布局生成变分自编码器网络

摘要

生成模型能够生成多种类型的布局（例如文档、用户界面或家具排列），是一种帮助设计流程并作为生成合成数据第一步的有用工具，同时也适用于其他任务。[1] 提出了一种基于变分自编码器（VAE）框架的变分 Transformer 网络（VTN），通过利用自注意力层的特性捕捉布局元素之间的高级关系，从而实现布局的边缘对齐和整体设计规则的学习，且无需显式监督。该方法生成的布局与训练数据高度相似，同时展现了多样性和吸引力。在公开可用的布局生成基准上，VTN 在多样性和感知质量方面达到了当前最先进的水平，并展示了其在文档布局检测流程中的潜力。

关键词：变分自编码器，生成模型，布局

1 引言

布局生成是场景设计、文档排版、图形生成以及数据合成等领域中的重要问题，其核心在于对布局中各元素的位置和相互关系进行建模，以生成具有合理结构和语义的布局。合理的布局不仅能够提升视觉表现效果，还对用户体验、信息传递和交互效率有显著影响，在用户界面设计、自然场景生成、出版物排版和广告设计等实际应用中具有广泛价值。然而，布局生成任务本身充满挑战，因为它需要同时处理离散类别（如文本、图像等元素类型）和连续变量（如元素的坐标和大小）之间的复杂关系。此外，布局生成还需要充分考虑元素之间的上下文依赖关系和全局一致性，从而生成既符合人类感知规则又具备创新性的布局设计。传统的布局生成方法多依赖于手工设计的规则（例如边距、对齐方式、元素数量限制等），虽然在特定场景中有效，但这些方法往往具有主观性强、适用性差、难以推广的问题，难以满足日益复杂的布局生成需求。近年来，随着深度学习技术的快速发展，基于生成模型（如生成对抗网络 GAN 和变分自编码器 VAE）的方法逐渐受到关注，成为解决布局生成问题的一种新思路。通过神经网络自动学习布局元素之间的复杂关系，这些方法展现出强大的建模能力，为布局生成领域带来了新的机遇。

2 相关工作

2.1 基于生成对抗网络（GAN）的方法

LayoutGAN [10] 首次将生成模型（GAN）应用于布局生成，通过生成器网络生成边界框注释，并使用卷积神经网络（CNN）作为判别器。同时引入了微分渲染模块将边界框转换为图像。尽管如此，该方法仅适用于单列文档，布局复杂度较低。

2.2 基于变分自动编码器（VAE）的方法

LayoutVAE [5] 提出了一种基于条件 VAE 的自回归模型，使用 LSTM 聚合时间信息，并通过第二个条件 VAE 建模类别计数分布。尽管其架构能处理一定复杂度的布局，但 LSTM 难以显式建模布局元素之间的关系。Neural Design Networks [9] 通过 VAE 和基于图卷积网络（GCN）的架构，学习布局元素间关系分布。最终的布局由另一个 GCN 生成。然而，该方法依赖于基于启发式的标签提取，不易泛化到不同的数据集。READ [12] 同样通过启发式提取布局元素关系，并使用递归神经网络（RvNN）驱动的 VAE 建模布局分布，但对启发式依赖较强，限制了其泛化性。Content-aware Generative Modeling [17] 利用 VAEGAN 生成条件于图像、关键词或属性的布局，重点在用户输入的额外条件指导下生成布局。与本研究不同的是，其依赖于额外的用户输入，而非无监督的布局生成。近期工作将 Transformer 与 VAE 相结合，利用注意力机制对高维数据建模。NLP 领域的相关研究已取得显著进展，但布局生成领域的应用尚需深入探索。

2.3 基于自监督学习的布局生成方法

Layout Generation with Self-attention [3] 通过自监督训练（如布局补全），结合基于 Transformer 的自回归解码器生成布局。尽管该方法生成效果较强，但需要优化大量超参数（如 Beam Width [15]），并且对布局分布的捕捉缺乏理论保证，主要依赖启发式规则。

2.4 特定任务导向的布局生成方法

一些研究专注于特定任务（如家具布局）[16] [4]，通过 CNN 估计物体位置的可能性。尽管这些方法在特定任务上表现良好，但其训练依赖特定的数据集（如已不可用的 SUNCG [14]），不适用于广泛布局生成。

