[Supplementary Material]
JigsawPlan: Room Layout Jigsaw Puzzle Extreme Structure from Motion using Diffusion Models

Sepidehsadat Hosseini, Mohammad Amin Shabani, Saghar Irandoust, Yasutaka Furukawa
Simon Fraser University
{sepidh, mshabani, sirandou, furukawa}@sfu.ca

The supplementary document provides more details on our system and the competing methods (Sect. 1), more details on JigsawPlan and RPLAN datasets (Sect. 2), ablation studies on RPLAN dataset (Sect. 3), and additional qualitative examples (Figs. 1, 2, and 3) as promised in the main paper. Figure 1 shows more qualitative evaluations of our approach against the three competing methods. Figure 2 visualizes the samples predicted by our method at step t. Figure 3 shows more qualitative evaluations of our method for Full JigsawPlan and Full RPLAN datasets. Please also see the supplementary video for more examples.

1. Methods details of our system and competing methods

We benefit from Transformers in our task in two ways. First, Transformers provide the capability of processing sequences with different lengths, which we use to process different number of room layouts/corners in the houses. Second, we utilize the self-attention module of Transformers to create optimal interaction and information-sharing among input tokens. These two features make Transformers an ideal backbone for our model. Our method uses six Transformer encoder blocks, and attention in each block has four heads. We also use an MLP For converting 128d Transformer output to rotation and position (6d). To keep the experiments fair, we use the same architecture for our transformer baselines as much as possible. In the following, we provide details corresponding to each of the baselines.

Transformer with a raster representation uses an encoder part of U-Net, which has 8 down-sampling blocks, converting each input room layout to a feature map of dimension 512. Each feature map (corresponding to a room layout) will become one input token for the Transformer. Input sequence's length is equal to the number of rooms in a house, and information is shared among different rooms. We use six Transformer encoder blocks, and attention in each block has four heads. We pass the output of Transformer to a U-Net up-sampling model with eight Up-sampling blocks to change the dimension from 512 to 256×256.

Transformer with a vector representation uses the same backbone as our method; a linear layer converts the 28d input vector (i.e., 2 for the original corner coordinate and 26 for the room/door type one-hot vector) to the 128d feature map, six Transformer encoder blocks, and the attention in each block have four heads. We also use an MLP to convert 128d output embedding to 6d output.

Diffusion model, one room per node encodes each node as corresponding to a room instead of a corner in the room. To ease the implementation, we set the maximum number of nodes per room to 20 and we pad extra nodes when the room has less than 20 nodes with 0. We flatten the conditions per room and then use a linear layer to convert it to a 128d embedding vector. Each feature map represents a room and an input token for Transformer, we use the same Transformer as our method. After the Transformer blocks, a linear layer converts 128d output to 6d (i.e., 2 for the position and 4 for the rotation).

Shabani *et al.* [1] takes the input layout of each room with the resolution of 256×256 with the same number of channels as the number of room types to pass each pixel as a one-hot vector of the corresponding room type. We use the same model as [1] and change the number of input channels to 11 for RPLAN and 26 for JigsawPlan. To generate the arrangement candidates, we use the given room layouts of our dataset to connect doors, while we also use overlap filtering to reduce the number of candidates.

Note that our dataset is significantly larger than the one in [1], enabling us to randomly select a positive or a negative candidate in each iteration and therefore remove the class imbalance weight used in [1]. During the training for each house, we randomly select a GT with the label 1 or a faulty candidate with the label [0, 1) based on the number of mismatched doors. During the test, we pass all the possible candidates of each house and select the candidate with the

Room Type	3	4	5	6	7	8	9	10	All
Master bedroom	0.20	0.26	0.29	0.32	0.39	0.49	0.49	0.53	0.34
Living room	0.52	0.56	0.59	0.65	0.71	0.75	0.79	0.79	0.65
Kitchen	0.42	0.47	0.52	0.59	0.68	0.71	0.76	0.79	0.59
Bathroom	0.54	0.70	0.85	0.96	1.12	1.28	1.33	1.47	0.96
Toilet	0.07	0.11	0.15	0.22	0.22	0.25	0.27	0.26	0.18
Corridor	0.07	0.12	0.15	0.19	0.25	0.32	0.37	0.45	0.21
Closet	0.13	0.18	0.22	0.32	0.48	0.68	0.92	1.22	0.41
Hall	0.35	0.55	0.68	0.77	0.85	0.91	0.98	1.08	0.73
Laundry room	0.05	0.05	0.05	0.07	0.10	0.13	0.18	0.23	0.09
Bedroom	0.34	0.69	1.08	1.32	1.51	1.74	1.93	2.10	1.23
Balcony	0.05	0.08	0.16	0.32	0.40	0.48	0.56	0.64	0.29
Dining room	0.13	0.12	0.12	0.14	0.17	0.19	0.22	0.25	0.15
Private office	0.00	0.01	0.02	0.05	0.05	0.07	0.08	0.10	0.05
Den	0.05	0.05	0.06	0.7	0.09	0.11	0.12	0.12	0.08
Storage	0.00	0.01	0.01	0.02	0.02	0.02	0.03	0.03	0.02
Others	0.00	0.01	0.01	0.02	0.03	0.04	0.04	0.07	0.03
Doors	2.84	3.82	4.82	5.43	6.92	8.02	6.10	10.18	5.90

