RVLM：更少的令牌、更优的分辨率扩展、更精细的空间定位

Savya Khosla[1,2] **Chris Donjoo Kim**[1] **Alex Schwing**[1]
Derek Hoiem[1] **Christopher Clark**[2] **Sangho Lee**[2]
[1]伊利诺伊大学厄巴纳-香槟分校 [2]艾伦人工智能研究所

arXiv:2611.29146v1 [cs.CV] 22 Nov 2026

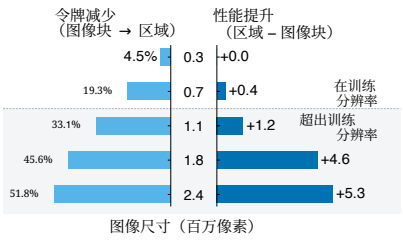
摘要

视觉语言模型（VLMs）使用固定的空间网格对图像进行词元化，因此视觉词元数量会随着像素面积增长，而不是随场景内容增长。我们提出根据视觉外观对相邻图像块进行分组，使词元能够表示连贯的图像区域，而非固定的空间窗口。我们引入 RVLM，这是一种经过训练、使用区域而非图像块作为视觉单元的视觉语言模型。通过分析通用视觉语言模型中的区域构建，我们发现，区域必须覆盖完整图像，在端到端训练期间保持空间连通性，并保留小字等精细结构，而语义分组的作用很小。因此，RVLM 使用无需训练的 Felzenszwalb-Huttenlocher 分割对图像进行划分，将每个 ViT 图像块分配到一个区域，并在每个区域内对图像块进行池化，形成可变长度的视觉令牌序列。图像块到区域的分配关系得以保留，从而通过由粗到细解码支持具有空间精度的预测。在涵盖视觉问答、指点、计数、分割和深度估计的各项任务中，与采用相同训练方式的图像块基线相比，RVLM 在视觉词元数量减少 21.5% 的同时，示例加权任务性能达到相当水平或有所提升。随着图像分辨率提高，这一优势不断扩大；在最高分辨率设置下，视觉词元数量减少 51.8%，性能提升 5.3 个百分点。在空间要求较高的任务上，性能增益最大，而此前的令牌缩减方法在这些任务上的性能下降也最为明显。RVLM 不增加任何模型参数，并将推理吞吐量最高提升 35.9%。在未见过的病理学图像和卫星图像上，以及在完全没有视频训练的视频基准测试中，RVLM 使用 2–3× 更少的视觉词元，同时性能与图像块基线相当。这些结果表明，区域可作为基于图像块的视觉表征的可扩展替代方案，同时保留基于视觉定位的预测所需的空间精度。

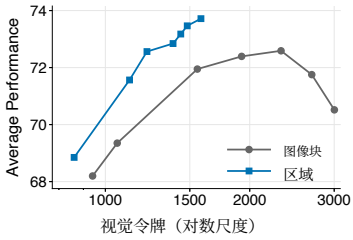
1 引言

将图像标记化为均匀网格中的图块，是视觉语言模型（VLM）的标准设计选择。然而，图像块标记化将令牌数量与像素数量挂钩，而非场景内容。因此，单个物体的高分辨率图像所需的令牌数，比低分辨率的杂乱场景图像更多；文档的空白边距也可能仅仅因为占据了更大面积，而消耗比文本更多的令牌。这种不考虑内容的分配方式浪费了视觉令牌预算，并限制了基于图像块的视觉语言模型在固定令牌预算下可处理的最高图像分辨率。

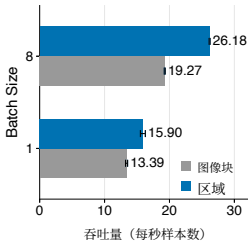
现有模型通过固定空间池化（Chen 等，2023a；2025b；Wang 等，2024b），或通过剪枝与合并图像块令牌（Bolya 等，2023；Chen 等，2024a；Yang 等，2025b）来减少视觉词元数量。固定池化忽略图像内容，而剪枝与合并可能会减少空间覆盖，或模糊视觉令牌与图像位置之间的对应关系，从而影响空间定位。内容自适应令牌化提供了另一种选择：让视觉单元跟随图像内容，而非固定网格（Lee 等，2026；Chen 等，2025a；Khosla 等人，2026）。然而，现有方法依赖学习特征或分割模型，这会增加计算成本，或使区域构建对视觉语言模型训练变得敏感（第 4.4 节）。我们发现，区域令牌的效果与其说取决于语义分组，不如说取决于区域是否覆盖整个



（a）随着图像尺寸增大，区域比图像块更具尺度优势。RVLM 使用更少的令牌，却能达到或超过采用相同训练方式的图像块基线。在测试时进行分辨率缩放时，效果提升最为显著（表10）。



（b）在给定的令牌预算下，RVLM 能够处理分辨率更高的图像，比基于图像块的模型表现更佳。在对分辨率敏感的任务上，提升最为显著。



（c）RVLM 更短的序列提升了推理吞吐量，批次大小从 1 增至 8 时，提升幅度也从 19% 增至 36%（表3）。

图1：**区域令牌随着图像分辨率提高而更具扩展性，同时提升推理效率**。详情请见第 4.2 节。

图像在端到端训练期间保持空间连通，并保留小字等精细结构。

我们提出了 RVLM，这是一种经过训练、使用区域而非图像块作为视觉单元的视觉语言模型。与固定空间池化不同，我们根据相邻图像块的视觉外观将其分组，形成更有意义的单元来池化视觉信息。我们使用无需训练的 Felzenszwalb-Huttenlocher 分割（Felzenszwalb 和 Huttenlocher，2004年）构建此类区域，将每个 ViT 图像块分配到一个区域，并在区域内池化图像块特征，以生成视觉令牌序列。RVLM 保留图像块到区域的分配关系，以支持空间落地预测。对于指点任务，它采用由粗到细解码来预测位置：首先预测相关区域，然后预测该区域内的图像块，最后预测图像块内的位置。与图像块基线相比，令牌化不会增加参数；即使单个词元代表图像的很大一部分，由粗到细解码也能保持空间精度。

我们在 35 项任务上评估 RVLM，并将其与采用匹配图像块的视觉语言模型以及规模相近的开放权重模型进行比较。结果表明，区域在保留基于视觉定位的预测所需的空间对应关系的同时，提升了分辨率缩放能力和推理效率（图1）。我们还研究了有效区域构建所需的特性（第 4.4 节）。

总而言之，我们的主要贡献如下：

1. **通用视觉语言模型中的区域令牌为何有效**。通过分析区域构建策略，我们发现，区域必须覆盖整幅图像，在端到端训练期间保持空间连通，并保留小字等精细结构，而语义分组作用不大。未覆盖所有像素的分割器无法支持对这些位置进行指点；而从视觉语言模型自身特征生成的区域则会随着编码器训练而变得零散。相比之下，学习式分割器以及基于独立冻结编码器进行的聚类，表现均不优于经典的 Felzenszwalb-Huttenlocher 分割方法；后者速度快、不增加模型参数，而且比 SLIC 更能保留小字。

2. **基于区域令牌构建的高效视觉语言模型**。我们提出 RVLM，这是一种以内容自适应的基于区域的令牌化取代固定空间池化的视觉语言模型。RVLM 保留了图像块到区域的对应关系，因此能够在减少视觉令牌的同时，达到或提升定位性能。与训练方式完全相同的基于图像块的视觉语言模型相比，RVLM 平均减少了 21.5% 的视觉令牌，推理吞吐量最高提升 35.9%。它还能更有效地扩展到更高的测试时分辨率：在测试的最高分辨率下，令牌减少 51.8%，性能提升 5.3 个百分点。对于计数、分割和深度等空间任务，其优势最为显著；以往减少令牌数量的方法在这些任务上的性能会大幅下降。要让区域令牌适用于密集输出，必须针对不规则区域调整池化、位置编码以及指点和分割解码器。若直接用区域令牌替换图像块令牌，PixMo-Point 上由粗到细指向的得分会骤降至 10.0；相比之下，图像块令牌的得分为 84.4，而采用我们的方法则达到 86.6（表 12）。

2 相关工作

视觉语言模型中的视觉令牌化。视觉语言模型使用视觉变换器将图像编码为图像块特征，再通过连接器将其映射到大语言模型嵌入空间（Liu 等，2023；Chen 等，2023b）。由于图像分辨率越高，图像块序列增长越快，连接器通常会减少传递给大语言模型的视觉令牌数量。一种方法采用固定数量的学习型查询（Li 等，2023；Alayrac 等，2022；Bai 等，2023），使令牌数量与图像尺寸脱钩，但会削弱令牌与图像位置之间的对应关系。更常见的方法是在固定的空间网格上对图像块特征进行池化。现有方法包括像素重排（Chen 等，2024b；Laurenc̦on 等，2024；Marafioti 等，2025）、自适应平均池化（Yao 等，2024）、卷积或可变形投影器（Cha 等，2024；Chu 等，2024；Ye 等，2023），以及对局部图像块特征进行交叉注意力计算（Li 等，2025；Deitke 等，2025；Clark 等，2026b）。尽管这些设计聚合特征的方式各不相同，但其空间池化单元由位置而非图像内容决定。增大池化窗口可以减少令牌数量，但也会使视觉表示变得更粗糙，可能损害定位性能（Marafioti 等，2025）。相比之下，RVLM 根据图像内容构建池化单元，生成空间范围各异的区域令牌，同时保留明确的图像块到令牌对应关系，以实现空间定位。