	归纳偏置	无监督关系	任意大小	分布学习
LayoutGAN [10]	✓	✓		✓
LayoutVAE [5]	✗	✓	实践困难	✓
READ [12]	✓	✗	✓	✓
NDN [9]	✓	✗	✗	✓
Gupta et al. [12]	✓	✓		✗
Ours	✓	✓	✓	✓

表 1. 方法比较

3 本文方法

3.1 本文方法概述

该论文提出了一种基于变分自动编码器（VAE）和 Transformer 框架的变分 Transformer 网络（VTNs），专为布局生成任务设计。VTNs 的核心目标是通过捕捉布局元素之间的复杂全局关系，并处理可变数量的元素分布，实现高质量、高效率的布局生成。该论文结合 VAE

和 Transformer，利用注意力机制显式建模元素间的多对多关系，无需手工设计规则或启发式标注；通过潜变量建模布局分布，支持可变长度布局数据；设计了自回归解码器和非自回归解码器两种解码方式，分别适用于捕捉元素的顺序依赖和高效生成。相比传统基于 LSTM 的 VAE 模型，VTNs 能更高效地处理大规模复杂布局，同时在生成质量和计算效率之间取得平衡，为布局生成任务提供了一种灵活且高效的解决方案。

3.2 特征提取模块

依据变分自编码器原理，特征提取模块分为数据预处理，编码器和解码器三个模块。

在数据预处理阶段，每个布局由 l 个边界框组成，$l = \{x_1, x_2, ..., x_l\}$。其中每个边界框 x_i 包含位置，尺寸和类别三个属性，即 $x_i = [x, y, w, h, label]$。类别属性，如标题，文本或图片，使用单热向量 (one-hot) 编码表示；而位置与尺寸属性离散化为固定区间表示。此数据格式即包含离散特征（类别），也包含连续特征（位置和尺寸），是布局生成的基础。

编码器的任务是学习布局后验分布 $q_\phi(z \mid x)$，将布局 x 编码为潜在变量 z。编码器基于 Transformer 实现，其多头自注意力机制用于捕捉布局中所有元素之间的全局关系；前馈层用于增强每个元素的特征表达；潜变量分布用于将后验分布建模为多元正态分布。

解码器负责从潜在变量 z 和布局分布生成边界框。文章提到了两种变体，其中自回归解码器将潜变量 z 聚合为单一表征，用于捕捉全局特征，会逐步生成当前布局元素，基于先前生成的元素决定下一个元素。其更适合捕捉元素间的顺序依赖，但计算复杂度较高。而非自回归解码器使用未聚合的潜在变量，能之间生成完整布局元素，在保持元素间的多样性的同时会造成局部细节的丢失。

图 1. 模型示意图

3.3 损失函数定义

该论文的训练基于变分自动编码器（VAE）的优化框架，目标是通过最大化 ELBO（Evidence Lower Bound）[7] 实现高质量的布局生成。其中 ELBO 目标函数定义如下：

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{z \sim q_\phi(z|x)} \left[\log(p_\theta(x \mid z))\right] - \beta KL(q_\phi(z \mid x) \| p(z)) \tag{1}$$

目标函数由重构损失与 KL 散度组成。重构损失 $\log(p_\theta(x \mid z))$ 表示从潜变量 z 生成的布局 x' 与原布局 x 之间的匹配程度；使用交叉熵损失分别计算类别位置和尺寸。对于类别这种离散特征，使用单热 (one-hot) 向量表示后，与离散化后的位置和尺寸计算每一维度的分类误差。而 KL 散度 $KL(q_\phi(z \mid x) \| p(z))$ 用于度量后验分布 $q_\phi(z \mid x)$ 与先验分布 $p(z)$ 的差异。

调节因子 β 用于平衡重构损失和 KL 散度，在自回归解码器中，$\beta = 1$，更加重视 KL 散度用于捕捉复杂关系；在非自回归解码器中，$\beta = 0.5$，用于平衡计算效率和生成质量。同时，采用了指数调节策略，即在训练初期设置较小的 β，逐步增大以防止后验塌陷。

