What a Comfortable World: Ergonomic Principles Guided Apartment Layout Generation
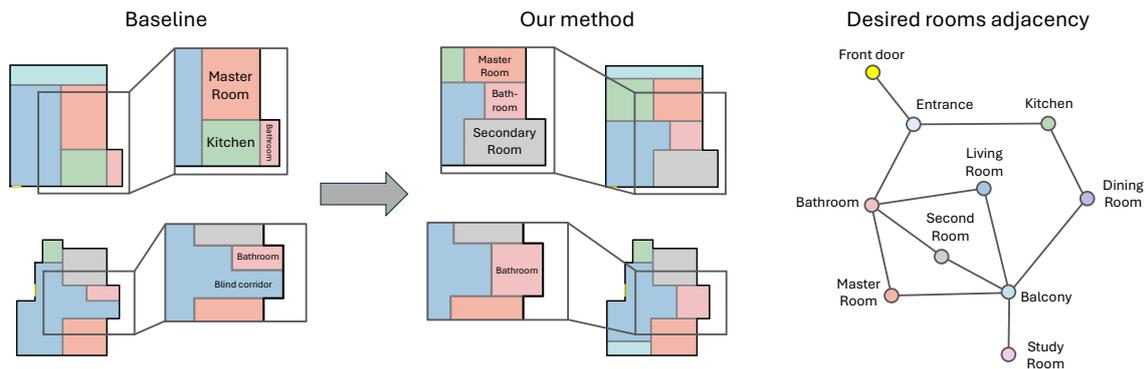
short1031

Figure 1: The left side illustrates representative examples of ergonomic floor plans generated by the proposed method, with comparison to a baseline method. The right panel displays the adjacency graph defining the desired spatial proximity between specific room pairs.

Abstract

Current data-driven floor plan generation methods often reproduce the ergonomic inefficiencies found in real-world training datasets. To address this, we propose a novel approach that integrates architectural design principles directly into a transformer-based generative process. We formulate differentiable loss functions based on established architectural standards from literature to optimize room adjacency and proximity. By guiding the model with these ergonomic priors during training, our method produces layouts with significantly improved livability metrics. Comparative evaluations show that our approach outperforms baselines in ergonomic compliance while maintaining high structural validity.

CCS Concepts

• *Computing methodologies* → *Shape modeling; Neural networks;* • *Applied computing* → *Architecture (buildings);*

1 Introduction

Automated generation of residential floor plans is a long-standing challenge in computer graphics, aiming to support architects with rapid exploration of design variations. Recent data-driven approaches, particularly those leveraging transformer architectures [RWC*19, VSP*17], have shown remarkable ability to learn complex spatial distributions from large datasets like RPLAN [WFT*19]. These models treat floor plan generation as a sequence prediction task, generating room boundaries and types token-by-token.

However, a fundamental limitation of pure data-driven methods is their reliance on the quality of training data. As observed in recent work on furniture arrangement [LGWM22], real-world datasets often contain "imperfect data"—layouts that are geometrically valid but potentially suboptimal regarding circulation or room place-

ment standards. A model trained naively on such data faithfully reproduces these inefficiencies. While Leimer et al. [LGWM22] addressed this by injecting differentiable guidance into the generation of discrete object arrangements, extending this paradigm to *structural floor plan generation* presents a distinct challenge. Unlike moving furniture within a fixed container, generating a floor plan requires optimizing the container itself—defining the topology, adjacency, and tiling of rooms simultaneously.

In this work, we propose a method to generate residential layouts with improved architectural compliance by integrating domain-specific principles directly into the generative process. We formulate a set of differentiable loss functions based on architectural standards [NN12], focusing on optimal entrance-to-kitchen proximity and bathroom accessibility. By incorporating these architectural adjacency and circulation priors into a GPT-2 based archi-

tecture [RWC*19] via a dynamic weighting scheme, we guide the model to produce layouts that adhere more closely to design guidelines than the training distribution.

Our contributions are: (1) a guideline-derived proximity cost formulation adapted for Manhattan-world room polygons; (2) a differentiable loss integration compatible with autoregressive token prediction; and (3) quantitative evidence that our method improves compliance metrics on the RPLAN dataset while maintaining high parsability, albeit with modest trade-offs in area coverage.

