基于隐式表达的户型图矢量化

摘要

本文提出了一种全新的基于隐式表达的户型图矢量化方法，它将二维的带有噪声的户型图图像作为输入，并重建出一个具备户型图底层矢量化结构的平面图。现有的重建方法通常采用启发式设计的多阶段管线，重建效果取决于第一阶段户型图几何原语（角点、墙壁、房间）的检测。相反，我们将户型图矢量化问题看作是二维形状重建的问题，并且使用基于隐式表达的方法对户型图进行矢量化重建。在 Structured3D 数据集上的定性和定量的实验评估表明，我们的方法优于现有的户型图重建方法，并且具备更强的整体结构感知能力。

关键词： 户型图矢量化；隐式神经表达

1 引言

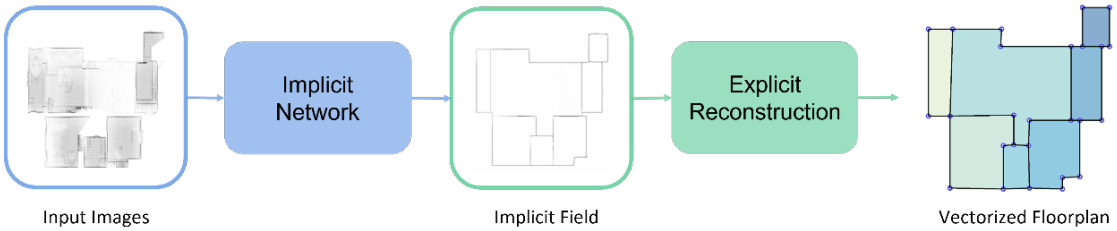

图 1. 模型管线

户型图是一种重要的数据表现形式，在机器人、AR/VR、室内设计等领域中均有广泛应用 [2] [3] [8] [9]。而户型图矢量化的总体目标是将一个室内场景转化为一个能够表征户型图几何结构的二维矢量图。具体来说，我们可以首先利用 RGB-D 相机、激光扫描仪或 SfM 系统捕获一个室内场景的三维点云，随后将点云数据沿着重力轴投影到二维平面中得到一个二维的密度图，通过设计一个矢量化的方法，将二维的密度图转换为一组二维的矢量化结构，用于表示户型图中每个房间的位置以及房间布局。通过上述方法采集到的二维密度图，尽管能够保留场景的建筑结构并且采集过程非常高效，但它也存在许多扫描过程中存留的噪声以及结构缺口，因此，户型图矢量化仍然是一项具有挑战性的任务。

自从深度神经网络出现之后，通过在大型的标注数据集上训练深度学习模型，户型图矢量化重建取得了突破性的进展。现有的方法大致可以分为两类：自上而下以及自下而上的方法。自上而下 [8] [31] 的方法首先利用图像分割模型 [17] 从密度图中分割出房间，随后采用优

化 [29] 或者是搜索 [5] 的技术将上述检测到的房间进行整合得到目标平面图；然而这类技术方式通常不是端到端训练的，它们的重建效果取决于第一阶段房间的分割效果以及第二阶段优化/搜索技术对于房间形状或布局结构的感知程度。而自下而上的方法 [9] [22] 首先检测平面图的角落，随后寻找角落之间构成的边缘 (即墙壁)，最后再将各个墙壁组合成一个矢量化的平面图。上述两类方法都是严格按照顺序进行的，如果第一阶段检测到的几何实体存在缺失，那么后续的重建效果会受到严重影响。

近几年来，隐式神经表达（Implicit Neural Representation）在三维任务中得到了广泛应用 [7] [12] [20] [24] [26]。对于空间中任意的一个坐标点，我们能够通过一个简单的多层感知机来将其映射为该坐标点在空间中的几何性质，该性质可以是与目标物体表面的最近距离 [26]，也可以是该坐标点在几何物体的表面的内外标志 [24]。相较于传统的体素网格 [4]、点云数据 [27] 或者是三角网格 [19]，隐式神经表达能够用非常少的参数来表征几何形状非常精细的细节，因此它被广泛地应用于三维形状 [24] [26]、三维场景 [20] 的重建任务中，但它在二维图像空间上的应用工作目前还比较少 [10]。

户型图矢量化重建问题，其本质上就是从一组包含很多噪声的信号中提取出结构化信号，这与三维重建的主要思想是一致的：输入粗糙的点云数据，输出高质量与高分辨率的三角网格。如果我们将二维的户型图视作二维的形状，那么我们就可以利用重建三维形状的方法来重建二维户型图，因此户型图重建问题适用于使用隐式神经表达的方法进行优化。在本次前沿技术报告中，我们提出使用一种紧凑的隐式神经函数 [35] 用于表征户型图并成功重建出户型图多层的矢量化结构。如图1展示的是本次项目的总体管线，我们首先利用一个隐式网络来预测输入户型图的隐式场，随后我们利用显式重建的方法从隐式表达中提取出户型图矢量化的几何元素。

通过使用隐式神经表征的方法重建户型图，我们无需像传统重建方法一样先检测几何实体，再利用复杂的优化方法聚合几何元素。我们的方法虽然简单，但具备更强的整体结构推理能力。我们在数据集 Structured3D [37] 上评估了我们的模型，定性和定量的实验结果表明，我们的模型在表现性能上优于近几年的户型图重建方法，且在实验结果上具有可观的提升空间。

2 相关工作

户型图重建的任务是：将原始的传感器数据（比如点云数据、户型图的投影密度图或 RGB 图像）转换为矢量化的几何结构。根据近几年的国内外研究现状以及技术发展的历史，我将户型图重建的算法分为三类：传统方法、基于深度学习的两阶段方法以及基于 Transformer [32] 架构的端到端方法。

2.1 户型图重建

2.1.1 传统方法

早期的传统方法依赖于基本的图像处理技术，比如霍夫变换 [1] 以及超像素分割 [28]。除此之外，过去还提出了利用更为复杂的技术来进行重建，比如利用基于图的模型 [16]，将户型图重建作为一个能量最小化的问题。这些方法具有一个共同的问题：需要涉及到大量的启发式方法或者参数来进行手工调整。

2.1.2 基于深度学习的两阶段方法

随着神经网络技术的发展，深度学习模型已经成为矢量化几何重建的基本框架。过去许多经典的方法都是采用了两阶段的模型设计，即模型首先检测出低水平的几何元素（比如角、墙壁以及房间），随后利用优化的方法将各个元素进行整合以完成重建。这其中比较经典的工作是 Floor-SP [8]，它首先利用预训练的 Mask R-CNN [17] 来检测户型图中的各个房间，随后通过求解最短路径问题来重建整个户型图。与 Floor-SP 类似，MonteFloor [31] 也是首先检测出单个房间，随后利用蒙特卡洛树搜索 [5] 的方法来重建整个户型图。上述的方法都是按照两步进行的，即先检测几何实体（角落、墙壁、房间），随后将其进行整合。当第一阶段检测到的实体存在问题时，无论第二阶段的方法有多鲁棒，其最终的重建效果都会受到严重影响。