内容自适应令牌化。另一类研究通过对视觉编码器的图像块特征进行语义聚类，构建内容自适应视觉词元（Jin 等，2024；Wu 等，2025；Lee 等，2026；Mao 等，2026；Khosla 等，2025；2026）。然而，这些方法依赖图像块特征的空间保真度；我们发现，端到端视觉语言模型训练会导致区域分配不稳定，产生空间上不连通的组（第 4.4 节）。另一些自适应长度的令牌化器将图像表示为可变长度的潜在序列（Duggal 等，2025；Yan 等，2025），但未明确建立词元与图像位置之间的对应关系，因此限制了它们在空间定位中的应用。相比之下，RVLM 直接从图像像素构建空间连通区域，同时保留图像块到区域的对应关系，从而实现精确的空间定位。此前也有研究探索在视觉编码器、图像自编码器和视觉语言模型中采用基于区域的视觉词元化（Lew 等，2025；Wang 等，2024a；Ma 等，2024；Yang 和 Gong，2025）。与大多数基于区域的表示方法类似（Shlapentokh-Rothman 等，2024；Khosla 等，2025；2026；Chen 等，2025a；Lee 等，2026），RVLM 根据预训练图像块编码器的特征生成区域令牌，在沿用强大编码器的同时，改变大语言模型进行推理的单位。与我们最接近的是 Chen 等（2025a），他们研究了基于 SAM 的区域词元化，并提出一种轻量级的可学习边界检测器，随后使用分水岭分割确保完整覆盖图像。然而，可学习的检测器和分割器会在区域构建过程中带来额外的计算和内存开销。RVLM 则采用免训练分割，仅依赖输入图像，因此无需增加模型参数，便可在前向传播之前获得区域。在我们的实现中，分割在数据加载期间由 CPU 执行；这一工程设计使其不占用加速器，并能与模型执行并行进行。

事后词元压缩。另一类研究通过剪除不太重要的图块来减少视觉词元（Zhang 等，2025a；Shang 等，2025；Chen 等，2024a；Zhang 等，2025c；Xing 等，2025；Yang 等，2025b），或合并冗余特征（Bolya 等，2023；Wen 等，2025b；Alvar 等，2025；Zhang 等，2025b）。尽管这些方法对图像级理解有效，但会降低空间定位任务上的性能（Wen 等，2025a；Chien 等，2025）。剪枝会移除被丢弃位置的信息，而合并则可能模糊词元与单个图块之间的对应关系。相比之下，RVLM 构建内容自适应池化单元，而非压缩现有词元序列，从而保留完整的图块覆盖范围和空间对应关系。我们在第 4.3 节中对这些方法在每张图像词元预算相同的条件下进行了比较。

3 方法

我们提出 RVLM，这是一种经过训练、能够将区域用作视觉单元的视觉语言模型（图 2）。传统的空间池化会对固定空间窗口内的图块进行汇总，而不考虑其视觉内容。与之不同，我们将外观相似且在空间上相连的图块分组，并将每组汇总为一个视觉词元。这些分组不必对应完整物体（正如 Chen 等（2025a）对物体子词元所提出的观点）；当一个物体跨越多个区域时，大语言模型可以整合这些区域对应词元中的信息，就像整合图像块令牌或子词词元中的信息一样。因此，我们的目标是

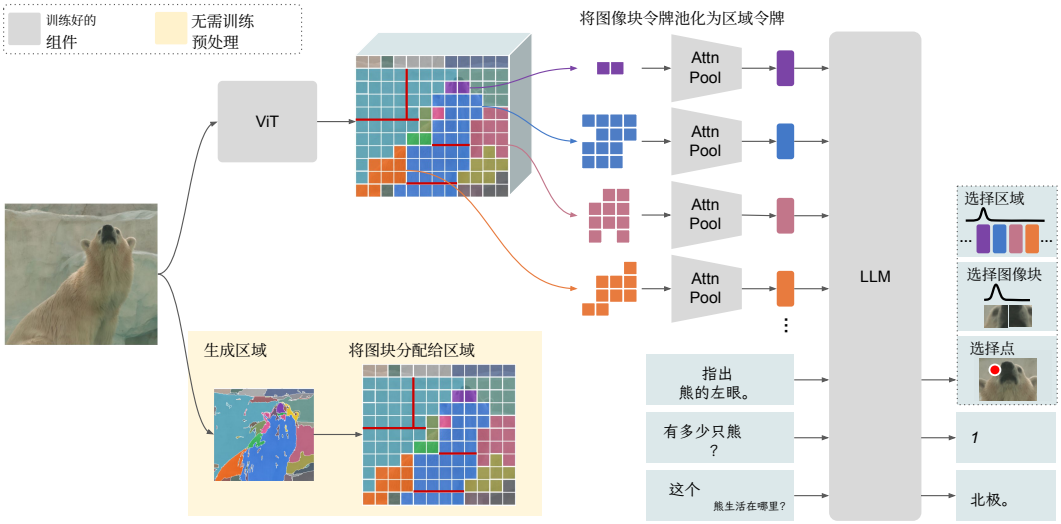


图 2：**RVLM 概览**。我们使用 Felzenszwalb-Huttenlocher 分割生成内容自适应区域，并将每个 ViT 图像块分配到一个区域。这是一种 CPU 端预处理，可与模型运行并行执行。每个区域内的图像块特征通过注意力池化进行聚合，生成供大语言模型使用的区域令牌序列。保留图像块到区域的对应关系后，即可实现由粗到细的空间定位：大语言模型先选择一个区域，再选择其中的一个图像块，最后选择图像块内的精确点位。

构建内容自适应池化单元，使其遵循局部图像结构，同时覆盖图像的每个像素。

3.1 建模

RVLM 遵循标准视觉语言模型架构，通过可训练连接器将预训练视觉编码器与大语言模型结合起来。我们使用 SigLIP2-SO400M/14（Tschannen 等，2025）初始化视觉编码器，并使用 Qwen3-4B-Instruct（Yang 等，2025a）初始化大语言模型。连接器从头开始初始化，并通过多头交叉注意力将视觉Transformer图块特征聚合为区域令牌。遵循 Molmo（Deitke 等，2025）的做法，视觉编码器将每张图像处理为一张低分辨率概览图和至多 K 张相互重叠的高分辨率裁剪图。如果图像所需的裁剪图超过 K 张，我们会将其缩小，直至符合这一数量限制。低分辨率概览图经过标准的 2×2 图像块池化处理，生成 196 个视觉令牌。对于高分辨率路径，我们从各个裁剪图中提取 ViT 特征，并将其拼接成一个统一的图像块网格，然后进行基于区域的注意力池化。所有基于视觉定位的预测（包括指点和分割）均基于这一高分辨率表示，而概览图仅用于提供全局上下文。训练时我们使用 $K{=}14$，并在推理时将其增加到 $K{=}32$，以研究如何扩展至更高的图像分辨率。

生成区域。 我们使用 Felzenszwalb-Huttenlocher（FH）分割（Felzenszwalb & Huttenlocher，2004年），将图像划分为完全覆盖的空间连通区域。FH 根据相对于内部变化的颜色相似度合并相邻组件。我们使用尺度参数 $k{=}100$，用于控制允许合并的激进程度；在边缘加权前使用 $\sigma{=}0.8$ 高斯平滑，以减少噪声；并将最小组件大小设为 32 个像素，小于该大小的组件会被吸收到相邻组件中。训练期间，图像较长的一边上限为 512 个像素，推理期间为 768 个像素；生成的标签图使用最近邻插值进行缩放。我们在数据加载器工作进程中使用 CPU 运行 FH，从而避免其与模型竞争 GPU 时间。我们将在第 4.4 节讨论其他区域生成策略。

将图块分配到区域。 我们将每个 ViT 图像块恰好分配给一个 FH 区域。为保留精细结构，我们将每个图像块分配给覆盖其至少 25% 像素的最小区域；如果没有区域达到此阈值，则将其分配给重叠面积最大的区域。为实现高效的批量注意力池化，我们将每个分组限制为 $w{=}16$ 个图块，并将较大的区域拆分为空间连续分组。这样既能保留用于空间定位的所有图块特征，又能控制填充量和每个标记的空间

范围。我们在附录 B.2 中研究了 w 的选择。这两条规则对于定位都是必需的：拆分操作确保在由粗到细指点的第二阶段中每个图块都可被选取；该阶段会针对每个区域标记考虑固定数量的图块，而穷举式的互斥分配则确保每个目标都可达且不存在歧义。缺少这两条规则时，由粗到细指点会失效（附录 B.4）。