2 Related Work

Early layout synthesis relied on constraint satisfaction or evolutionary algorithms, but the field has shifted toward deep generative models. Graph-based approaches like constraint graphs [PGK*21] and House-GAN++ [NHC*21] represent layouts as nodes and edges, ensuring topological validity but often struggling with precise geometry. Most recently, transformer-based autoregressive models have become the state-of-the-art for vector graphics generation. SceneFormer [WYN20] and FaçAID [PSPSM24] demonstrate that transformers can effectively sequence geometric tokens to generate indoor scenes and building facades, respectively. We build on this sequence-based representation, adapting the GPT-2 architecture [WDS*20] to predict room polygons as coordinate tokens.

Purely generative models often fail to capture high-level functional requirements. To address this, neuro-symbolic methods hybridize neural generation with explicit rule-based constraints. LayoutEnhancer [LGWM22] pioneered this for indoor scenes by using differentiable scalar functions to penalize suboptimal furniture arrangements during training. This allowed the model to improve upon its own training data. Our work extends this principle from the *interior content* domain (furniture) to the *structural domain* (floor plans). We adapt the loss formulation to operate on polygon vertices and adjacencies, ensuring that the generated structures align with established architectural circulation guidelines [NN12].

3 Ergonomic cost

Using domain-specific literature [NN12] we determined that minimizing the following distances has a positive impact on the overall comfort of apartment use: (i) The distance from the entrance room to the front door. (ii) The distances from the entrance room, living room, master room and second room to the bathroom. (iii) The distances from the entrance room and dining room to the kitchen.

Additionally, it is desirable for the balcony to be adjacent to one of the following rooms: kitchen, dining room, living room, master room, secondary room, or study room.

Taking the above aspects into account, we define a set of room-type-specific cost functions to quantify layout ergonomics.

Entrance cost. For each entrance room $r \in R_{\text{entrance}}$, the entrance cost is defined as

$$E_{\text{entrance}}(r) = dist(r, d),$$

where d is the front door and *dist* denotes distance in meters.

Kitchen cost. For kitchens, we first pair each entrance and dining room with the nearest kitchen present in the floor plan. The cost

associated with a given kitchen is then computed as the average distance to all entrances and dining areas assigned to it:

$$E_{\text{kitchen}}(r) = \frac{1}{|A(r)|} \sum_{r' \in A(r)} dist(r, r'),$$

where $r \in R_{\text{kitchen}}$ is a kitchen and $A(r)$ is a set of rooms assigned to kitchen r.

Bathroom cost. The bathroom cost is defined in an analogous manner, but for each entrance room, living room, master room and second room we assign the nearest bathroom:

$$E_{\text{bathroom}}(r) = \frac{1}{|A(r)|} \sum_{r' \in A(r)} dist(r, r'),$$

where $r \in R_{\text{bathroom}}$ is a bathroom.

Balcony cost. For balconies, the cost is defined as the minimum distance to any of the preferred adjacent room types:

$$E_{\text{balcony}}(r) = \min_{r' \in R'} dist(r, r'),$$

where R' is a set of desired neighbors for a balcony – kitchens, dining rooms, living rooms, master rooms, secondary rooms and study rooms.

Overall ergonomic cost is computed as the mean cost over all rooms of applicable types:

$$E = \frac{1}{|R^*|} \sum_{r \in R^*} E_{a(r)}(r),$$

with $R^* = R_{\text{entrance}} \cup R_{\text{kitchen}} \cup R_{\text{bathroom}} \cup R_{\text{balcony}}$ and $a(r)$ is the type of room r.

4 Floor plans generation

Following trends in related work, we employ the GPT-2 model [RWC*19] (specifically the implementation included in the Hugging Face library [WDS*20]), which is based on the transformer architecture [VSP*17]. In this section, we describe the application of this model to the task of floor plan design, with a particular emphasis on the non-trivial incorporation of expert domain knowledge.

4.1 Floor plan representation

We encode each floor plan as a token sequence $S = (b, d, r_1, \ldots, r_n)$, where b is the boundary, d the front door placement, and r_i the i-th room. Boundary and door appear first; room order may vary.