Table 1. JigsawPlan dataset consists of floorplans with 3 to 10 rooms. The table shows average number of rooms with a specific room type based on the total number of rooms in the house.

highest score as the final prediction.

2. Datasets Details

JigsawPlan database consists of 98K houses/apartments, which we divide into 93K training and 5K testing samples. Table 1 shows average number of rooms with a specific room type based on the total number of rooms in the house.

In the RPLAN dataset, we divide 60K samples in RPLAN to 55K train and 5K test. The number of rooms in a house ranges from 3 to 8. Concretely 99, 582, 5083, 19551, 21921, and 13235 houses contain 3, 4, 5, 6, 7, and 8 rooms, respectively.

3. Ablation studies on RPLAN

The main paper shows the ablation studies on the JigsawPlan dataset. The supplementary will present the same study results on RPLAN dataset. Table 2 shows the impact of our attention module and door matching loss on performance and Table 3 shows the impact of noise in the room type and door detection on our performance, although there is a performance drop, our method still works better than the competing methods.

Table 2. Contributions of our two attention mechanisms (R-SA, G-SA) and the door matching loss (L_{match}). Full RPLAN without the GT rotations are used for the evaluations. ✓indicates the feature being used.

R-SA	G-SA	L_{match}	MPE (↓)	GED (↓)
	✓		24.92	1.58
✓	✓		23.66	1.47
✓		✓	36.91	2.0
	✓	✓	21.93	1.07
✓	✓	✓	10.6	0.98

Table 3. Effects of the room-type (R-type) and the Door information. Full RPLAN without the GT rotations is used. ✓indicates the information being used. When a room-type is not used, we set a zero vector as a room-type one-hot vector. When the door information is not used, we do not pass the door-corner nodes to the network. Note that GED cannot be computed when the door information is removed.

Train		Test		MPE(↓)	GED (↓)
R-Type	Door	R-Type	Door		
✓	✓		✓	20.25	1.48
	✓		✓	16.54	1.12
✓	✓	✓		15.46	N/A
✓		✓		15.30	N/A
✓	✓	✓	✓	10.65	0.98