2.1.3 基于注意力机制的端到端的方法

在自然语言处理领域中，Transformer [32] 架构取得了巨大成功并且被广泛应用为许多大语言模型的主干架构。它的贡献主要是引入了自注意机制（self-attention）以及交叉注意机制（cross-attention）作为基本的构建模块，这两种机制能够对输入序列中的每个元素之间建模出密集的联系，从而使得模型能够对整个输入序列进行更好地理解。这种基于注意力的机制也有利于许多视觉任务，包括语义分割、图像生成、视频分类以及目标检测 [13] [6] [38]，并且在相应任务也取得了强大性能。

在众多利用注意力机制的视觉任务中，最经典的工作是来自 Facebook AI 的 DETR [6]，相较于传统的目标检测，它消除了人为定义的锚点以及非极大值抑制模块，直接将目标检测问题转换为集合预测问题，具体来说，给定一组固定的可学习的查询对象，DETR 能够结合全局图像上下文与对象之间的关系来进行推理，直接输出最终的预测集，集合中的每个元素表示相应的查询对象是否表征为图片中的一个目标物体以及相应的目标框的几何坐标。

由于 DETR 的强大性能，LETR [33] 也将注意力机制引入直线检测任务之中。LETR 的思路很简单，就是将一条线段看作是一个目标框，即一个矢量化的线段可以被看作为目标框的一个对角线，随后按照标准的目标检测的框架来进行线段检测，它在直线检测中的两个数据集中取得了最先进的性能 [15] [18]。

LETR 的强大性能表明，基于注意力机制的方法能够直接学习和检测到图像中的几何结构。因此尤其适用于户型图重建，基于这个观察，近两年的户型图重建算法框架都是基于 Transformer 架构的。最新的两个工作分别是 HEAT [9] 以及 RoomFormer [36]。HEAT 提出了一种基于注意力的结构化重建模型，主要的贡献就是设计了一个新的户型图边缘分类架构。具体而言，它首先检测出户型图的各个角点，随后组合各个角点来构成一对对侯选边，在得到侯选边后，通过注意力机制将图像特征融合到每个侯选边中，并通过一个 Transformer 解码器来判断每个侯选边是否构成户型图的边缘。尽管该方法能够端到端地训练，但如果第一步检测出的端点存在问题，模型就无法重建出户型图中相应的边缘，从而在生成的户型图中产生缺口。为了避免 HEAT 由于两阶段产生的问题，RoomFormer 设计了一个单阶段的结构化预测问题：将每个房间看作一个顶点数量可变的多边形，将整个户型图看作一个数量可变的多边形集合序列，通过设计一个新的 Transformer 架构，RoomFormer 能够单阶段地并行地生成多个房间的多边形。特别地，他们根据房间顶点的排列顺序，将一个房间表征为一个按照排列顺序的顶点集合，在推理时，直接按照顺序依次将顶点连上即可生成预测的房间。如果网络生成的序列中顶点间的排列顺序存在问题，那么会影响后续的户型图重建。因此如何

用一个合适的形式来表征平面图是户型图重建方法的一个关键因素。

2.2 隐式神经表达

传统的信号表征通常是离散的，比如，图像可被参数化为一个个离散的像素，音频可被参数化为一系列离散的振幅样本，三维形状可被表征为离散的体素网格 [4]、点云 [27] 或者是三角网格 [19]。而隐式神经表征是对各种信号进行参数化的一种新方法，它能够将一个信号参数化为一个连续的函数，将信号中的一个坐标点（图像空间中的一个像素点），映射为该坐标点的任何性质（图像空间中相应像素点的 RGB 颜色）。然而这种连续的函数无法用一个数学形式来表达，因此，我们通常通过神经网络来对该函数进行近似 [30]。

2.2.1 三维任务

与显式表征相比，隐式神经表征对信号进行参数化所需的内存与空间分辨率无关，它能够在任意的空间分辨率下进行采样，因此，隐式神经表达能够用非常少量的参数来表征形状非常精细的细节，而体素网格则需要非常大的分辨率，内存开销随之上涨。基于上述的观察，隐式神经表征已被广泛用于三维物体形状、室内外场景的三维表面以及三维结构的外观建模 [35]。其中在三维物体重建上，隐式神经表达得到了非常大的进展，尽管表征的形式不同（occupancy [24], signed distance [26], unsigned distance [14]），他们都学习了一个连续的函数来预测查询点 (x, y, z) 与物体曲面之间的关系。一些比较早期的方法 [25] 是为每个对象单独学习一个独立的隐式神经表征，这种方法不具备泛化性能。因此解决方案为建立一个不同对象之间共享的一个连续函数空间。例如，在 DeepSDF [26] 中，作者提出利用一个编码器把每个形状单独编码成为一个隐向量，不同的形状共享相同的编码器，即共享相同的连续函数空间，随后对隐向量进行优化即可得到对应形状的距离函数。总的来说，隐式神经表达在三维任务中已经得到了广泛应用。

2.2.2 二维任务

尽管在隐式神经表征在三维任务中得到了广泛应用，但是它在图像空间上的应用还相对不足。目前的应用工作主要集中在图像表示上：在 Siren [30] 中，作者观察到，之前由多层感知机和 ReLU 参数化的隐式神经表征无法重建出图片中的精细细节。他们提出使用周期性的激活函数（正弦波）来替代 ReLU 函数，并证明了它能够以更高的质量来对图像进行建模。然而，上述的方法也是没有泛化性能，当不同图像共享隐式函数空间时，上述的方法就无法高保真地重建出复杂的图像。因此，在 LIIF [10] 中，作者提出了用于连续图像表征的局部隐式图像函数，其中每个图像都被表征为一个二维特征图，所有的图像共享一个解码器，它根据输入的坐标和相邻的特征向量输出相应坐标的 RGB 值。除了上述图像生成的工作之外，据我们了解，目前还未有在其他图像任务中使用隐式神经表达的工作。

在户型图矢量化的任务中，模型的输入是具有很多噪声的密度图，模型的输出是该图片中户型图的矢量化几何结构，包括各个房间的形状、墙壁以及顶点。而在三维重建的任务中，模型的输入是具有很多噪声的点云数据，输出是目标形状的高精度的三角网格。总体来看，两个任务是有共性的，户型图重建相较于其他图像任务，我们更加注重于利用模型学习图像中的几何结构，这个思想是与三维重建任务是一致的，二维的户型图可以被视作二维的形状，我