Each segment starts with a type token (s_b for boundary, s_d for door, s_r^t for a room of type t) followed by quantized vertex coordinates (x, y) at fixed resolution:

$$b = (s_b, x_{b1}, y_{b1}, x_{b2}, y_{b2}, \ldots, x_{bk}, y_{bk})$$
$$d = (s_d, x_{d1}, y_{d1}, x_{d2}, y_{d2})$$
$$r_i = (s_i^t, x_{i1}, y_{i1}, x_{i2}, y_{i2}, \ldots, x_{il}, y_{il})$$

Following [WYN20, PGK*21, LGWM22, PSPSM24], we augment each token with an xy-index and a vertex-index. The xy-index is 1 for x-coordinates, 2 for y-coordinates, and 0 otherwise. The vertex-index is 0 for start tokens and counts coordinate pairs (1 for the first vertex, 2 for the second, etc.). Each index has its own learned

embedding, added to the token and positional embeddings during the forward pass.

4.2 Ergonomic loss

To inject expert knowledge into the model, we introduce a custom loss function using the ergonomic cost terms described in Section 3.

First, we define a differentiable distance metric between rooms. For room polygons r and s let $V_r = \{p_1^r, p_2^r, \ldots p_{n_r}^r\}$ and $V_s = \{p_1^s, p_2^s, \ldots p_{n_s}^s\}$ be the sets of their vertices. Then the distance metric D is defined as:

$$D(r,s) = \langle e, \text{softmin}(\beta \cdot e) \rangle, \text{ where } e_{ij} = \|p_i^r - p_j^s\|_2.$$

Here, β is a temperature parameter that determines the hardness of the softmin function (our experiments use $\beta = 10$).

Similarly to ergonomic cost, ergonomic loss is defined as a combinations of room-type-specific losses. For each entrance it is equal to the differentiable distance between the entrance polygon and the front door polygon.

$$L_{entrance}(r) = D(r,d),$$

where d is the front door polygon. The mean value over all entrances in a floor plan forms the full entrance loss term:

$$L_{entrances} = \frac{1}{|R_{entance}|} \sum_{r \in R_{entance}} L_{entrance}(r).$$

For kitchens, each entrance room and dining room distance is calculated to each kitchen. Then a differentiable minimum function is used to determine the distance from the nearest kitchen. The kitchen loss term is a mean value of these distances. Let $r_i \in R_{entrance} \cup R_{dining}$, $r_j \in R_{kitchen}$ and $\Delta_{ij} = D(r_i, r_j)$, then

$$L_{kitchen}(r_i) = \langle \Delta_{ij}, \text{softmin}(\beta \cdot \Delta_{ij}) \rangle,$$

$$L_{kitchens} = \frac{1}{|R_{entrance} \cup R_{dining}|} \sum_{r_i \in R_{entrance} \cup R_{dining}} L_{kitchen}(r_i).$$

Loss for bathrooms is defined in an analogical way, but using the proximity of entrances, living rooms, master rooms and second rooms:

$$L_{bathrooms} = \frac{1}{|R''|} \sum_{r_i \in R''} L_{bathroom}(r_i),$$

where $R'' = R_{entrance} \cup R_{living} \cup R_{master} \cup R_{second}$.

Balcony loss is similar, but here we are taking a differentiable minimum of proximate distance losses:

$$L_{balconies} = \langle L_{balcony}(r_i), \text{softmin}(\beta \cdot L_{balcony}(r_i)) \rangle, \ r_i \in R'.$$

Final ergonomic loss is a mean value of all possible to calculate losses

$$L_E = \frac{\sum_a \delta_a L_a}{\sum_a \delta_a},$$

with $a = \{entrances, kitchens, batchrroms, balconies\}$ and $\delta_a = 1$ if the corresponding loss is applicable to the particular floor plan and $\delta_a = 0$ otherwise.

4.3 Ergonomic loss usage

We integrate our ergonomic loss into the training process using a methodology similar to one presented in [LGWM22]. The ergonomic loss is calculated using ground-truth sequences where a single token is replaced by a value $\bar{v}$ derived from the model's predicted probability distribution.

Token $\bar{v}$ is equal to the expected value in a small window around the most likely value of the token

$$\bar{v} = \frac{\sum_j \mathcal{N}(v_j|\hat{v}, \sigma) P(v_j) v_j}{\sum_j \mathcal{N}(v_j|\hat{v}, \sigma) P(v_j)},$$

with $\mathcal{N}(x|\hat{v}, \sigma)$ being the normal distribution centered at $\hat{v}$ with standard deviation σ. $\hat{v}$ is equal to the most probable token and $P(v)$ is a estimated by the model. During our experiments σ was equal to $1/\rho$, where ρ is the resolution of floor plans quantization (number of tokens, which represents x- or y-coordinates).