4 复现细节

4.1 与已有开源代码对比

查阅 paperwithcode 可知，截止至本文写作时，并未有相应开源代码。

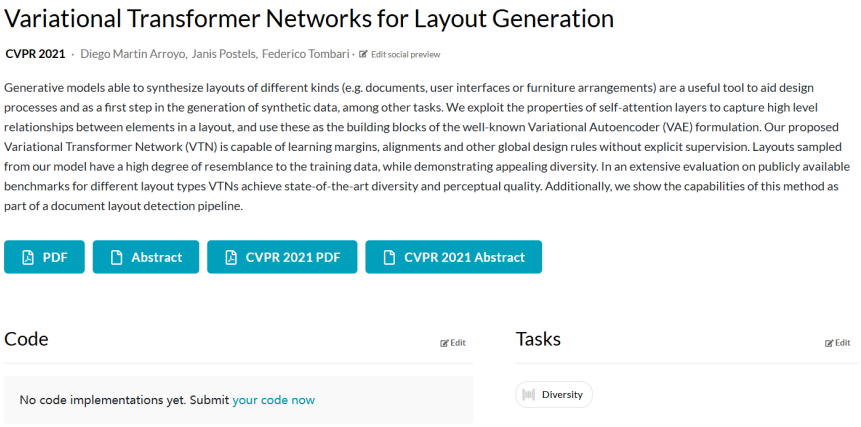

图 2. 查询结果

下面介绍主要复现工作。

4.1.1 数据获取与分析

本复现工作使用的是 RICO 数据集，主要结构如下图所示：

```
{
    .
    .
    .
    {
      "iconClass": "cart",
      "resource-id": "com.mobile.android.patriots:id/action_shop",
      "ancestors": [
        "android.support.v7.widget.AppCompatTextView",
        "android.widget.TextView",
        "android.view.View",
        "java.lang.Object"
      ],
      "clickable": true,
      "class": "android.support.v7.view.menu.ActionMenuItemView",
      "bounds": [
        1272,
        98,
        1440,
        266
        ],
      "componentLabel": "Icon"
    }
   .
   .
   .
   {
     "ancestors": [
       "android.widget.LinearLayout",
       "android.view.ViewGroup",
       "android.view.View",
       "java.lang.Object"
     ],
     "class": "android.support.design.widget.TabLayout$TabView",
     "bounds": [
       90,
       280,
       342,
       448
     ],
     "clickable": true,
     "componentLabel": "Text_Button",
     "textButtonClass": "all"
   },
   .
   .
   .
}
```


图 3. RICO 数据样例

在复现过程中，使用算法 1进行数据的获取。

Algorithm 1 extract_nodes

Input: `data`: JSON content as a dictionary or list

Output: `nodes`: A list of extracted node information

Initialize an empty list `nodes` **if** *`data` contains `bounds`* **then**
    Append `data['bounds']` to `node_info`
if *`data` contains `componentLabel`* **then**
    Add the index of `data['componentLabel']` in `_rico25_labels` to `node_info`
else
    Append the default value 0 to `node_info`
Append `node_info` to `nodes`
if *`data` contains child nodes in `children`* **then**
    foreach *`child` in `data['children']`* **do**
        Recursively call `extract_nodes(child)` Extend the result into `nodes`
return `nodes`

4.1.2 基础结构

本文的基础结构采用了基于带有多头注意力机制的 Transformer 的变分自动编码器(VAE)框架。模型将布局中每个节点的位置和大小归一化处理后视为连续值，并将节点的类别视为

离散值。对于位置和大小的预测问题，模型将其建模为回归问题，通过连续值回归精确预测节点的位置和尺寸；而对于节点的类别，则建模为分类问题，通过分类模型预测节点所属的类别种类。这种设计充分利用了 VAE 的潜变量表达能力以及 Transformer 的注意力机制，能够有效建模布局节点间的全局关系，同时结合分类和回归任务处理布局数据中的离散和连续属性，从而实现高效且精准的布局生成。