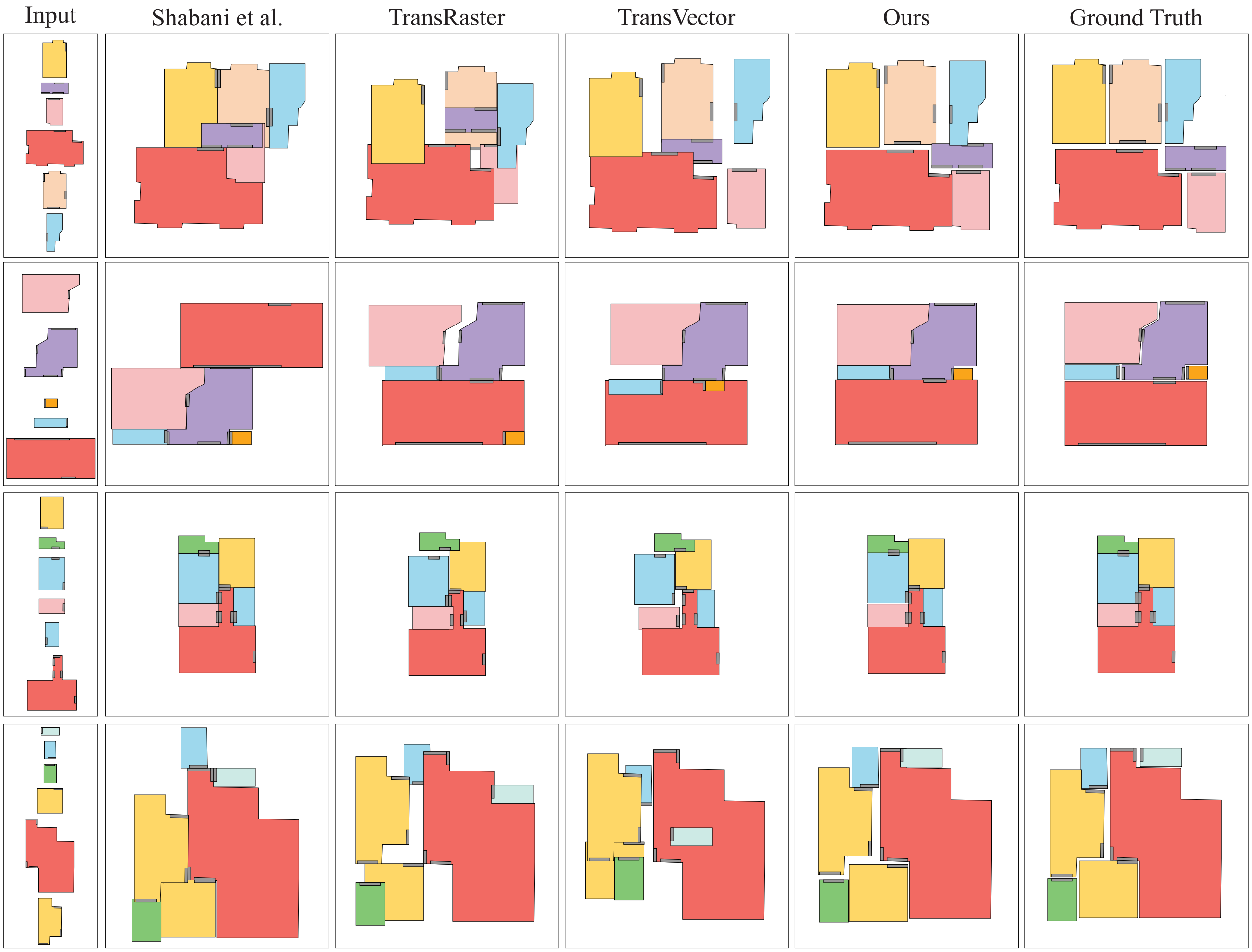

Figure 1. Qualitative evaluations of our approach against the three competing methods. The top two rows are from Small JigsawPlan. The bottom two rows is from Small RPLAN. The GT rotations are given for all the cases to enable comparisons with all the methods.

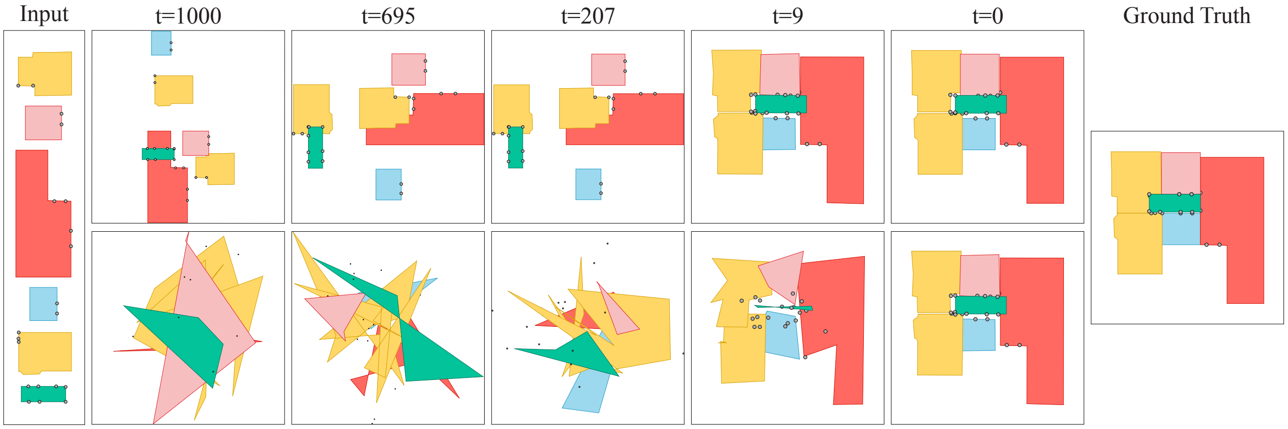

Figure 2. Visualization of predicted layouts at step "t"s. At t=1000, position parameters at each corner are initialized by a Gaussian noise, and at t=0, there is the final predicted layout. The top row shows the predicted layout without averaging/voting, and the bottom row shows with averaging/voting. To make it more clear, we show doors by their corners.