将图块池化为区域令牌。我们使用交叉注意力连接器将每个图像块组转换为一个视觉词元。对于每个图像块组，我们提取视觉编码器倒数第三层和倒数第九层的特征，并将其拼接起来。为捕捉每个区域的空间结构，我们添加绝对正弦位置嵌入，以编码每个图像块的空间坐标（t，h，w）。遵循 Clark 等人（2026b）的方法，我们将有效图块特征的均值用作多头交叉注意力层的查询，并将各个图块特征用作键和值。小于 w=16 的图像块组会进行填充，注意力计算时会屏蔽填充位置。随后，通过多层感知机将所得表示投影到大语言模型嵌入空间中，从而为每个图像块组生成一个视觉词元。与固定窗口池化不同，该连接器能够自适应地聚合来自大小不一、空间上连续的区域的特征。值得注意的是，在我们的设置中，池化并不会丢弃图块级特征：定位头可以读取未池化的 ViT 图像块特征，我们会连同图像块到区域的分配一起保留这些特征（附录 A.3）。

位置编码。 我们使用交错式多模态旋转位置编码（M-RoPE）（Huang 等人，2026年）为大语言模型中的每个区域令牌设置位置。具体而言，我们为每个令牌分配一个三维位置（t，h，w），其中 t 表示图像在输入序列中的位置，而（h，w）是距离该区域空间中心最近的图像块坐标。这样，区域令牌便与原始图块网格处于同一空间坐标系中，同时不要求区域呈矩形。因此，多模态旋转位置编码（M-RoPE）为大语言模型编码区域在图像中的位置，而区域的空间结构则通过在将图像块组注意力池化为区域令牌之前添加的绝对图块级位置嵌入进行编码。

由粗到细的空间定位。 对于指点任务，我们遵循 MolmoPoint（Clark 等人，2026a）的由粗到细解码策略，并将其调整为适用于区域令牌。大语言模型不直接预测图像坐标，而是输出一个特殊令牌，并将其隐藏状态用作查询，与所有区域令牌对应的键进行评分（点积检索）；随后选取得分最高的区域。第二个特殊令牌以同样方式从该区域拥有的图像块中检索一个 ViT 图像块，第三个特殊令牌则预测该点在所选图像块内的 4×4网格位置。我们利用图像块到区域的分配关系，将预测的区域、图像块及局部位置映射回图像坐标。这种解码策略使基于区域的视觉语言模型能够实现精确定位；否则，由于一个区域可能覆盖图像很大一部分，这项任务会颇具挑战（附录 B.4）。对于分割任务，我们复用指点的第一阶段，为目标对象选取一个锚定区域。随后，大语言模型生成一个特殊的分割词元，并利用其隐藏状态预测三个层级的掩码分数：区域、各区域内的图像块，以及各图像块内的 4×4 单元。对于每个单元，我们将其区域级、图像块级和单元级分数相加，得到一个掩码逻辑值。利用保留的图像块到区域的分配关系，我们将这些逻辑值映射回对应的图像位置，应用 Sigmoid 函数，并以 0.5 为阈值进行二值化，得到二值掩码。这样，无需规则的池化网格，不规则的区域令牌也能支持密集且空间精确的预测。

3.2 训练

我们分两个阶段联合训练视觉编码器、连接器和大语言模型，遵循 Molmo2 的训练方案（Clark 等，2026b）。阶段1包括 31,971 步多模态预训练，涵盖图像描述、指点、计数、光学字符识别和文本指令数据。阶段2包括 24,000 步监督微调，新增图像问答、分割和深度预测。我们将下一词元交叉熵与 MolmoPoint （ Clark 等，2026a ）的空间定位目标结合进行优化。指点任务在由粗到细解码的每个层级使用交叉熵，而分割任务使用 focal 损失和 Dice 损失。附录 A 提供了完整的训练配置。

4 实验

我们评估了在通用视觉语言模型中，区域令牌能否替代图像块作为视觉表示。为隔离视觉词元化的影响，我们训练了 PVLM——一种基于图像块的视觉语言模型，其架构、训练数据和训练方案均与 RVLM 相同。这两个模型

表 1：**与开放权重视觉语言模型的比较**。我们以相同方式训练的图像块和区域模型，在图像问答、指点和计数基准上均取得了有竞争力的表现。因此，PVLM 为隔离基于区域的视觉词元化的影响提供了强有力的基线。对于 TextVQA，由于测试服务器不可用，我们报告了模型在验证集上的结果。

模型	VQAv2	AI2D	PixMo-Point	PixMo-Count	CountBench	TextVQA	DocVQA
InternVL3.5-4B	78.1	82.6	–	62.4	82.2	77.9	92.4
PLM-3B	84.4	90.9	–	63.0	87.1	84.3	93.8
Qwen3-VL-4B	81.7	84.1	43.7	58.0	85.5	81.0	**95.3**
Molmo2-4B	86.6	**95.6**	83.4	88.1	**93.9**	85.0	92.4
PVLM-4B	**87.1**	95.4	85.2	90.1	90.2	**85.5**	94.4
RVLM-4B	87.0	95.1	**86.4**	**90.4**	91.8	85.0	93.7

仅在视觉词元化策略上有所不同。首先，我们将这两个模型与规模相近的开放权重视觉语言模型进行比较（第 4.1 节）。随后，我们在涵盖图像问答、指点、计数、分割和深度估计的 35 项任务上评估区域词元化的效果（第 4.2 节）。接着，我们将 RVLM 与事后词元压缩方法进行比较（第 4.3 节）。最后，我们分析区域生成策略，并确定高效基于区域的视觉语言模型所需具备的特性（第 4.4 节）。附录 B 补充了设计消融实验，以及在未见领域、位置编码、模型规模、纯文本任务和视频任务上的结果。

4.1 与开放权重视觉语言模型的比较

表 1 将 RVLM 和 PVLM 与规模相近的开放权重视觉语言模型在标准图像基准测试上的表现进行了比较。两种模型在图像问答、指点和计数任务上均取得了具有竞争力的性能。重要的是，RVLM 使用的是内容自适应区域令牌，而非传统的图像块令牌，仍然取得了这一性能。这些结果也表明，PVLM 是一个强大的基线。由于 PVLM 和 RVLM 采用相同的架构、训练数据和训练方案，我们后续的比较将视觉标记化的影响单独分离出来。

4.2 区域与图像块令牌化

我们在涵盖视觉问答、指点、计数、分割和深度的35项任务上比较了PVLM和RVLM（表 2）。RVLM的家族均衡平均值提高了0.7个百分点，按样例加权的平均值提高了0.6个百分点，同时视觉令牌减少了21.5%。这表明，区域能够缩短视觉序列，同时不牺牲性能。

区域能提升空间任务的表现。 表2显示，提升最大的是计数（+0.9）、分割（+1.3）和深度（+1.9），其中SA-Co Gold、DDAD、Waymo和nuScenes上的提升尤为显著。这些提升表明，在对连贯的物体和表面进行推理时，区域是更有效的表征单元。性能下降最大的是InfoVQA（−4.9）、ScreenSpot Pro（−3.3）和ChartQA（−0.8），因为相关信息往往只占据很小的区域（例如，细字体的小字）。失败案例分析表明，这些极精细的细节有时会被吸收到周围较大的区域中，从而降低空间精度。

区域随图像尺寸增大而更具扩展性。我们按原生分辨率将样例分为五个区间，以测试区域词元化和图像块标记化如何随图像尺寸变化而扩展。图1a 总结了结果，完整细分结果见表10。随着分辨率提高，RVLM 的性能提升逐步增大，同时使用的视觉令牌比图像块基线越来越少。因此，区域词元化能更充分地利用额外的图像细节，同时避免图像块标记化带来的序列长度快速增长。这使其特别适用于文档、屏幕、机器人学和密集视觉预测等高分辨率场景。

在令牌预算固定的情况下，区域能带来更好的性能。我们从评测套件的所有任务中选取高分辨率图像，共得到 21,273 个示例，图像分辨率介于 2.08 至 2.63 MP 之间，并在七种“最高分辨率”设置下重新评估 RVLM 和 PVLM。在每种设置中，所选高分辨率示例的图像分辨率都被限制在指定的最高值，从而能够衡量随着可用图像细节逐步增加，模型性能如何变化。由此得到

表2：**图像块与区域令牌化**。 RVLM 在使用更少视觉令牌的同时，整体性能不低于甚至优于训练方式完全相同的图像块基线。不过，其主要优势在于能够进行更高分辨率的处理，如图1b所示。按示例加权的平均值对每个示例赋予相同权重，而按任务系列加权的平均值则对每个任务类别赋予相同权重。令牌数量取所有示例的平均值。

模型	按示例加权的平均值	按任务系列均衡加权的平均值	令牌数量
PVLM	82.8	79.1	985
RVLM	83.4 (↑0.6)	79.8 (↑0.7)	773 (↓21.5%)

(a) **推理/理解**。在通用图像问答任务中，区域的表现与图块大致持平。在文档式视觉问答任务中，区域在 ChartQA、DocVQA 和 TextVQA 上具有竞争力，只有在 InfoVQA 上表现明显下降。