4.2 实验环境搭建

实验环境如下：在实现细节方面，使用 Adam [6] 优化器进行训练，学习率固定为 10^{-3}，

设备名	厂商	存储空间
显卡	GeForce P6000	24GB
内存	未知	251GB
硬盘	未知	4TB

表 2. 基础设备环境

在每次前向传播时，均将布局的位置和尺寸反归一化到整数后重新进行归一化。每次训练 100 个 epoch。

4.3 创新点

在一般的布局生成任务中，位置和尺寸通常被处理为离散属性，通过将布局区域划分为网格或固定区间进行离散化，从而简化计算和优化。然而，本文提出了一种新的尝试，即将位置和尺寸作为连续属性进行建模，在布局生成过程中直接对其进行连续值预测。这种方法在生成阶段通过反归一化将预测的连续值映射回实际的布局坐标和尺寸范围，随后通过平滑函数 (smooth_round) 在对齐到相应的整数的同时保留梯度，以保证布局的一致性和可用性。该创新点的优势在于能够更精确地保留布局元素的几何特性，避免因过度离散化而导致的精度损失；同时，平滑函数的使用也不会造成梯度消失。此外，这种连续建模的方式为布局生成提供了更大的灵活性，适用于更复杂、更多样化的布局生成任务。

5 实验结果分析

5.1 数据集分析

实验评估使用了多种公开可用的数据集，这些数据集覆盖了文档布局、自然场景、家具排列和移动端 UI 等领域，包括以下几个关键数据集：

PubLayNet [18]：包含 33 万份科学文档的机器标注样本，类别包括文本、标题、图像、列表和表格。

RICO [2]：包含 91,000 个移动应用用户界面设计样本，有 27 种元素类别，如按钮、工具栏和列表项。

COCO-Stuff [11]：包含 10 万张自然场景图片，涵盖 80 个“事物”和 91 个“场景”的类别，

剔除了较小的边界框和“is_crowd”标签的实例。
SUN RGB-D [13]：包含 10,000 个样本的场景理解数据集，标注了家居物品的语义区域。

5.2 评估方法

实验的评估方法主要从两个高层次的维度展开：感知质量和多样性。感知质量的评估注重布局的视觉效果和对齐程度，这在很大程度上是主观的，不同的数据集可能对感知质量的定义和要求各不相同。因此，很难定义一个单一的指标来全面衡量布局的感知质量。为了更加细致地评估，实验采用了多个独立的指标，分别针对感知质量或多样性展开分析。在感知质量的评估中，实验引入了对齐和重叠指标，这主要是针对像 PubLayNet 和 RICO 这样具有严格定义边界框的布局数据集。通过计算边界框之间的总重叠面积占整体页面面积的比值（Overlap Index）以及元素之间的平均交并比（IoU），来衡量生成布局的对齐质量。此外，还使用了一种基于对齐损失的度量方法，该方法参考了 [8] 的工作，进一步量化了布局元素之间的对齐程度。在多样性的评估中，实验采用了 DocSim 度量的独特匹配和 Wasserstein 距离两种方法。DocSim 度量通过统计生成布局与真实布局之间的独特匹配数量来衡量多样性，这一指标不仅反映了多样性，同时也间接表明了感知质量。为了更全面地评估布局分布的多样性，实验还计算了生成数据与真实数据分布之间的 Wasserstein 距离。具体来说，通过对两种边缘分布（类别分布和边界框分布）分别进行计算，将类别分布视为离散变量，而边界框分布则采用连续变量（包括中心点坐标、宽度和高度）表示。在实际操作中，这些距离通过有限的样本集估算得出。

5.3 定量分析

实验中通过重现已有方法（如 LayoutGAN 和 LayoutVAE）的结果，与当前方法进行了对比。在 PubLayNet 和 RICO 数据集上，与现有的对齐指标相比，新方法表现出较好的感知质量和对齐效果。在 RICO 数据集的 NDN（非显式约束）实验中，展示了新方法在无监督关系发现上的优越性。

	IoU	**Overlap**	**Alignment**	**W class ↓**	**W bbox ↓**	**# unique matches ↑**
LayoutVAE [5]	0.171	0.321	0.472	-	0.045	241
Gupta et al. [12]	0.039	0.006	0.361	**0.018**	0.012	546
Ours (autoregressive)	0.031	0.017	0.347	0.022	0.012	**697**
Real data	0.048	0.007	0.353	-	-	-