们可以利用重建三维形状的方法来重建二维户型图，在本次报告中，我们将在三维重建中取得广泛应用的隐式神经表达应用到二维的户型图矢量化的任务之中。

3 本文方法

3.1 户型图隐式表达

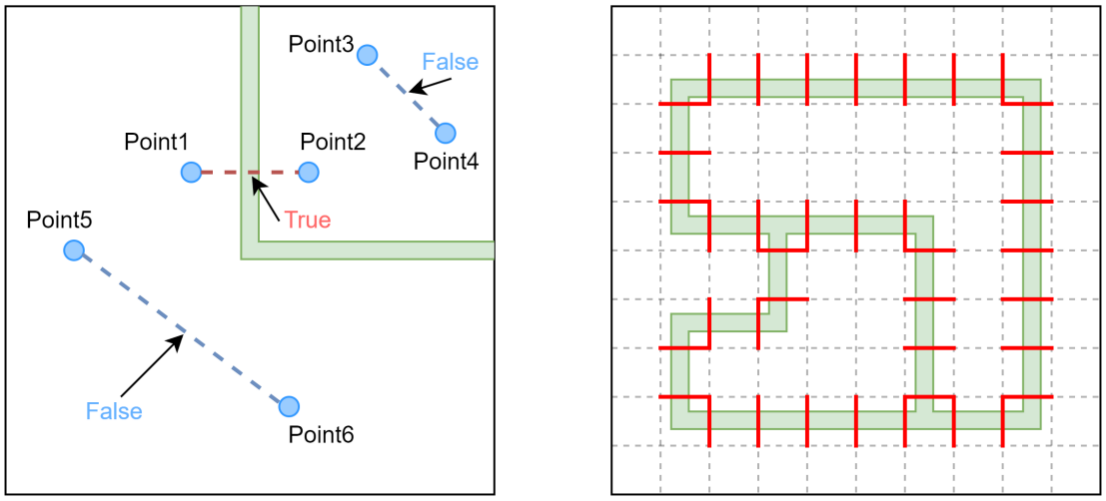

图 2. 隐式表达

在本次前沿技术报告中，我们提出了使用隐式表达的方法进行户型图矢量化，主要关注户型图二维平面中不同点之间的关系。具体来说，我们使用一个二进制标志来表示平面中的两个点是否被户型图的任意边隔开。假设户型图是由一系列组成边的连续的点构成集合：$S \subset R^2$。给定平面中的两个点 $p_1 \in R^2$ 和 $p_2 \in R^2$，我们能够得到一个由这两个点构成的线段 e，其中：$e(p_1, p_2) = \{p_1 + k * (p_2 - p_1) | k \in [0, 1]\}$。如果一条线段 e 与户型图的边相交，则至少存在一个点同时属于该线段和户型图，此时我们用一个二进制标志 b 来指示该线段是否与户型图相交，它被定义为如下形式：

$$b(p_1, p_2, S) = \begin{cases} 1, \exists p \in e(p_1, p_2), p \in S \\ 0, \text{otherwise} \end{cases}$$

图2左侧展示的是关于二进制标志的例子，其中（点 1，点 2）与户型图有交叉，而（点 3，点 4）和（点 5，点 6）与户型图没有交叉，因此（点 1，点 2）的二值码为 1，（点 3，点 4）和（点 5，点 6）的二值码为 0。在图 2 右侧，我们展示了一个完整户型图的隐式表征，其中二值码为 1 的点对用红色粗体表示。从图中可以看出，这种点对的形式尤其适合表征具有多层结构的户型图。需要注意的是，输入的点对并不局限于水平点对或竖直点对。

3.2 户型图显式重建

那么如何从这种隐式表达重建出户型图的显式结构呢？直观上，我们可以采用 Marching Cubes [23] 的方法来将上述隐式表征转化为矢量化结构。

3.2.1 Marching Cubes

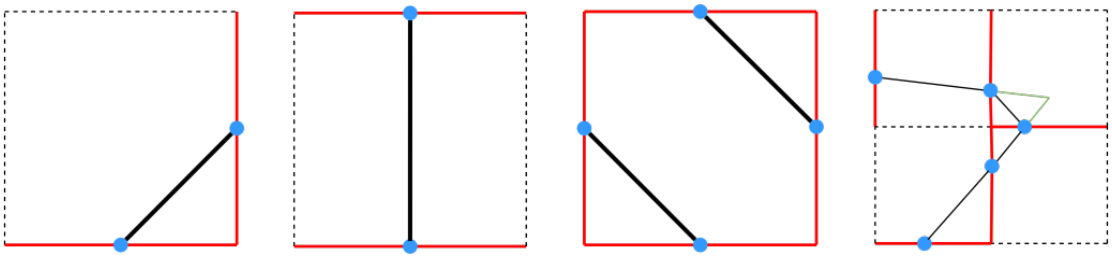

图 3. Marching Cubes

Marching Cubes 首先将整个平面分成 $N \times N$ 个正方形网格，其中每个格子大小为 $\frac{256}{N} \times \frac{256}{N}$。随后我们定位在那些与户型图交叉的正方形格子中，此时正方形格子的四条边构成了四个点对，我们计算这四个点对的二值码来判断这四条边与户型图的交叉情况。对于那些与户型图交叉的边，我们定位交叉边与户型图相交的具体位置。如图3所示，图中的红色粗线表示的是正方形中与户型图交叉的边，蓝色的点表示为交叉边与户型图相交的位置，在得到上述二者信息后，我们通过查找预先设定的表来将那些蓝色的点连接起来，这些蓝点的点对最终会构成户型图的一段边缘，我们将每个选定正方形的蓝点按上述规则进行连接最终构成了户型图的矢量化结构。

但是 Marching Cubes 会存在一个问题：由于查找表是固定的，连线的两个端点一定是在正方形网格上，因此对于右图中的这种绿色尖锐结构，Marching Cubes 算法无法进行重建，因此如果我们使用 Marching Cubes 算法对上述隐式表征进行重建，那么最终重建出来的各个角点的尖锐结构会遭到破坏。

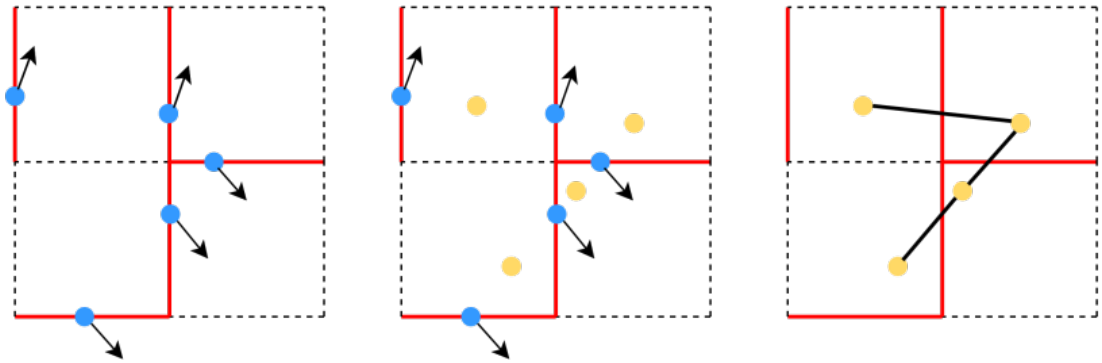

图 4. Dual Contouring

3.2.2 Dual Contouring

由于使用 Marching Cubes 算法进行显式重建会存在无法保留尖锐细节的问题，我们打算采用 Dual Contouring [21] 算法解决这个问题。与 Marching Cubes 相同，我们定位在那些与户型图交叉的正方形格子中，对于每个正方形格子，我们需要格子交叉边与户型图相交的位置（蓝点），并且还需要户型图在相应位置的法线（黑色箭头），随后我们利用这两种输入来建模二次误差函数并对其进行优化，随后在正方形格子内生成一个顶点（黄点），最后我们将