Task	PVLM	RVLM
图像问答		
A-OKVQA 直接回答	75.5	74.8
A-OKVQA 多项选择	88.6	88.2
AI2D	95.4	95.1
OK-VQA	65.8	66.0
ScienceQA	98.2	97.8
VQAv2	87.2	87.1
ChartQA	83.7	82.9
DocVQA	93.2	92.8
InfoVQA	79.2	74.3
TextVQA	85.5	85.0
任务平均值	85.2	84.4
令牌数量	793	700

(b)**定位**。 在每个系列中，区域的表现都达到或超过图像块，同时使用的视觉令牌减少了32-41%。

Task	PVLM	RVLM
指点		
OSWorld-G	51.6	52.0
PixMo-Point	85.2	86.4
PointBench	63.8	63.3
SA-Co Gold	87.5	90.1
ScreenSpot Pro	29.3	26.0
ScreenSpot v2	82.7	85.0
任务平均值	66.7	67.1
令牌数量	1,793	1,065
计数		
CountBenchQA	90.2	91.8
PixMo-Count	90.1	90.4
任务平均值	90.2	91.1
令牌数量	1,655	990
分割		
PixMo-Seg	58.0	59.8
RefCOCO	82.0	82.5
RefCOCO+	78.5	78.7
RefCOCOg	80.3	81.1
SA-Co Gold	50.7	54.2
任务平均值	69.9	71.2
令牌数量	1,390	950

（c）**三维**。 我们报告了 12 个数据集上点到深度的尺度不变$\delta_{1.25}$ 结果。区域的平均表现与图像块相当或更优，同时使用的视觉令牌减少了 43%。

Task	PVLM	RVLM
深度		
Argoverse 2	77.9	81.8
Aria AEA	68.5	69.5
Aria Nymeria	54.6	55.6
DDAD	81.7	87.7
DIODE	92.2	94.0
EgoExo4D 自我	89.1	88.9
EgoExo4D Exo	84.0	82.9
nuScenes	84.7	88.8
NYU Depth v2	97.8	97.5
Stereo4D	89.0	88.6
SUN RGB-D	98.2	98.2
Waymo	84.1	90.8
任务平均值	83.5	85.4
令牌数量	1,617	916

图1b展示了令牌预算相匹配的比较。由于区域能够在相同视觉令牌预算下表示更高分辨率的输入，因此当大语言模型接收数量相当的视觉令牌时，RVLM 的表现始终优于 PVLM。这使 RVLM 能够将节省的令牌用于提高输入分辨率，并在固定的大语言模型上下文预算下提升性能。

区域能够跨领域、尺度和模态泛化。在未见过的病理学和卫星图像上，RVLM 使用的视觉令牌约为 PVLM 的一半，表现却与之相当（附录B.5）。对于两种模型都未经过训练的视频，RVLM 使用更少令牌（App. B.8），表现与 PVLM 的差距仍在 0.5 点以内。在 17 亿规模下也呈现相同趋势（App. B.7），而区域词元化基本不会改变纯文本任务的表现（App. B.8）。

成本分析。 我们在表 3 中报告端到端吞吐量和内存测量结果。区域词元化带来的吞吐量提升源于大语言模型处理的视觉序列更短。FH分割为每张图像增加了约 49 毫秒的 CPU 预处理时间。使用八个数据加载器工作进程时，在两种批次大小下，这一预处理都能跟上 GPU 执行速度，因此表 3 中的吞吐量包含了区域构建过程，且未占用加速器时间。此外，在超高分辨率输入下，FH 的成本仍保持受限，因为 FH 在缩小后的副本上运行，且其长边上限为768像素。针对不同图像分辨率的更细粒度分析（表11）显示，对于批次大小 1，在一张和四张裁剪图像的情况下，PVLM 略快；此时，减少令牌数量带来的收益尚不足以抵消区域池化开销。从八张裁剪图像开始，RVLM 在两种批次大小下都实现了更低的延迟，并且在更高分辨率下，延迟和内存节省幅度不断增加。这些结果是在使用40亿参数模型的情况下测得的；更大的大语言模型可能会因视觉序列缩短而获得更高的效率提升。

表3：**推理成本**。吞吐量是在 PixMo-Cap 中随机抽取的 512 个样本上，按实际运行时间计算的每秒处理样本数，其中包括数据加载和区域构建。生成上限为一个输出词元，因此吞吐量反映的是预填充速度。测量使用单张 NVIDIA H100，并在 64 个预热批次后进行。我们报告五次运行的平均值和标准差。较短的视觉序列能够提高吞吐量并降低峰值内存，且批量大小越大，收益越明显。表11展示了不同图像分辨率下更细粒度的成本比较。

模型	批次大小 1		批次大小 8	
	吞吐量（每秒样本数）↑	峰值内存（GiB）↓	吞吐量（每秒样本数）↑	峰值内存（GiB）↓
PVLM	13.39±0.16	19.98	19.27±0.07	26.75
RVLM	**15.90±0.38**	**19.75**	**26.18±0.07**	**24.69**
Δ	**+18.7%**	**−1.2%**	**+35.9%**	**−7.7%**

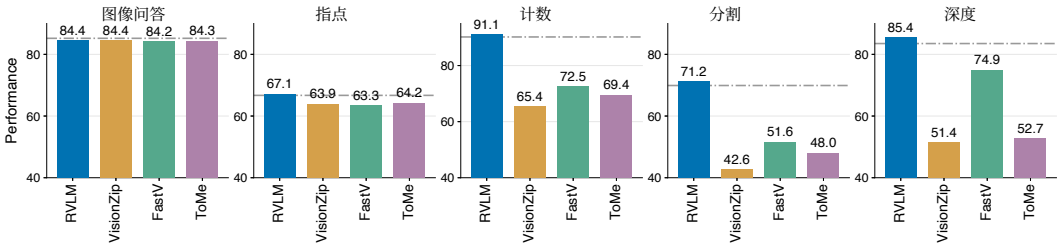


图3：**事后词元压缩方法在需要空间精度的任务上表现急剧下降**。在指点、分割、计数和深度估计方面，RVLM 优于令牌剪枝（FastV）、令牌合并（ToMe）和混合方法（VisionZip）；而在图像问答方面，所有方法的表现相当。水平线表示 PVLM 基线。

4.3 RVLM 与词元压缩的比较

缩短视觉序列的另一种方法是通过剪枝（FastV（Chen 等，2024a））、合并（ToMe（Bolya 等，2023））或二者结合（VisionZip（Yang 等，2025b））来压缩图像块令牌。我们将这些方法应用于 PVLM，并在图3中与 RVLM 进行比较。为进行受控比较，我们通过将每种压缩方法的目标令牌数量设置为 RVLM 针对该图像生成的区域令牌数量，使每张图像的视觉词元预算保持一致。这样便为基线方法提供了一个按图像变化的信号，而这些方法自身的设定并未假定该信号，从而使它们能够对稠密场景采取比稀疏场景更弱的压缩。

如图3所示，事后压缩在很大程度上保留了图像理解任务的性能，这与此前关于视觉序列包含冗余的研究发现一致。然而，我们观察到，指点、分割和点到深度预测任务的性能大幅下降，这与 Chien 等人（2025年）的研究发现一致。剪枝会移除单独的空间表征，而合并则会将多个位置的信息整合到共享词元中。相比之下，RVLM 保留了从每个图像块到一个区域标记的显式映射，使其由粗到细解码器能够恢复空间精确的预测。因此，RVLM 的优势不仅在于使用更少的视觉词元，还在于缩短序列的同时保留空间覆盖和对应关系。

4.4 构建区域

RVLM 需要将图像块映射到区域。我们评估了多种区域生成策略，并分析了它们的失败案例。分析表明，对于通用视觉语言模型，区域必须覆盖整幅图像，在端到端训练期间保持空间连通，并保留小字等精细结构，而语义分组的作用不大。我们将在下文讨论各项策略，并在表4和图4中总结研究结果。

区域编码器网络。我们首先尝试用 T-REN（Khosla 等人，2026）取代基于图像块的视觉变换器。T-REN 是一种高效的图像编码器，可直接生成区域令牌。尽管这种方法为基于区域的视觉语言模型提供了一个很有前景的自然起点，但也揭示了两个根本性挑战。

表4：**比较区域生成策略**。其属性源自第 4.4 节讨论的失效模式。使用 Qwen3-1.7B 作为语言模型完成第一阶段训练后，在 PixMo-Cap（上限）和 PixMo-Point（点）上的性能报告如下。