Since the ergonomic loss is differentiable w.r.t. the room polygon vertices, the loss value for a given predicted token $\bar{v}$ is computed only when both $\hat{v}$ and its corresponding ground-truth element represent x- or y-coordinate of a room polygon vertex.

During the learning process we are combining the standard cross-entropy loss L_C with our ergonomic loss.

$$L = (1-\alpha) \cdot L_C + \alpha \cdot L_E,$$

where α is equal to the value of our ergonomic loss of the ground-truth floor plan divided by a scaling factor γ and clamped to $[0;1]$ range. During our experiments γ was equal to 30 and it was derived from the distribution of ergonomic losses calculated on test samples from the dataset. This definition ensures that ergonomic loss has stronger influence when training on non-ergonomic samples from the dataset. This makes the model lean more on expert knowledge for data points that don't align with ergonomic guidelines.

5 Results

We train on RPLAN [WFT*19] (quantized to 256). After validity filtering, the dataset contains 80,405 plans. We use a 90/5/5% train/val/test split. The training set was expanded to ∼159,000 examples via augmentation (rotation, symmetry, room permutation). The model (GPT-2, 25 layers, 16 heads, 256 embedding dim, 320 token context, 20M parameters) is trained for 150 epochs. A baseline model with identical configuration but trained with standard cross-entropy loss (no ergonomic term) serves as a comparison. Inference

Table 1: Quality metric values for test results of both the baseline and our full method as well as ground truth RPLAN dataset metrics.

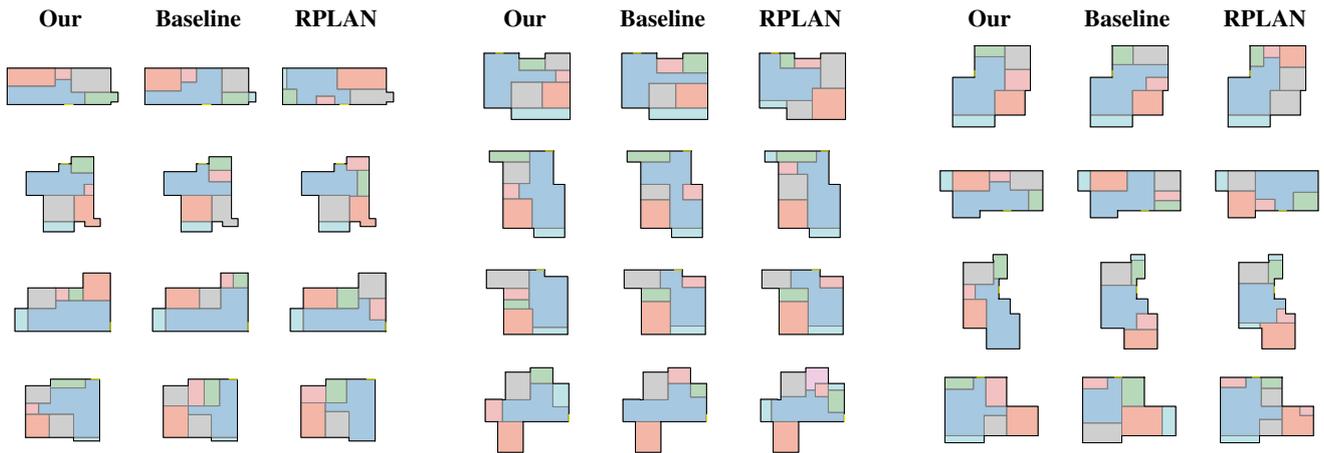| Metrics	Baseline	Our	RPLAN
Parsability (%)	0.999	0.999	1
Validity (%)	0.970	0.977	1
Fully covered (%)	0.798	0.742	1
No overlapping (%)	0.980	0.978	1
Ergonomic cost (m)	0.502	0.353	0.509
Perfect ergonomic cost (%)	0.343	0.437	0.281

Figure 2: Qualitative comparison of results between baseline, our method and samples form RPLAN dataset.
Legend: ■ - living room, ■ - master room, ■ - kitchen, ■ - bathroom, ■ - study room, ■ - second room, ■ - balcony.

uses greedy decoding. Testing starts from ground truth boundaries and doors, though fully generative mode is possible.