黄点相连即构成户型图的一段边缘。如图4所示，通过使用 Dual Contouring，我们能够重建出户型图中尖锐的顶点结构。

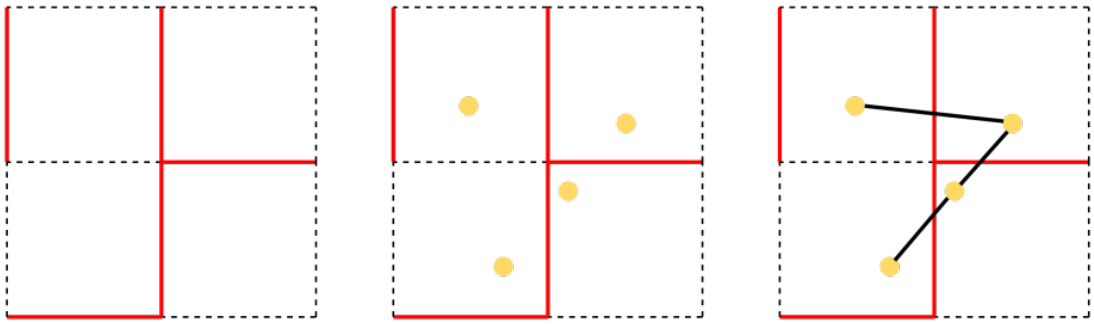

图 5. Neural Dual Contouring

相较于 Marching Cubes，Dual Contouring 的缺点是它需要户型图的梯度信息作为输入以计算每个正方形格子内的顶点坐标。为了解决这个问题，Neural Dual Contouring [11] 提出使用一个单独的神经网络来预测每个正方形格子内顶点的位置，从而避免了对于户型图梯度信息的需求。如图5所示，NDC 首先预测每个正方形格子的四条边是否与形状有交叉（红色实线），随后 NDC 使用另一个网络来预测每个正方形格子内顶点的位置（黄点），最后，对于每个相邻的黄点，如果它们之间存在红色实线，则把它们相连。通过使用 Neural Dual Contouring 的方式，我们能够避免对于户型图梯度的输入要求，从而能够显式地重建出户型图的尖锐细节。

综上所述，为了能够从我们的隐式表征中重建出户型图的显式结构，除了需要预测空间中的隐式场，即二维平面任意两点与户型图的交叉情况，我们还需要额外设计一个网络分支，用于预测二维平面中每个正方形格子内顶点的位置，在本项目中，我们设定网格的分辨率为 32×32，每个正方形格子的大小为 8×8。

4 复现细节

4.1 与已有开源代码对比

本项目的主要复现贡献在于把应用在三维的 NDC [11] 给迁移到二维的户型图矢量化任务中，因此在代码实现上不仅需要参考 NDC 的源代码，也需要借助二维户型图矢量化的工作的源码来帮助复现，这里我主要借用了 HEAT [9] 的代码来构建整个项目架构。

4.2 模型管线

如图1展示的是本次项目的总体模型管线。模型的输入是一组使用激光雷达扫描的粗糙的密度户型图，输出的是该图片内部对应的户型图矢量化结构。我们首先设计了一个网络架构，用于学习输入图片中户型图的隐式场以及预测正方形网格内顶点的位置。随后我们再通过一个后处理模块来将户型图的隐式表征进行显式重建得到最终的矢量化户型图。接下来的章节我将首先介绍项目的隐式网络的模型，随后介绍项目的后处理模块。