策略	完整图像块覆盖	连通性在……后保留训练	无外部区域编码器	编码器-独立区域	区域在图像处加载	Cap ↑	点 ↑
区域编码器网络							
采用 SigLIP-2 的 T-REN	✓	✗	✓	✗	✗	52.9	77.6
T-REN 搭配 DINOv3	✓	✗	✓	✗	✗	51.9	80.8
T-REN 搭配 C-RADIOv4-H	✓	✗	✓	✗	✗	52.9	84.4
基于特征的聚类							
DiVT（冻结的视觉变换器）	✓	✓	✓	✗	✗	51.4	79.8
DiVT（可训练视觉变换器）	✓	✗	✓	✗	✗	52.6	80.2
DiVT（双视觉变换器）	✓	✓	✗	✗	✗	53.1	86.4
学习式分割器							
SAM	✗	✓	✗	✓	✗	–	–
EPOC	✓	✓	✗	✓	✗	53.1	86.6
免训练像素聚类							
SLIC	✓	✓	✓	✓	✓	53.0	86.5
FH（本文方法）	✓	✓	✓	✓	✓	53.8	

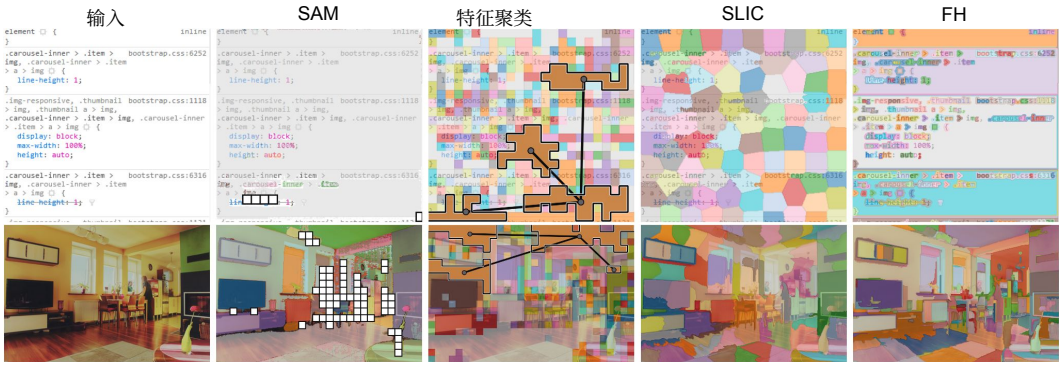


图4：**比较区域生成方法**。SAM 会留下未覆盖的图像区域（白色方框），而基于学习特征的方法（DiVT、T-REN）会在视觉语言模型训练期间生成不连通的区域。SLIC 往往无法分离文本，而 FH 能保留精细文本结构，生成连通区域并实现完整的图像覆盖。

首先，区域质量很大程度上取决于视觉编码器的选择。T-REN 依赖密集图块特征的空间结构来生成区域，因此某些视觉骨干网络比其他网络更适合这一任务。如表4所示，我们使用搭配 SigLIP-2、DINOv3 和 C-RADIOv4-H 的 T-REN 训练 RVLM，发现基于 SigLIP-2 的 T-REN 空间精度不足，而 DINOv3 能提供更好的空间特征，但向大语言模型的迁移效果较差。在评估的视觉编码器中，C-RADIOv4-H 的整体表现最佳。更关键的是，使用 T-REN 时，区域在视觉语言模型训练过程中会失去空间连通性。即使使用 C-RADIOv4-H，在仅训练 1,000 步后，PixMo-Cap 的文本密集型图像中就有 ∼37% 的区域令牌在空间上彼此断开。单个位置ID无法忠实表示多个不连通的组件，因此模型难以将视觉内容与其位置关联起来。对于结构化文本图像，这一问题尤为突出（附录中的图5）。强制保持连通性会产生近似矩形的区域，牺牲了区域令牌化所追求的内容自适应性。冻结 T-REN 虽能缓解区域分离问题，却会降低性能。

基于特征的聚类。遵循 DiVT（Lee 等，2026），我们尝试通过将视觉Transformer特征相似的图块分组来构建区域。我们首先采用 DiVT 的训练设置：冻结视觉编码器，同时训练连接器并微调大语言模型。然而，与端到端训练相比，冻结 ViT 会限制下游性能。因此，我们允许 ViT 与大语言模型和连接器联合训练，但发现随着图块嵌入不断适应，生成的区域会变得空间上彼此断开，重现了 T-REN 中观察到的问题。

为避免这种权衡，我们使用单独的冻结 ViT 进行图块分组，同时允许视觉语言模型的视觉编码器自由训练，从而将区域生成与视觉语言模型训练解耦。使用 C-RADIOv4-H 作为冻结编码器后，这一变体在使用更少视觉令牌的同时，性能接近完全训练的图像块基线。不过，它需要对每张图像额外执行一次 ViT 前向传播，从而增加参数量、延迟和内存开销。

学习式分割器。 分割模型（如 SAM（Kirillov 等人，2023年））可用于获得非常精确的分割结果。然而，使用 SAM 为一张图像生成所有区域既缓慢又耗费计算资源。更重要的是，SAM 无法保证完整的图像覆盖，导致某些图块没有对应的区域（图4和附录B.5）。这对指点任务尤其成问题，因为无法可靠地预测那些未纳入视觉表示的位置。EPOC（Chen 等，2025a）通过学习式边界检测器结合分水岭分割来解决覆盖不完整的问题。然而，其区域构建会带来预处理开销。例如，使用八个数据加载器工作进程时，FH 的分割吞吐量为 159 张图像/秒，而 EPOC 为 23 张图像/秒，相差 ∼7× 。FH 的预处理可以与视觉语言模型的执行并行进行，而 EPOC 依赖加速器的流水线限制了这种重叠。我们还发现，EPOC 在文本密集型图像上的表现不如 FH，而这类图像是表4中 PixMo-Cap 性能差距的主要来源。

免训练像素聚类。 SLIC（Achanta 等人，2012年）和 FH（Felzenszwalb 和 Huttenlocher，2004年）等方法避免了先前方法的局限。它们无需训练，直接对图像像素进行操作，因此区域质量不受视觉语言模型的视觉编码器影响。此外，它们还能提供空间连通区域和完整的图像覆盖，而无需增加模型参数或前向传播。尽管这些方法往往产生更细的划分，且令牌减少效果不如 T-REN 或 SAM，但它们仍能提供内容自适应的视觉词元，同时满足端到端视觉语言模型训练和空间定位的要求。仅根据输入计算区域分配也简化了训练。由于它们不依赖模型，因此在图像进入模型之前，区域令牌的数量和空间范围都是已知的，数据加载器工作进程可以分配位置ID，并与 GPU计算并行地打包可变长度序列。相比之下，区域由网络生成的方法必须在数据加载期间运行该网络，从而增加预处理成本（EPOC：FH 为每秒 159 张图像，而 EPOC 为每秒 23 张图像）；或者，当网络与视觉语言模型一起训练时（T-REN、采用可训练视觉变换器的 DiVT），将区域构建推迟到前向传播阶段。后者需要使用具有固定视觉令牌预算的占位序列：预算过大时会浪费计算资源，预算过小时则会截断区域序列。

总结。 有三项特性可以区分出能够跨任务发挥作用的区域策略与不能发挥作用的区域策略。(1) 区域必须覆盖完整图像，而 SAM 无法保证这一点。(2) 在端到端训练过程中，区域必须保持空间连通，而从视觉语言模型自身特征中得到的区域无法做到这一点。(3) 区域还必须保留小字等精细结构，而 SLIC 和大多数学习式方法在这方面表现不佳。此外，语义分组的作用不大，因为大语言模型可以跨物体部件聚合信息，就像它能够跨图块聚合信息一样。FH 满足全部三项特性，且无需增加参数或前向传播，因此我们在 RVLM 中采用了 FH。

5 结论

我们提出 RVLM，这是一种以内容自适应区域令牌取代固定空间池化的视觉语言模型。对区域构建的分析表明，对于通用视觉语言模型，区域必须覆盖完整图像，在端到端训练期间保持空间连通，并保留小字等精细结构，而语义分组所带来的增益甚微。因此，快速且无需训练的基于像素的分割器便已足够。区域令牌在减少视觉令牌用量的同时，能够保持或提升性能，且图像分辨率越高，优势越明显。RVLM 保留了完整图像块覆盖和明确的图像块到区域对应关系，无需增加模型参数即可支持精确的基于视觉定位的预测。这些结果在未见过的病理学和卫星图像、17亿规模以及未经视频训练的视频任务上均成立。总之，这些发现确立了基于区域的令牌化是一种适用于通用视觉语言模型的高效替代方案，可取代固定空间池化。未来工作可将这一方法扩展至长视频，例如对跨视频帧的区域进行分组，并使用稀疏混合专家骨干网络进行评估。

人工智能使用声明

在本研究中，我们使用生成式人工智能工具提升作者撰写文本的清晰度和可读性，协助进行稿件格式编排，创建仅用于开发阶段的探索性可视化，并重构和优化现有代码，同时不改变底层方法。作者手动审阅了所有经人工智能辅助生成的材料，并检查了人工智能生成的代码。我们通过确认代码变更后的输出与原始实现的输出一致，对这些变更进行了验证。我们还根据原始来源核实了事实性主张和引文，并对照源数据检查了表格条目。我们对本研究的最终内容负责，包括借助生成式人工智能制作的所有文本、主张、代码和材料。