表 3. 定量分析

5.4 定性分析

在 PubLayNet 和 RICO 数据集的布局生成任务上，新方法能够更好地捕获元素间的距离和位置关系。对比现有方法，特别是 LayoutVAE，新方法在处理大元素数量的布局时优势显著。

图 4. 定性分析

5.5 复现结果

图中以蓝色线条表示原始布局 (Original)，以红色虚线表示重建布局 (Reconstructed)。

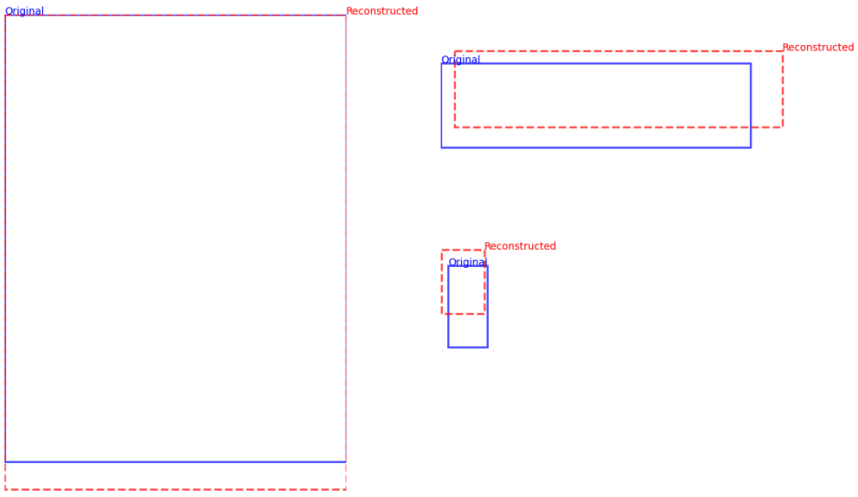

图 5. 复现结果

在对齐精度方面，可以看出重建的矩形位置与原始矩形的偏差较小，说明模型能够较为准确地预测布局元素的位置。然而，某些情况下（如右下角的小型矩形）存在明显的偏移，可能与模型对小尺寸布局元素的处理精度有关；在尺寸比例方面，对于左侧的大型矩形，重建

布局能够很好地保持原始矩形的尺寸比例，这说明模型在处理大尺寸元素时具有较好的拟合能力。但对于右侧的小型矩形，部分重建矩形的尺寸比例与原始矩形存在一定差异，可能是由于模型在小尺寸范围内的精度不足。

6 总结与展望

6.1 总结

本文提出了一种基于变分自动编码器（VAE）和 Transformer 框架的变分 Transformer 网络（VTNs），旨在解决布局生成任务中的关键问题。模型充分利用了 Transformer 的多头注意力机制，有效捕捉布局元素之间的全局关系，而无需依赖启发式标注或人工规则。通过构建以多元正态分布为核心的潜变量模型，本文方法能够适应布局数据中元素数量的变化，并通过两种解码器（自回归解码器和非自回归解码器）的设计，在生成能力与计算效率之间取得平衡。此外，为了提升生成质量，本文优化了传统 VAE 框架中的 ELBO 目标函数，采用 KL 散度调节策略避免后验塌陷，并通过离散化的方法对布局元素的类别、位置和尺寸进行建模。实验结果表明，本文方法在多种公开数据集上实现了布局生成性能的显著提升，生成的布局不仅符合元素的语义关系，还展现出了良好的分布拟合能力和全局一致性。在复现时，我创新性地将布局元素的位置和尺寸建模为连续属性，而非传统的离散化处理，从而减少了离散化带来的信息损失，使生成结果更加精确和细致。同时，模型在解码阶段通过反归一化和对齐操作确保布局的实际可用性，实现了连续值和整数值的平滑转换。

6.2 展望

尽管本文在布局生成领域取得了显著的进展，但仍有一些未解问题和潜在方向值得进一步研究和探索。首先，在处理极大规模的布局数据时，如何进一步优化模型的计算效率和内存需求是一个重要课题。尽管本文已经通过优化 Transformer 结构参数减少了计算复杂度，但在面对实际应用中超大规模的布局数据时，仍需探索更高效的模型结构或分布式计算方法。