Qualitative results. Figures 1 and 2 illustrate our results. Ergonomic loss minimization is evident in bathroom placement (closer to master/secondary rooms) and reduced blocking of functional rooms (e.g., kitchen blocking bathroom in baseline, Fig. 1).

Quantitative results. We define test metrics: *Parsability* (successfully parsed sequences); *Validity* (no self-intersections); *Fully covered* (complete interior coverage); *No room overlapping*; *Ergonomic cost* (Sec. 3, lower is better); and *Perfect ergonomic cost* (% with 0 cost). Table 1 compares these metrics. Our model outperforms the baseline on ergonomics while maintaining comparable generation quality. The dynamic loss weighting (Sec. 4.3) enables learning ergonomic layouts despite the imperfect ergonomic adherence of the training data.

6   Limitations and Conclusions

Limitations. While our method successfully optimizes ergonomics, we observe a trade-off in geometric packing efficiency, resulting in slightly lower area coverage compared to the baseline. Additionally, the current model operates unconditionally, lacking explicit constraints for room counts or types, which limits direct user control. Finally, rare logical inconsistencies persist, such as entrance doors opening into unconventional spaces like balconies.

Conclusions. We presented a transformer-based data-driven method for residential floor plan generation that integrates domain-specific ergonomic principles. By incorporating differentiable loss functions for room adjacency and proximity, our model outperforms the baseline in ergonomic metrics despite the limitations of the training data. This approach demonstrates that incorporating domain-specific expert knowledge into training can effectively guide generative models toward more architecturally compliant and practical designs.

As future work, we plan to add conditional control over room types and counts and to improve geometric packing and logical consistency while preserving the benefits of the guided loss.

References

[LGWM22] LEIMER K., GUERRERO P., WEISS T., MUSIALSKI P.: Layoutenhancer: Generating good indoor layouts from imperfect data. In *SIGGRAPH Asia 2022 Conference Papers* (2022), pp. 1–8. 1, 2, 3

[NHC*21] NAUATA N., HOSSEINI S., CHANG K.-H., CHU H., CHENG C.-Y., FURUKAWA Y.: House-gan++: Generative adversarial layout refinement network towards intelligent computational agent for professional architects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 13632–13641. 2

[NN12] NEUFERT E., NEUFERT P.: *Architects' data*. John Wiley & Sons, 2012. 1, 2

[PGK*21] PARA W., GUERRERO P., KELLY T., GUIBAS L. J., WONKA P.: Generative layout modeling using constraint graphs. In *Proceedings of the IEEE/CVF international conference on computer vision* (2021), pp. 6690–6700. 2

[PSPSM24] PLOCHARSKI A., SWIDZINSKI J., PORTER-SOBIERAJ J., MUSIALSKI P.: FaçAID: A Transformer Model for Neuro-Symbolic Facade Reconstruction. In *SIGGRAPH Asia 2024 Conference Papers* (2024), SA '24. `doi:10.1145/3680528.3687657`. 2

[RWC*19] RADFORD A., WU J., CHILD R., LUAN D., AMODEI D., SUTSKEVER I., ET AL.: Language models are unsupervised multitask learners. *OpenAI blog 1*, 8 (2019), 9. 1, 2

[VSP*17] VASWANI A., SHAZEER N., PARMAR N., USZKOREIT J., JONES L., GOMEZ A. N., KAISER Ł., POLOSUKHIN I.: Attention is all you need. *Advances in neural information processing systems 30* (2017). 1, 2

[WDS*20] WOLF T., DEBUT L., SANH V., CHAUMOND J., DELANGUE C., MOI A., CISTAC P., RAULT T., LOUF R., FUNTOWICZ M., ET AL.: Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations* (2020), pp. 38–45. 2

[WFT*19] WU W., FU X.-M., TANG R., WANG Y., QI Y.-H., LIU L.: Data-driven interior plan generation for residential buildings. *ACM Transactions on Graphics (TOG) 38*, 6 (2019), 1–12. 1, 3

[WYN20] WANG X., YESHWANTH C., NIESSNER M.: SceneFormer: Indoor Scene Generation with Transformers. *CoRR abs/2012.09793* (2020). `arXiv:2012.09793`. 2