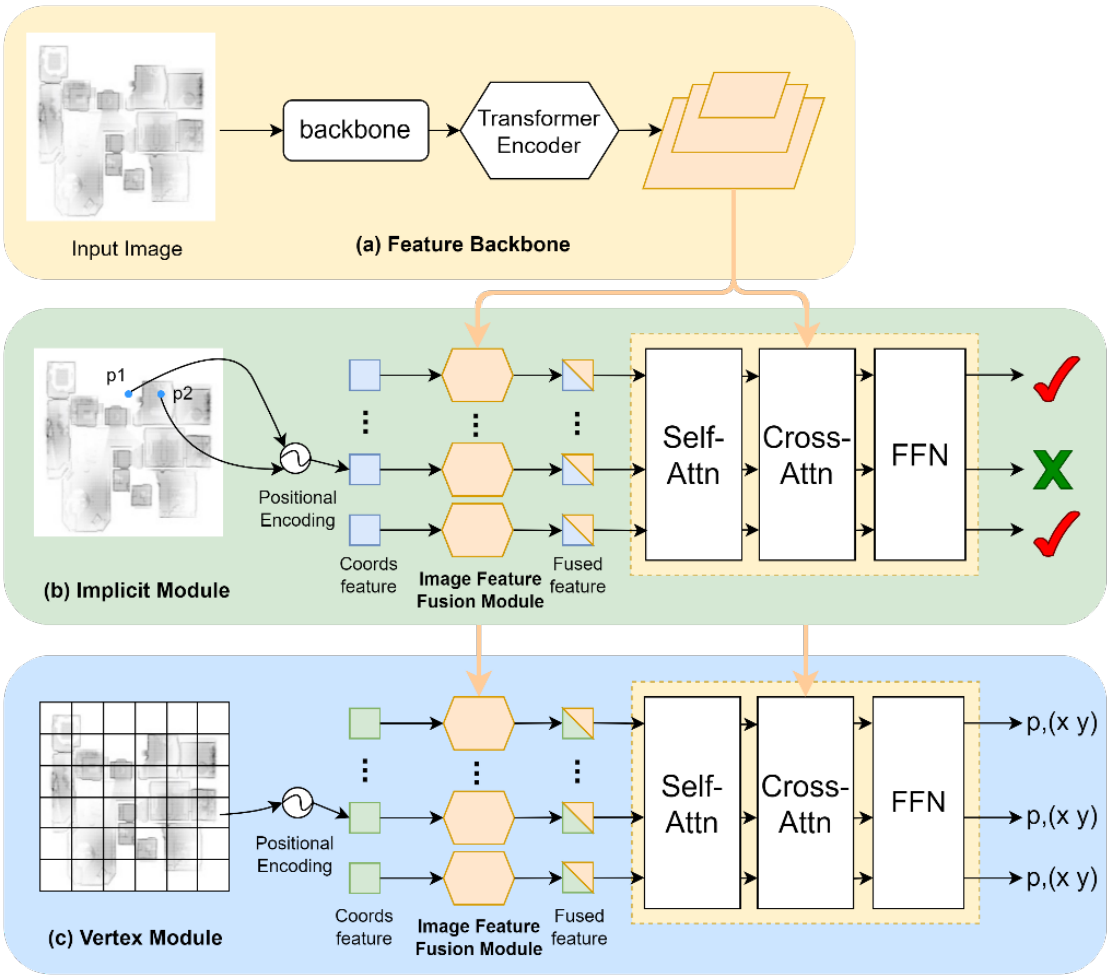

图 6. 网络架构

4.3 网络架构

图6展示的是本次项目的网络架构，它主要包含三个模块：(a) 一个 CNN 特征骨干网络用于提取输入图像的特征，随后接着一个 Transformer 的编码器用于增强前面提取的 CNN 特征，(b) 一个隐式场学习模块用于学习户型图的隐式场，(c) 一个顶点预测模块用于预测 32×32 的网格中每个正方形格子是否存在顶点，若存在顶点则预测相应格子内的顶点位置。在得到户型图的隐式场以及网格顶点后，我们通过一个后处理手段将户型图的隐式表征进行显式重建得到最终的矢量化户型图。

4.3.1 特征提取模块

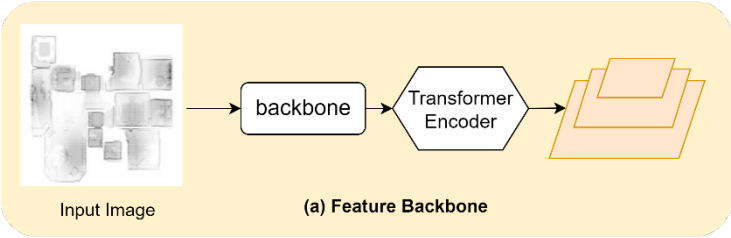

图 7. 特征提取模块

对于输入的一张包含噪声的户型图图片，我们首先使用一个 ResNet50 基干网络来从中提取语义信息。由于我们需要结合输入图片的局部和全局的图像特征来准确地预测二维空间中任意两个点之间的关系，我们需要利用基干网络的 L 个分辨率不同的特征图 $F_l \in R^{C\times H_l\times W_l}$ 来构成多尺度的金字塔特征图 $\{F_l\}_{l=1}^{L}$。为了进一步提炼特征，每一层特征图 $F_l \in R^{C\times H_l\times W_l}$ 会被展开成一个特征序列 $\{F'\}_l \in R^{C\times H_lW_l}$，同时，为了表征特征序列中每个特征元素之间的空间关系，我们还在特征序列中增加了位置编码 $E_l \in R^{C\times H_lW_l}$。我们将不同尺度的特征序列按顺序进行拼接得到一个多尺度的特征序列 $F' \in R^{C\times\sum_{l=1}^{L}H_lW_l}$，通过编码器中的自注意力模块，特征序列 F 中的每个特征元素能够自适应地与其他位置、其他尺度的特征元素进行交互，使得模型不仅能够关注图片局部区域的信息，在与不同位置、不同尺度的特征元素交互的过程中模型也能整合图像的全局信息。

4.3.2 隐式场学习模块

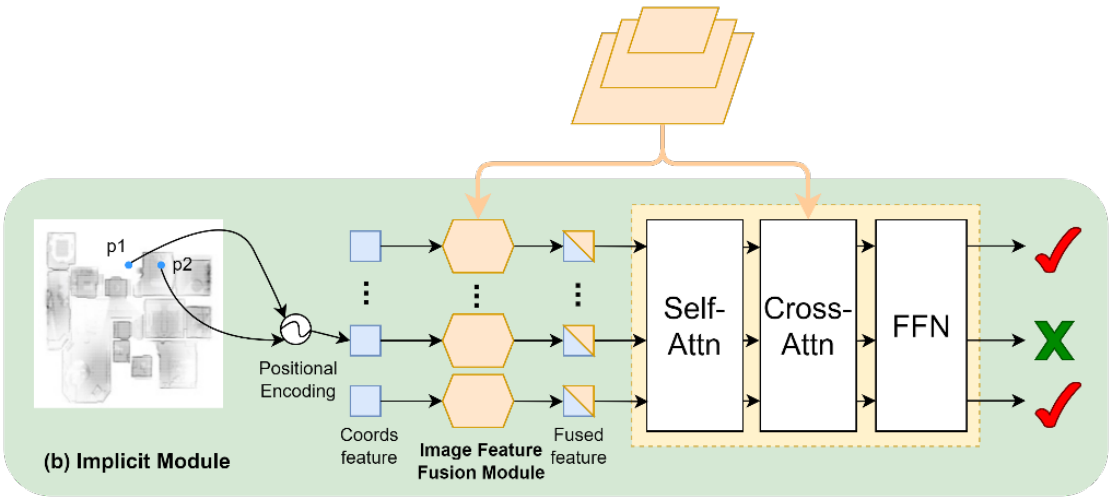

图 8. 隐式场学习模块

我们通过隐式场学习模块来预测输入户型图图片的隐式场。整个模块的输入是图片中任意采样的一个点对 $p = \{(x_1, y_1), (x_2, y_2)\}$，输出的是一个 $\{0, 1\}$ 的布尔值来表示该点对 p 是否与户型图交叉。整个隐式模块主要分为三个步骤：1) 采样点对特征初始化；2) 图像特征融合；3) 隐式解码。

给定输入图片中采样的任意一组点对 $p = \{(x_1, y_1), (x_2, y_2)\}$，该点对会作为一个节点输入至后续模块中，与 HEAT [9] 相同，我们首先使用三角位置编码 (trigonometric positional encoding) 来初始化该节点的位置特征 $f_{\text{coord}} \in R^C$

随后，我们需要从图像中采样该节点 f_{coord} 相应位置的图像信息 f_{img} 以利于后续学习。图像特征融合模块的示意图如图9所示，输入是一组节点的位置特征 $f_{\text{coord}} \in R^C$，输出是该节点融合图像信息 f_{img} 之后的融合特征 $f \in R^C$。我们在该模块中对输入特征 f_{coord} 注入相应采样位置附近的图像特征 f_{img}，使得输出特征 f 不仅包含该节点的位置信息，也能够表征该节点在图片中的语义信息。