可复现性声明

我们在第 3 节介绍模型架构、区域构建流程、图像块到区域的分配以及空间定位形式化。附录 A 记录了训练配置、数据混合、预处理参数、任务形式化、评估数据集和评估指标。第 4 节和附录 A 介绍了与基于图像块的模型、令牌压缩方法及其他区域构建策略进行比较的实验设置。

参考文献

Radhakrishna Achanta、Appu Shaji、Kevin Smith、Aurelien Lucchi、Pascal Fua 和 Sabine S¨usstrunk。与最先进的超像素方法相比的 SLIC 超像素。*PAMI*，2012年。

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, 和 Karen Simonyan。《Flamingo：一种用于少样本学习的视觉语言模型》。载于 *NeurIPS*，2022年。

Saeed Ranjbar Alvar、Gursimran Singh、Mohammad Akbari 和 Yong Zhang。《Divprune：面向大型多模态模型的多样性视觉令牌剪枝》。载于 *CVPR*，2025年。

Jinze Bai、Shuai Bai、Shusheng Yang、Shijie Wang、Sinan Tan、Peng Wang、Junyang Lin、Chang Zhou 和 Jingren Zhou。《Qwen-VL：一种用于理解、定位、文本阅读及其他任务的通用视觉-语言模型》。*arXiv*，2023年。

Daniel Bolya、Cheng-Yang Fu、Xiaoliang Dai、Peizhao Zhang、Christoph Feichtenhofer 和 Judy Hoffman。《令牌合并：更快的 ViT》。发表于*ICLR*，2023年。

Junbum Cha、Wooyoung Kang、Jonghwan Mun 和 Byungseok Roh。《Honeybee：用于多模态大语言模型的局部性增强投影器》。发表于*CVPR*，2024年。

Delong Chen、Samuel Cahyawijaya、Jianfeng Liu、Baoyuan Wang 和 Pascale Fung。《子对象级图像标记化》。发表于*ICML*，2025a。

Jun Chen、Deyao Zhu、Xiaoqian Shen、Xiang Li、Zechun Liu、Pengchuan Zhang、Raghuraman Krishnamoorthi、Vikas Chandra、Yunyang Xiong 和 Mohamed Elhoseiny。《MiniGPT-v2：作为视觉语言多任务学习统一接口的大语言模型》。*arXiv*，2023a。

Liang Chen、Haozhe Zhao、Tianyu Liu、Shuai Bai、Junyang Lin、Chang Zhou 和 Baobao Chang。第 2 层之后，一张图像只需 1/2 个令牌：用于大型视觉语言模型的即插即用推理加速。发表于 *ECCV*，2024a。

Xi Chen、Xiao Wang、Soravit Changpinyo、AJ Piergiovanni、Piotr Padlewski、Daniel Salz、Sebastian Goodman、Adam Grycner、Basil Mustafa、Lucas Beyer、Alexander Kolesnikov、Joan Puigcerver、Nan Ding、Keran Rong、Hassan Akbari、Gaurav Mishra、Linting Xue、Ashish V Thapliyal、James Bradbury、Weicheng Kuo、Mojtaba Seyedhosseini、Chao Jia、Burcu Karagol Ayan、Car- los Riquelme Ruiz、Andreas Peter Steiner、Anelia Angelova、Xiaohua Zhai、Neil Houlsby 和 Radu Soricut。《Pali：联合扩展的多语言图像语言模型》。发表于*ICLR*，2023b。

Zhe Chen、Jiannan Wu、Wenhai Wang、Weijie Su、Guo Chen、Sen Xing、Muyan Zhong、Qinglong Zhang、Xizhou Zhu、Lewei Lu、Bin Li、Ping Luo、Tong Lu、Yu Qiao 和 Jifeng Dai。《Internvl：扩展视觉基础模型并使其适配通用视觉语言任务》。发表于*CVPR*，2024b。

Zhe Chen、Weiyun Wang、Yue Cao、Yangzhou Liu、Zhangwei Gao、Erfei Cui、Jinguo Zhu、Shen- glong Ye、Hao Tian、Zhaoyang Liu、Lixin Gu、Xuehui Wang、Qingyun Li、Yiming Ren、Zixuan Chen、Jiapeng Luo、Jiahao Wang、Tan Jiang、Bo Wang、Conghui He、Botian Shi、Xingcheng Zhang、Han Lv、Yi Wang、Wenqi Shao、Pei Chu、Zhongying Tu、Tong He、Zhiyong Wu、Huipeng Deng、Jiaye Ge、Kai Chen、Kaipeng Zhang、Limin Wang、Min Dou、Lewei Lu、Xizhou Zhu、Tong Lu、Dahua Lin、Yu Qiao、Jifeng Dai 和 Wenhai Wang。通过模型、数据和测试时扩展，拓展开源多模态模型的性能边界。*arXiv*，2025b。

Tzu-Chun Chien、Chieh-Kai Lin、Shiang-Feng Tsai、Ruei-Chi Lai、Hung-Jen Chen 和 Min Sun。感知定位的令牌剪枝：从剪枝导致的视觉定位性能骤降中恢复。*arXiv*，2025年。

Xiangxiang Chu、Limeng Qiao、Xinyu Zhang、Shuang Xu、Fei Wei、Yang Yang、Xiaofei Sun、Yiming Hu、Xinyang Lin、Bo Zhang 和 Chunhua Shen。Mobilevlm v2：更快、更强的视觉语言模型基线。*arXiv*，2024年。

Christopher Clark、Yue Yang、Jae Sung Park、Zixian Ma、Jieyu Zhang、Rohun Tripathi、Moham- madreza Salehi、Sangho Lee、Taira Anderson、Winson Han 和 Ranjay Krishna。MolmoPoint：使用定位令牌提升视觉语言模型的指点能力。发表于 *ECCV*，2026a。

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, Vincent Shao, Yue Yang, Weikai Huang, Ziqi Gao, Taira Anderson, Jianrui Zhang, Jitesh Jain, George Stoica, Winson Han, Ali Farhadi 和 Ranjay Krishna。《Molmo2：面向具备视频理解和定位能力的视觉语言模型的开放权重和数据》。载于*CVPR*，2026 b。

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Moham- madreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison- Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Bor- chardt, Dirk Groeneveld, Jen Dumas, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi 和 Aniruddha Kembhavi。《Molmo 与 pixmo：面向最先进视觉语言模型的开放权重和开放数据》。载于*CVPR*，2025年。

Shivam Duggal、Sanghyun Byun、William T. Freeman、Antonio Torralba 和 Phillip Isola。单一 - p用于最小程序搜索的自适应图像标记化。在 *NeurIPS*，2025年。

Pedro F. Felzenszwalb 和 Daniel P. Huttenlocher。高效的基于图像的图像分割。*IJCV*，2004年。

Jie Huang、Xuejing Liu、Sibo Song、RuiBing Hou、Hong Chang、Junyang Lin 和 Shuai Bai。重新审视视觉语言模型中的多模态位置编码。在 *ICLR*，2026年。

Peng Jin、Ryuichi Takanobu、Wancai Zhang、Xiaochun Cao 和 Li Yuan。Chat-univi：统一视觉表示赋能大语言模型实现图像和视频理解。在 *CVPR*，2024年。

Savya Khosla、Sethuraman TV、Barnett Lee、Alexander Schwing 和 Derek Hoiem。REN：基于图块的图像编码器生成的快速高效区域编码。在*NeurIPS*，2025年。

Savya Khosla、Sethuraman TV、Aryan Chadha、Alexander Schwing 和 Derek Hoiem。T-REN：学习与文本对齐的区域令牌，可改善密集视觉语言对齐和可扩展性。发表于*ECCV*，2026年。

Alexander Kirillov、Eric Mintun、Nikhila Ravi、Hanzi Mao、Chlo´e Rolland、Laura Gustafson、Tete Xiao、Spencer Whitehead、Alexander C. Berg、Wan-Yen Lo、Piotr Doll´ar 和 Ross B. Girshick。分割一切。发表于国际计算机视觉大会，2023年。Hugo Laurenc,on、Andr´es Marafioti、Victor Sanh 和 L´eo Tronchon。构建更好的视觉语言模型：理解与未来方向。*arXiv*，2024年。Hyun Lee、Hyemin Jeong、Yejin Kim、Hyungwook Choi、Hyunsoo Cho、Soo Kyung Kim 和 Joon-seok Lee。面向多模态大语言模型、更接近词语的图像标记化。发表于*CVPR*，2026年。Jaihyun Lew、Soohyuk Jang、Jaehoon Lee、Seungryong Yoo、Eunji Kim、Saehyung Lee、Jisoo Mok、Siwon Kim 和 Sungroh Yoon。面向视觉变换器的超像素标记化：保持视觉令牌的语义完整性。*arXiv*，2025年。Junnan Li、Dongxu Li、Silvio Savarese 和 Steven Hoi。BLIP-2：利用冻结图像编码器和大语言模型进行语言-图像预训练。发表于*ICML*，2023年。Wentong Li、Yuqian Yuan、Jian Liu、Dongqi Tang、Song Wang、Jie Qin、Jianke Zhu 和 Lei Zhang。TokenPacker：面向多模态大语言模型的高效视觉投影器。*IJCV*，2025年。Haotian Liu、Chunyuan Li、Qingyang Wu 和 Yong Jae Lee。视觉指令微调。发表于*NeurIPS*，2023年。Chuofan Ma、Yi Jiang、Jiannan Wu、Zehuan Yuan 和 Xiaojuan Qi。Groma：用于定位多模态大语言模型的局部视觉标记化。发表于*ECCV*，2024年。Junyuan Mao、Qiankun Li、Linghao Meng、Zhicheng He、Xinliang Zhou、Kun Wang、Yang Liu 和 Yueming Jin。Granulon：利用自适应多粒度语义唤醒面向多模态大语言模型的像素级视觉编码器。发表于*CVPR*，2026年。