其次，本文的方法主要针对二维布局生成任务，而在三维布局（如室内家具设计、三维场景建模等）或动态布局（如视频内容中的多帧布局生成）中，布局元素之间的关系会更加复杂。如何将本文方法扩展到这些更高维度、更动态的布局场景，是一个具有重要意义的研究方向。

此外，多模态布局生成任务也是未来值得关注的热点。例如，在同时包含文本、图像和其他类型元素的复杂场景中，如何建模不同模态元素之间的相互关系并生成协调一致的布局仍需进一步研究。可以尝试结合其他生成模型（如扩散模型或 GAN）与 Transformer 结构，进一步提升布局生成的多样性与细节表现。

最后，在实际应用中，用户需求往往是多样化的，因此如何在生成过程中加入用户偏好或特定约束（如内容对齐、分组布局等），使生成结果更加贴合用户需求，也是未来研究的重要方向之一。为此，可以考虑引入基于人机交互的生成方法或通过强化学习探索布局生成中的约束优化问题。

总的来说，未来的研究可以通过进一步优化模型性能、拓展应用场景、增强生成的可控性和多样性，为布局生成任务提供更强大的技术支持，并推动其在智能设计、文档排版、场

景建模等领域的广泛应用。

参考文献

[1] Diego Martin Arroyo, Janis Postels, and Federico Tombari. Variational transformer networks for layout generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13642–13652, 2021.

[2] Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual Symposium on User Interface Software and Technology*, UIST '17, 2017.

[3] Kamal Gupta, Alessandro Achille, Justin Lazarow, Larry Davis, Vijay Mahadevan, and Abhinav Shrivastava. Layout Generation and Completion with Self-attention. *arXiv e-prints*, page arXiv:2006.14615, June 2020.

[4] Paul Henderson, Kartic Subr, and Vittorio Ferrari. Automatic Generation of Constrained Furniture Layouts. *arXiv e-prints*, page arXiv:1711.10939, November 2017.

[5] Akash Abdu Jyothi, Thibaut Durand, Jiawei He, Leonid Sigal, and Greg Mori. Layoutvae: Stochastic scene layout generation from a label set. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 9894–9903. IEEE, 2019.

[6] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[7] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[8] Hsin-Ying Lee, Lu Jiang, Irfan Essa, Phuong B Le, Haifeng Gong, Ming-Hsuan Yang, and Weilong Yang. Neural design network: Graphic layout generation with constraints, 2020.

[9] Hsin-Ying Lee, Weilong Yang, Lu Jiang, Madison Le, Irfan Essa, Haifeng Gong, and Ming-Hsuan Yang. Neural design network: Graphic layout generation with constraints. In *Proceedings of European Conference on Computer Vision (ECCV)*, August 2020.

[10] Jianan Li, Jimei Yang, Aaron Hertzmann, Jianming Zhang, and Tingfa Xu. Layoutgan: Generating graphic layouts with wireframe discriminators. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[11] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014.

[12] Akshay Gadi Patil, Omri Ben-Eliezer, Or Perel, and Hadar Averbuch-Elor. READ: recursive autoencoders for document layout generation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 2316–2325. IEEE, 2020.

[13] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 567–576. IEEE Computer Society, 2015.

[14] Shuran Song, Fisher Yu, Andy Zeng, Angel X. Chang, Manolis Savva, and Thomas A. Funkhouser. Semantic scene completion from a single depth image. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 190–198. IEEE Computer Society, 2017.

[15] Volker Steinbiss, Bach-Hiep Tran, and Hermann Ney. Improvements in beam search. In *The 3rd International Conference on Spoken Language Processing, ICSLP 1994, Yokohama, Japan, September 18-22, 1994*. ISCA, 1994.

[16] Kai Wang, Manolis Savva, Angel X Chang, and Daniel Ritchie. Deep convolutional priors for indoor scene synthesis. *ACM Transactions on Graphics (TOG)*, 37(4):70, 2018.

[17] Ying Cao Xinru Zheng, Xiaotian Qiao and Rynson W.H. Lau. Content-aware generative modeling of graphic design layouts. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2019)*, 38, 2019.

[18] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. Publaynet: largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022. IEEE, Sep. 2019.