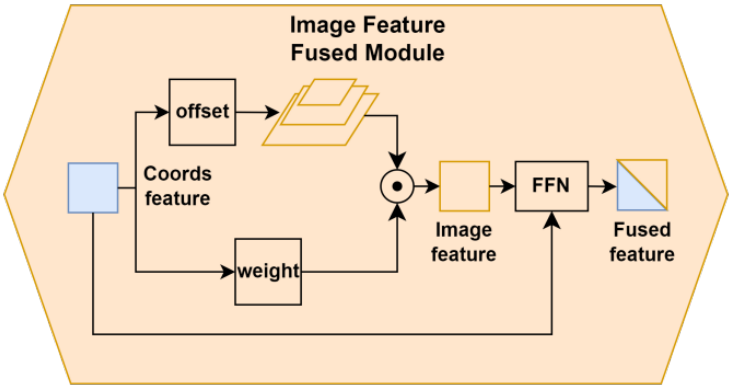

图 9. 图像特征融合

最后，我们将融合特征 f 输入至 Transformer 解码器中来判断输入点对 p 是否与户型图相交。如下图所示，解码器由 6 层构成，每一层包含了一个自注意力模块 (Self-Attention Module)，一个交叉注意力模块 (Cross-Attention Module) 以及一个前向反馈网络 (FFN)。每个解码器层的输入是由 Transformer 编码器增强的多尺度特征金字塔序列 $F \in R^{C \times \sum_{l=1}^{L} H_l W_l}$ 以及来自前一层解码器的输出特征 $f' \in R^C$。对于每个融合特征 f，它们输入至解码器之后首先通过自注意力模块互相交互，随后在交叉注意力模块中，每个融合特征将会关注并聚合相应位置的多尺度图像信息。最后，我们通过一个前馈网络 (FFN) 输出一个布尔值 $0, 1$ 来预测每个融合特征对应的点对 p 与户型图是否相交。

4.3.3 顶点预测模块

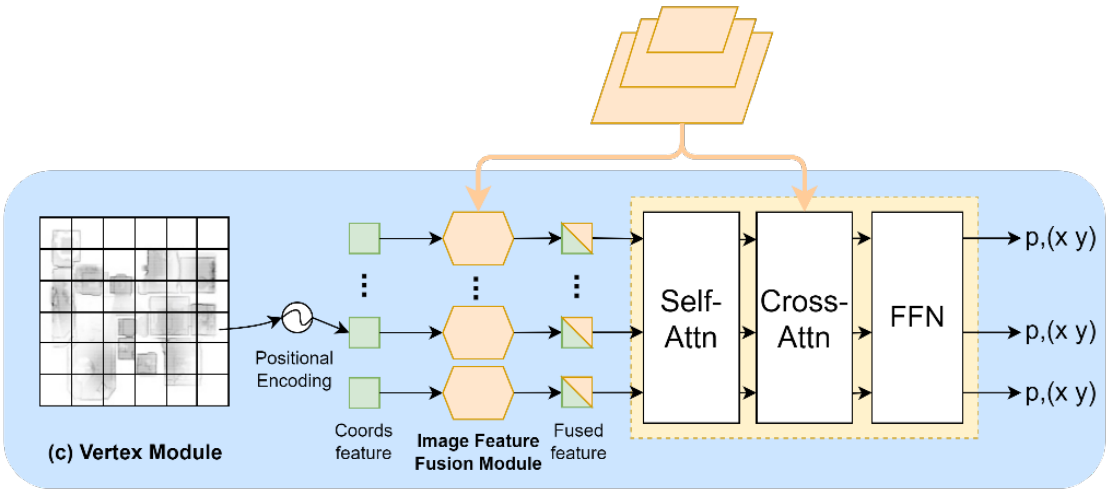

图 10. 顶点预测模块

在顶点预测模块，我们取输入图片的一个 8×8 大小的正方形格子，将格子内 64 个像素点的坐标进行拼接得到一个向量 $g \in R^{64\times 2}$，随后将其输入至一个多层感知机 (MLP) 得到对应正方形格子的位置特征 $f_{\text{coord}} \in R^2 56$，我们使用该位置特征 f_{coord} 作为对应的 Transformer 节点。

与隐式学习模块相同，每个正方形格子的位置特征 f_{coord} 也会输入至图像特征融合模块得到一个融合特征 $f \in R^2 56$，随后我们将融合特征 $f \in R^2 56$ 输入至顶点解码器，顶点解码器的结构与隐式解码器基本相同，在顶点解码器的最后一层前馈网络 (FFN) 中，对于每个输入特征 f，前馈网络会输出一个概率 $p \in [0, 1]$ 以及一个二维坐标 (x, y) 以表示该正方形格子出现顶点的概率以及对应顶点的二维坐标。

4.4 后处理模块

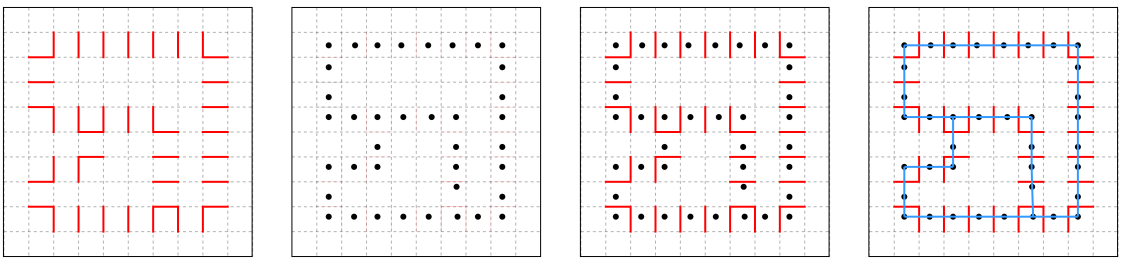

图 11. 后处理

在得到户型图的隐式场和预测顶点后，我们需要一个后处理模块来将户型图的隐式表征进行显式重建。具体来说，输入一张 256×256 大小的图片：

1. 在图像域中取 32×32 的离散网格，其中每个正方形格子大小为 8×8

2. 对于每一个正方形格子，分别将正方形各自的四条边输入至模型的隐式场预测分支中，预测正方形格子的四条边是否与户型图交叉

3. 模型的顶点预测分支预测正方形格子内顶点的位置

4. 对于预测的相邻两个顶点，若它们之间为交叉边，则将这两个顶点相连。

如图11所示，通过 Dual Contouring 我们能够一段一段地重建出户型图的矢量化结构。

4.5 损失函数

为了训练整个模型，我们需要一组输入图片 $\{X_i|i \in 1,...,N\}$ 以及相应输入图片的矢量化结构 $\{S_i|i \in 1,...,N\}$，其中 N 为训练图片的数量。首先针对第一个用于学习输入图片隐式场的隐式学习分支，对于每个训练样本，我们从中采样 M 组点对 $\{p_1, p_2\}$ 并根据标签的矢量化结构计算对应点对的二进制标志 $b(p_1, p_2) \in \{0, 1\}$；针对第二个用于学习网格顶点的顶点预测分支，我们使用 Dual Contouring 算法来计算输入图片中正方形网格对应的顶点 $V_{\text{gt}} \in \{c_i, (x_i, y_i)\}_{i=1}^{K}, c_i \in \{0, 1\}$ 作为标签，其中 K 为一张图片中正方形网格的数量。我们定义 $\hat{b}(p_1, p_2)$ 是隐式场预测分支的输出，$\{\hat{c_i}, (\hat{x_i}, \hat{y_i})\}_{i=1}^{K}$ 是顶点预测分支的输出。在训练过程中，我们使用了两个损失项：L_B 用于二值码预测，L_V 用于顶点预测；其中 L_B 的定义式如下：

$$L_B = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{M}\sum_{m=1}^{M}\text{BCE}(b_m(p_1, p_2), \hat{b}_m(p_1, p_2))$$