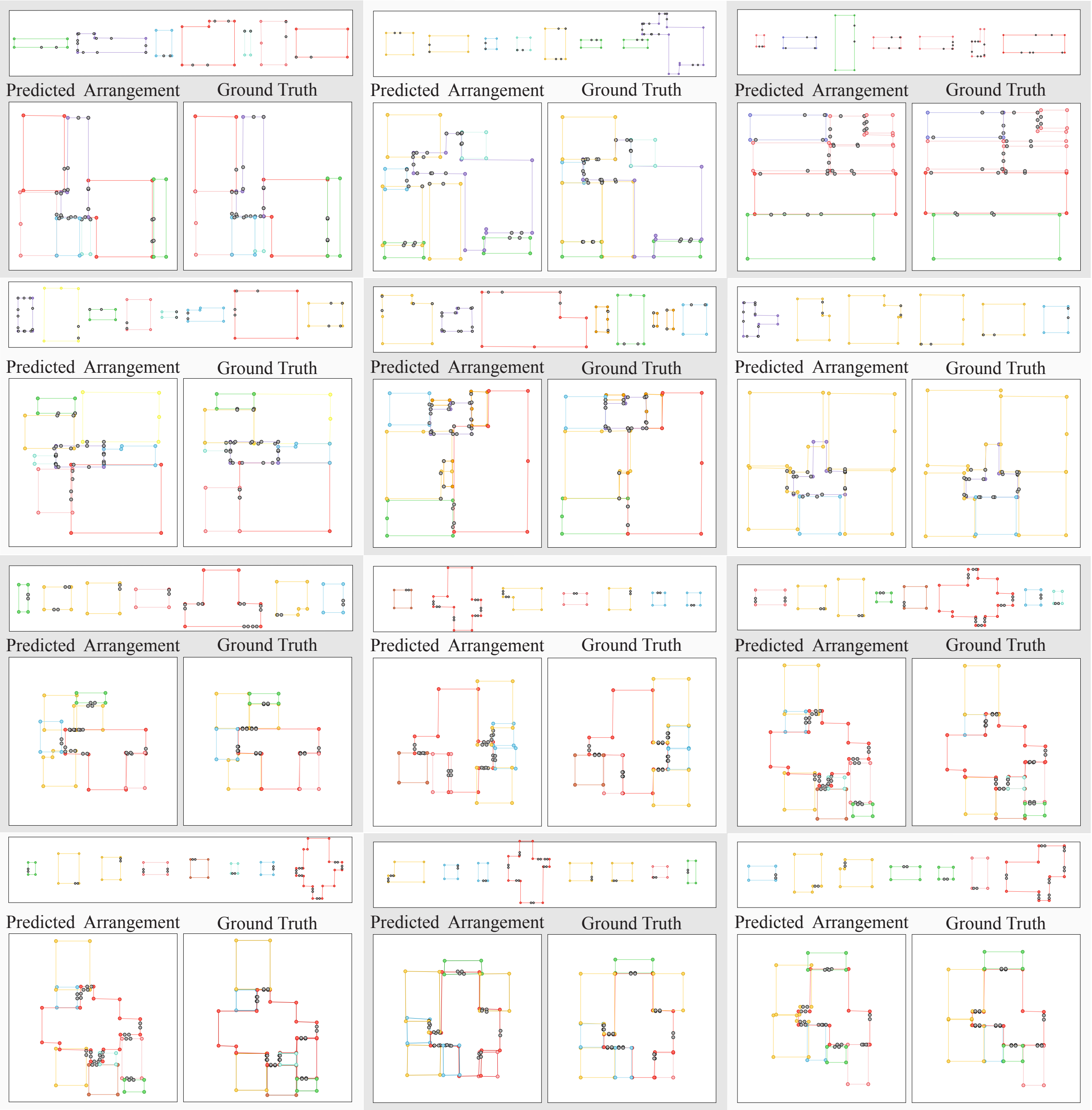

Figure 3. Qualitative evaluations of our method for Full JigsawPlan dataset without GT rotations top two rows, and Full RPLAN dataset without GT rotations bottom two rows. We show edges and corners here to show overlaps and noisy annotations more clear.

References

[1] Mohammad Amin Shabani, Weilian Song, Makoto Odamaki, Hirochika Fujiki, and Yasutaka Furukawa. Extreme structure from motion for indoor panoramas without visual overlaps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021. 1