Andr´es Marafioti、Orr Zohar、Miquel Farr´e、Merve Noyan、Elie Bakouch、Pedro Cuenca、Cyril Za-kka、Loubna Ben Allal、Anton Lozhkov、Nouamane Tazi、Vaibhav Srivastav、Joshua Lochner、HugoLarcher、MathieuMorlon、LewisTunstall、Leandrovon Werra 和 ThomasWolf。SmolVLM：重新定义小型高效的多模态模型。*arXiv*，2025年。

Yuzhang Shang、Mu Cai、Bingxin Xu、Yong Jae Lee 和 Yan Yan。Llava-prumerge：面向高效大型多模态模型的自适应词元缩减。发表于国际计算机视觉大会，2025年。Michal Shlapentokh-Rothman、Ansel Blume、Yao Xiao、Yuqun Wu、Sethuraman TV、Heyi Tao、Jae Yong Lee、Wilfredo Torres、Yu-Xiong Wang 和 Derek Hoiem。重新审视基于区域的表征。发表于*CVPR*，2024年。Michael Tschannen、Alexey Gritsenko、Xiao Wang、Muhammad Ferjad Naeem、Ibrahim Alabdul- mohsin、Nikhil Parthasarathy、Talfan Evans、Lucas Beyer、Ye Xia、Basil Mustafa、Olivier H´enaff、Jeremiah Harmsen、Andreas Steiner 和 Xiaohua Zhai。Siglip 2：具备更强语义理解、定位能力和密集特征的多语言视觉-语言编码器。*arXiv*，2025年。Mengyu Wang、Yuyao Huang、Henghui Ding、Xinlong Wang、Tiejun Huang、Yao Zhao、Yunchao Wei 和 Shuicheng Yan。区域原生视觉词元化。发表于*ECCV*，2024a。Peng Wang、Shuai Bai、Sinan Tan、Shijie Wang、Zhihao Fan、Jinze Bai、Keqin Chen、Xuejing Liu、Jialin Wang、Wenbin Ge、Yang Fan、Kai Dang、Mengfei Du、Xuancheng Ren、Rui Men、Dayiheng Liu、Chang Zhou、Jingren Zhou 和 Junyang Lin。Qwen2-vl：增强视觉-语言模型在任意分辨率下的世界感知能力。*arXiv*，2024b。Zichen Wen、Yifeng Gao、Weijia Li、Conghui He 和 Linfeng Zhang。多模态大语言模型中的令牌剪枝：我们是否在解决正确的问题？发表于*ACL*论文集，2025a。Zichen Wen、Yifeng Gao、Shaobo Wang、Junyuan Zhang、Qintong Zhang、Weijia Li、Conghui He 和 Linfeng Zhang。别再在多模态语言模型中寻找重要词元：重复性更为关键。发表于自然语言处理实证方法会议，2025b。Shengqiong Wu、Hao Fei、Xiangtai Li、Jiayi Ji、Hanwang Zhang、Tat-Seng Chua 和 Shuicheng Yan。迈向多模态大语言模型中令牌化的语义等价性。发表于*ICLR*，2025年。

Long Xing、Qidong Huang、Xiaoyi Dong、Jiajie Lu、Pan Zhang、Yuhang Zang、Yuhang Cao、Conghui He、Jiaqi Wang、Feng Wu 和 Dahua Lin。面向高效大型视觉-语言模型的锥形视觉集中。发表于*CVPR*，2025年。

Wilson Yan、Volodymyr Mnih、Aleksandra Faust、Matei Zaharia、Pieter Abbeel 和 Hao Liu。《Elas-tictok：用于图像和视频的自适应分词》。发表于 *ICLR*，2025年。

An Yang、Anfeng Li、Baosong Yang、Beichen Zhang、Binyuan Hui、Bo Zheng、Bowen Yu、Chang Gao、Chengen Huang、Chenxu Lv、Chujie Zheng、Dayiheng Liu、Fan Zhou、Fei Huang、Feng Hu、Hao Ge、Haoran Wei、Huan Lin、Jialong Tang、Jian Yang、Jianhong Tu、Jianwei Zhang、Jianxin Yang、Jiaxi Yang、Jing Zhou、Jingren Zhou、Junyang Lin、Kai Dang、Keqin Bao、Kexin Yang、Le Yu、Lianghao Deng、Mei Li、Mingfeng Xue、Mingze Li、Pei Zhang、Peng Wang、Qin Zhu、Rui Men、Ruize Gao、Shixuan Liu、Shuang Luo、Tianhao Li、Tianyi Tang、Wenbiao Yin、Xingzhang Ren、Xinyu Wang、Xinyu Zhang、Xuancheng Ren、Yang Fan、Yang Su、Yichang Zhang、Yinger Zhang、Yu Wan、Yuqiong Liu、Zekun Wang、Zeyu Cui、Zhenru Zhang、Zhipeng Zhou 和 Zihan Qiu。《Qwen3 技术报告》。*arXiv*，2025a。

Senqiao Yang、Yukang Chen、Zhuotao Tian、Chengyao Wang、Jingyao Li、Bei Yu 和 Jiaya Jia. Visionzip：视觉语言模型中，更长更好，但并非必需。在 *CVPR*，2025b。

Xiaobo Yang 和 Xiaojin Gong。将 SAM 重新用于高效视觉投影器，以实现基于多模态大语言模型的指代图像分割。*TOMM*，2025年。

Linli Yao、Lei Li、Shuhuai Ren、Lean Wang、Yuanxin Liu、Xu Sun 和 Lu Hou。Deco：在多模态大语言模型中，将词元压缩与语义抽象解耦。*arXiv*，2024年。

Jiabo Ye、Anwen Hu、Haiyang Xu、Qinghao Ye、Ming Yan、Yuhao Dan、Chenlin Zhao、Guohai Xu、Chenliang Li、Junfeng Tian、Qian Qi、Ji Zhang 和 Fei Huang。mplug-docowl：用于文档理解的模块化多模态大语言模型。*arXiv*，2023年。

Qizhe Zhang、Aosong Cheng、Ming Lu、Renrui Zhang、Zhiyong Zhuo、Jiajun Cao、Shaobo Guo、Qi She 和 Shanghang Zhang。超越文本-视觉注意力：利用视觉线索实现视觉语言模型中的有效令牌剪枝。载于国际计算机视觉大会，2025a。

Qizhe Zhang、Mengzhen Liu、Lichen Li、Ming Lu、Yuan Zhang、Junwen Pan、Qi She 和 Shanghang Zhang。超越注意力或相似度：最大化多模态大语言模型中令牌剪枝的条件多样性。载于*NeurIPS*，2025b。

Yuan Zhang、Chun-Kai Fan、Junpeng Ma、Wenzhao Zheng、Tao Huang、Kuan Cheng、Denis Gu- dovskiy、Tomoyuki Okuno、Yohei Nakata、Kurt Keutzer 和 Shanghang Zhang。Sparsevlm：用于高效视觉语言模型推理的视觉令牌稀疏化。发表于 *ICML*，2025c。

附录

我们在 A节中提供训练和评估的详细信息。B节报告了关于 FH 参数、连接器宽度和分辨率缩放的补充实验，以及针对使用区域令牌进行指点、向未见领域的泛化、替代位置编码、模型规模、纯文本评估和视频评估的设计消融实验。C节讨论了局限性。

A 实现细节

A.1 架构

我们采用 MolmoPoint（Clark 等人，2026a）作为图像块基线（PVLM）的基础，以及 RVLM 的初始架构，这一选择源于它在规模相近的开放权重视觉语言模型中表现出色（表 1）。我们使用 Qwen3-4B-Instruct（Yang 等，2025a）初始化大语言模型，并使用 SigLIP2-SO400M/14（Tschannen 等，2025）初始化视觉变换器。大语言模型的隐藏维度为 2560，包含 36 层、32 个注意力头和 8 个 KV头。视觉编码器处理 378 像素的输入，图像块大小为 14。连接器拼接来自第 -3 层和第 -9 层的视觉Transformer特征，然后采用 16 头交叉注意力，将每个图像块组的均值作为查询，并将其组成特征作为键和值。池化表示经由多层感知机投影到大语言模型嵌入空间。我们使用裁剪边距为 $[4, 4]$ 的自适应重叠裁剪图像。