其中 BCE 为标准的二元交叉熵损失函数，L_V 的定义式如下：

$$L_V = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{K}\sum_{k=1}^{K}\{\text{BCE}(c_k, \hat{c_k}) + 1_{c_k=1}|(x_i, y_i) - (\hat{x}_i, \hat{y}_i)|\}$$

其中 $||$ 为 L1 损失，综上，整个模型的损失函数如下：

$$L = L_B + \alpha L_V$$

其中 α 为顶点预测分支损失的权重。

5 实验结果分析

5.1 数据集与评估指标

在本次研究报告的实验部分我们采用的数据集是 Structured3D [37] 数据集。Structured3D 数据集是一个大规模的户型图数据集，它一共包含了 3500 个房屋的平面图，其中涵盖曼哈顿和非曼哈顿布局。该数据集包含丰富的标注信息，其中包括 16 种房间类型以及门和窗的标注。在数据集划分上，我们遵循预设的 3000 个训练样本、250 个验证样本以及 250 个测试样本的划分。与以往工作相同 [8] [9] [36]，我们将数据集中的多视角 RGB-D 全景图转换为点云数据，并将点云数据沿着纵轴投影为 256×256 像素的密度图像。每个像素的密度值是投影到该像素的点的数量除以最大点数，因此密度图像中的每个像素值都被归一化为 [0,1] 之间。

5.2 对比方法

我选取了户型图重建中的 5 个前沿工作来进行比较：Floor-SP [8], MonteFloor [31], HAWP [34], LETR [33] 以及 HEAT [9]。其中 Floor-SP 和 MonteFloor 都首先使用 Mask R-CNN[6] 来分割房间，随后采用优化技术来重建户型图。HAWP 和 LETR 是线段检测中的通用方法，我们使用它们来进行户型图重建。HEAT 是基于 Transformer 架构的端到端的户型图重建方法，是目前户型图矢量化领域中表现性能最好的模型之一。

5.3 定量评估

表 1. 定性评估

Method	Room			Corner			Angle		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
Floor-SP [8]	89.0	88.0	88.0	81.0	73.0	76.0	80.0	72.0	75.0
MonteFloor [31]	95.6	**94.4**	95.0	**88.5**	77.2	82.5	**86.3**	75.4	**80.5**
HAWP [34]	77.7	87.6	82.3	65.8	77.0	70.9	59.9	69.7	64.4
LETR [33]	94.5	90.0	92.2	79.7	78.2	78.9	72.5	71.3	71.9
HEAT [9]	96.9	94.0	**95.4**	81.7	**83.2**	**82.5**	77.6	**79.0**	78.3
Ours	**97.3**	92.1	94.6	79.0	80.1	79.5	71.6	72.8	72.2

表1展示了本次实验中所有方法的量化指标。从表中可以看出，这次项目目前在房间的重建上效果优于 Floor-SP, HAWP 以及 LETR。尽管目前我们在户型图的角点检测上与最前沿的 HEAT 还有一小段的差距，其中 F1 分数相差 3，但在房间上的重建效果与它水平相当，其中 F1 分数只落后 0.8，并且在精度上优于 0.4。对于 HEAT 来说，当模型遗漏了重要的角点时，模型的边缘预测会出现错误从而导致无法重建出某个完整的房间。以上的实验结果表明我们项目中使用的隐式表征相较其他方法，其包含的结构信息更强，即使角点预测不如 HEAT 或其他方法，我们也能够通过空间中连续的隐式函数来将附近的角点连接起来从而重建出完整的户型图结构；因此如果后续我们能够优化角点的检测，那么模型在其他两个几何层面上的效果应当会有所提升。

