-icp. In *Robotics: science and systems*, volume 2, page 435. Seattle, WA, 2009. 2

[30] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2937, 2013. 2

[31] Sinisa Stekovic, Shreyas Hampali, 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*, pages 187–203. Springer, 2020. 7

[32] Cheng Sun, 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*, pages 1047–1056, 2019. 5, 6, 7

[33] Sebastian Thrun. Probabilistic robotics. *Communications of the ACM*, 45(3):52–57, 2002. 2

[34] James Unicomb, Ravindra Ranasinghe, Lakshitha Dantanarayana, and Gamini Dissanayake. A monocular indoor localiser based on an extended kalman filter and edge images from a convolutional neural network. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9. IEEE, 2018. 2

[35] Julien Valentin, Matthias Nießner, Jamie Shotton, Andrew Fitzgibbon, Shahram Izadi, and Philip HS Torr. Exploiting uncertainty in regression forests for accurate camera relocalization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4400–4408, 2015. 2

[36] Helmut Vogel. A better way to construct the sunflower head. *Mathematical biosciences*, 44(3-4):179–189, 1979. 4

[37] Johanna Wald, Torsten Sattler, Stuart Golodetz, Tommaso Cavallari, and Federico Tombari. Beyond controlled environments: 3d camera re-localization in changing indoor scenes. In *European Conference on Computer Vision*, pages 467–487. Springer, 2020. 1

[38] Shenlong Wang, Sanja Fidler, and Raquel Urtasun. Lost shopping! monocular localization in large indoor spaces. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2695–2703, 2015. 2

[39] Yoshiaki Watanabe, Karinne Ramirez Amaro, Bahriye Ilhan, Taku Kinoshita, Thomas Bock, and Gordon Cheng. Robust localization with architectural floor plans and depth camera. In *2020 IEEE/SICE International Symposium on System Integration (SII)*, pages 133–138. IEEE, 2020. 2

[40] Wera Winterhalter, Freya Fleckenstein, Bastian Steder, Luciano Spinello, and Wolfram Burgard. Accurate indoor localization for rgb-d smartphones and tablets given 2d floor plans. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3138–3143. IEEE, 2015. 2

[41] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photorealistic dataset for structured 3d modeling. *arXiv preprint arXiv:1908.00222*, 2(7), 2019. 4

[42] Wenzhao Zheng, Jiwen Lu, and Jie Zhou. Structural deep metric learning for room layout estimation. In *European Conference on Computer Vision*, pages 735–751. Springer, 2020. 2