5.4 定性评估

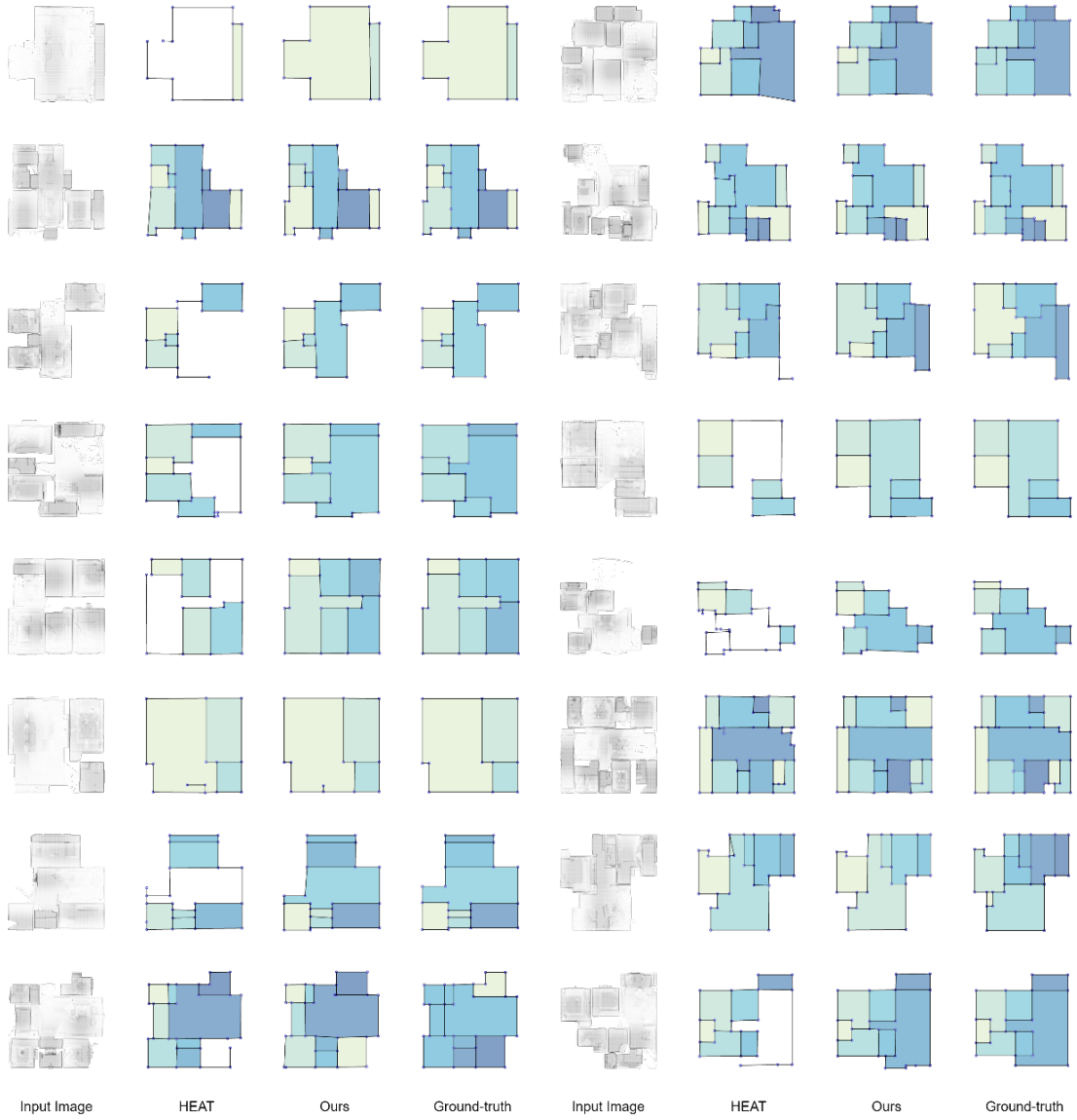

图 12. 定性评估

图12展示了我们方法的可视化重建效果，其中我们与 HEAT 和 Ground-Truth 结构进行对比。首先对于 HEAT 来说，它主要分为两个连续的步骤，首先检测角点，任意两个角点构成一组点对，随后 HEAT 利用一个解码器来判断给定一组点对，该点对是否构成户型图边缘。从图中可以看出，当 HEAT 第一步骤的角点没有预测出来时，那么对于它的解码器来说该区域就不存在候选的边缘，不存在候选的边缘也就意味着解码器无法预测出该区域的户型图边缘，因此 HEAT 无法重建相应区域的户型图结构。而对于我们的方法来说，即使某个区域的角点没有预测出来，但模型仍然学习到该区域中户型图的隐式表征，因此模型能够将连线与该区域相交的其他两个角点连接起来以重建这部分的户型图结构。

6 总结与展望

在本次前沿技术报告中，我们提出使用隐式神经表达的方法进行户型图矢量化，并设计了一个隐式网络用于学习输入户型图图片的隐式表征。实验结果表明，相较传统的户型图重建模型，我们的方法简单但更具有整体结构推理的能力。相信我们的工作能在户型图矢量化领域开辟新的路径，推动二维图像结构化重建的前沿阵地。

但目前我们的方法与其他前沿方法一样，当输入图片的某个区域完全没有密度或者密度很小，相应区域的结构基本上都无法很好地完成重建。目前来看，户型图矢量化的工作基本上都是以二维的密度图作为输入，基本上没有方法使用三维的点云数据作为输入来增强模型对于户型图的二维结构化感知；相较于其他所有方法，我们的优势在于我们能够使用同样的表征方式来重建三维的户型图表面，因此后续我将会延续我们的工作，通过借助三维的点云数据来增强模型在二维户型图上的重建能力以推动结构化重建领域的发展。

参考文献

[1] Antonio Adan and Daniel Huber. 3d reconstruction of interior wall surfaces under occlusion and clutter. In *2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pages 275–281. IEEE, 2011.

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

[3] Maarten Bassier and Maarten Vergauwen. Unsupervised reconstruction of building information modeling wall objects from point cloud data. *Automation in construction*, 120:103338, 2020.

[4] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[5] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[7] Rohan Chabra, Jan E Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed

3d reconstruction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pages 608–625. Springer, 2020.

[8] Jiacheng Chen, Chen Liu, Jiaye Wu, and Yasutaka Furukawa. Floor-sp: Inverse cad for floorplans by sequential room-wise shortest path. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2661–2670, 2019.

[9] Jiacheng Chen, Yiming Qian, and Yasutaka Furukawa. Heat: Holistic edge attention transformer for structured reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3866–3875, 2022.

[10] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8628–8638, 2021.

[11] Zhiqin Chen, Andrea Tagliasacchi, Thomas Funkhouser, and Hao Zhang. Neural dual contouring. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022.

[12] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019.

[13] Bowen Cheng, Anwesa Choudhuri, Ishan Misra, Alexander Kirillov, Rohit Girdhar, and Alexander G Schwing. Mask2former for video instance segmentation. *arXiv preprint arXiv:2112.10764*, 2021.

[14] Julian Chibane, Gerard Pons-Moll, et al. Neural unsigned distance fields for implicit function learning. *Advances in Neural Information Processing Systems*, 33:21638–21652, 2020.

[15] Patrick Denis, James H Elder, and Francisco J Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part II 10*, pages 197–210. Springer, 2008.

[16] Yasutaka Furukawa, Brian Curless, Steven M Seitz, and Richard Szeliski. Manhattan-world stereo. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1422–1429. IEEE, 2009.

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[18] Kun Huang, Yifan Wang, Zihan Zhou, Tianjiao Ding, Shenghua Gao, and Yi Ma. Learning to parse wireframes in images of man-made environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 626–635, 2018.

[19] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE international conference on computer vision*, pages 2307–2315, 2017.

[20] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6001–6010, 2020.

[21] Tao Ju, Frank Losasso, Scott Schaefer, and Joe Warren. Dual contouring of hermite data. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 339–346, 2002.

[22] Chen Liu, Jiaye Wu, and Yasutaka Furukawa. Floornet: A unified framework for floorplan reconstruction from 3d scans. In *Proceedings of the European conference on computer vision (ECCV)*, pages 201–217, 2018.

[23] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998.

[24] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019.

[25] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[26] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019.

[27] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[28] Xuebin Qin, Shida He, Xiucheng Yang, Masood Dehghan, Qiming Qin, and Jagersand Martin. Accurate outline extraction of individual building from very high-resolution optical images. *IEEE Geoscience and Remote Sensing Letters*, 15(11):1775–1779, 2018.

[29] Alexander Schrijver. *Theory of linear and integer programming*. John Wiley & Sons, 1998.

[30] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33:7462–7473, 2020.

[31] Sinisa Stekovic, Mahdi Rad, Friedrich Fraundorfer, and Vincent Lepetit. Montefloor: Extending mcts for reconstructing accurate large-scale floor plans. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16034–16043, 2021.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[33] Yifan Xu, Weijian Xu, David Cheung, and Zhuowen Tu. Line segment detection using transformers without edges. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4257–4266, 2021.

[34] Nan Xue, Tianfu Wu, Song Bai, Fudong Wang, Gui-Song Xia, Liangpei Zhang, and Philip HS Torr. Holistically-attracted wireframe parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2788–2797, 2020.

[35] Jianglong Ye, Yuntao Chen, Naiyan Wang, and Xiaolong Wang. Gifs: Neural implicit function for general shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12829–12839, 2022.

[36] Yuanwen Yue, Theodora Kontogianni, Konrad Schindler, and Francis Engelmann. Connecting the dots: Floorplan reconstruction using two-level queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 845–854, 2023.

[37] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 519–535. Springer, 2020.

[38] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.