A.2 训练方案

我们分两个阶段训练 RVLM，并在整个过程中联合优化视觉编码器、连接器和大语言模型。阶段1进行多模态预训练，共31,971步。阶段2从阶段1检查点初始化，并再进行24,000步监督微调。表5总结了训练配置，表6展示了数据混合。该训练方案采用自同期准备中的工作，并非本文的贡献。

A.3 任务形式化

我们概述模型执行各项任务的方式。指点任务形式遵循 Molmo-Point（Clark 等人，2026a），而分割和点到深度任务形式则采用自同期准备中的工作，并非本文的贡献。

图像问答和图像描述。我们将图像问答和图像描述形式化为自回归文本生成。给定一张图像和一段文本提示，模型以图像的区域令牌表示为条件，并使用标准的下一令牌目标生成答案或描述。这些任务不使用任务专用预测头或输出表示。

指点与计数。 我们使用由粗到细的层级结构表示图像中的位置。对于每个点，模型首先选择一个区域令牌，然后选择一个分配给该区域的 ViT 图像块，最后选择所选图像块内的 4×4 个空间单元。区域选择会将大语言模型生成的第一个指点令牌的隐藏状态与区域令牌的大语言模型隐藏状态中的键进行匹配；图像块选择会将第二个指点令牌的隐藏状态与分配给所选区域令牌的图像块的保留视觉Transformer图像块特征中的键进行匹配；最后，第三个指点令牌通过线性分类器预测所选图像块内的分箱。利用保留的图像块到区域的分配关系，将预测分箱的中心映射回图像坐标。这些决策通过专用分类头以自回归方式生成，并在每个层级进行交叉熵监督。模型通过学习得到的点序列结束类别来终止点序列。对于计数，模型使用相同的定位表示，为每个符合查询条件的实例预测一个点。随后，模型以文本形式生成点集的基数。因此，计数以一组明确定位的实例为依据，而非仅通过直接标量回归完成。

分割。 对于指代分割，模型首先为目标掩码选择一个锚定区域，并使用目标掩码覆盖率最高的区域进行监督。以此锚定区域为条件，模型随后生成一个特殊的分割词元，其隐藏状态用作查询。区域逻辑值是该查询与区域令牌键的点积。图块逻辑值是该查询与每个保留的 ViT 图像块特征经线性投影后的点积。单元逻辑值是该查询与每个图像块的 16 个单元特征的点积，

表5：**RVLM 的训练超参数**。两个阶段均联合训练视觉编码器、连接器和大语言模型。阶段2从阶段1的最终检查点初始化。

设置	阶段1	阶段2
训练步数	31,971	24，000
全局批次大小/微批次大小	128 / 8	128 / 8
打包序列长度	3072	3072
最大训练裁剪数	14	14
最大打包图像数	28	28
优化器	AdamW	AdamW
β_1, β_2	(0.9, 0.95)	(0.9, 0.95)
ε	10^{-6}	10^{-6}
权重衰减	0	0
ViT 学习率	1×10^{-5}	2×10^{-5}
大语言模型学习率	2×10^{-5}	2×10^{-5}
连接器学习率	2×10^{-4}	2×10^{-5}
ViT 预热	2000	200
大语言模型预热	2000	200
连接器预热	200	200
学习率调度	余弦	余弦
最终学习率比例	0.1	0.1
梯度裁剪	1.0	1.0
硬件	16× NVIDIA B300	16×NVIDIA B300
分布式训练	FSDP	FSDP
精度	bf16	bf16

这些单元特征由一个线性层生成，该层将每个 ViT 图像块特征映射为每个 4×4 单元对应的一个特征；此处不使用额外的特征图。对每个分箱，将这三种分数相加，再利用保留的图像块到区域的分配映射回图像。随后将所得掩码概率通过双线性插值调整至原始图像分辨率，并以 0.5 为阈值进行二值化。通过重复锚点与掩码预测序列生成多个掩码。训练时，密集掩码采用区域选择监督，并结合焦点损失和 Dice 损失。

点到深度。 给定一张图像和一个或多个查询位置，模型会预测每个点的深度。查询点编码在提示中，输出则包含每个查询对应的由粗到细的点表示，随后是一个可测量性指示符；若该点可测量，还会给出以米为单位的深度。深度以文本形式生成，精确到小数点后两位。

A.4 评估

我们在35项任务上进行评估，涵盖图像问答（包括文档/OCR问答）、指点、计数、指代分割和点到深度。表7列出了每个数据集、结果中报告的指标，以及对任务的一句话描述。除非另有说明，分数均为百分比，分数越高表示性能越好。

B 补充实验

B.1 FH分割参数消融

我们研究 Felzenszwalb-Huttenlocher（FH）超参数中的尺度（k）、最小区域大小（min region size）和工作分辨率（最长边）如何影响下游准确率和视觉词元数量。所有实验均使用相同的 RVLM SFT检查点，仅改变推理时FH设置。我们在涵盖通用视觉问答、文档 OCR、指点、计数、指代分割和深度的六任务子集上进行评估。结果如表8所示。

GeneralVQA 和深度具有鲁棒性。 A-OKVQA DA 在所有设置下均保持在 74.8-75.5 之间，NYU $\delta_{1.25}$ 则保持在 97.4-97.6 之间。对于这些任务，FH 主要改变词元压缩，而不会影响准确率。

表6：**训练数据混合**。图像块模型和区域模型共用这些数据混合。

任务类别	数据集
阶段1（多模态预训练）	
图像描述	PixMo-Cap
OCR	olmOCR-mix
指点 / 计数	PixMo-Points、PixMo-Count、CoSyn-point、COCO、LVIS
分割	COCO、LVIS
深度	SUN RGB-D、NYU Depth v2、DIODE、HM3D、DDAD、Waymo、Argoverse 2、nuScenes、Aria Digital Twin、Aria Everyday Activities、Aria Nymeria、EgoExo4D、Stereo4D
相机	Stereo4D、SUN RGB-D、NYU Depth v2、EgoExo4D、DDAD、DIODE、nuScenes、Argoverse 2、Waymo、ScanNet++、ARKitScenes、iBims-1、Aria Nymeria
纯文本自然语言处理	Tulu-4
阶段2（监督微调）	
图像问答	VQAv2、A-OKVQA（多项选择和直接回答）、OK-VQA、ScienceQA、AI2D、TallyQA
图像描述	PixMo-Cap、PixMo-CapQA、PixMo-AskModelAnything
OCR	TextVQA、ChartQA、DocVQA、InfoVQA、ST-VQA、TabMWP
指点 / 计数	PixMo-Points、PixMo-Count、CoSyn-point
分割	PixMo-Points、COCO、LVIS、EntitySeg
深度	SUN RGB-D、NYU Depth v2、DIODE、HM3D、DDAD、Waymo、Argoverse 2, nuScenes、Aria Digital Twin、Aria Everyday Activities、Aria Nymeria、EgoExo4D, Stereo4D
纯文本自然语言处理	Tulu-4

文本密集型理解受益于更细的分割。 较粗的分割会损害文档 OCR。将 k 从 100 提高到 400 会使 InfoVQA 下降 2.2 分，将最小尺寸从 32 提高到 128 则会使其下降 3.6 分。相反，提高 FH 的工作分辨率会带来帮助，当最大边为=1024时，得分可达到 75.2。

随着粒度变粗，定位性能会平缓下降。 当区域变得更粗或 FH 工作分辨率降低时，PixMo-Point、PixMo-Count 和 RefCOCO 的得分通常会变化 ≤ 1.5 点。激进压缩（k=400 或最大边长=384）以少量定位性能损失为代价，大幅减少令牌数量。

令牌数量随 FH 粒度变化。</style}平均视觉令牌数从 k=50 时的 1171 降至 k=400 时的 686；最小尺寸为=16 时为 1032，降至 128 时为 736。增大最大边长主要会增加高分辨率光学字符识别/指点图像的令牌数量（InfoVQA：从 384→1024 时的 1053→1669 ），而较小的自然图像在最大边长超过其原生尺寸后，令牌数量便趋于饱和。

实用启发式。 在评估中，我们对所有任务采用同一 FH 设置（k=100、最小区域大小 32、最大边长 768；第 3.1 节）。对于文本密集型输入，建议降低 k 或最小区域大小；当令牌预算比精细细节更重要时，则建议调高这些参数。工作分辨率主要影响小字（InfoVQA：最大边长为 384 时得分为 71.0，最大边长为 1024 时为 75.2）。

B.2 连接器宽度消融实验

区域连接器在每个区域标记中最多聚合 w 个 ViT 图像块。包含超过 w 个图块的区域会拆分为多个连接器行，而较短的行则会填充至宽度w。因此，增大 w 可以减少传递给大语言模型的视觉令牌数量，但也会增大填充后的连接器输入。我们通过分析这种结构上的权衡，来确定默认值 $w = 16$。具体而言，我们在 FH 配置固定为 (k，最小尺寸，最大边长) =(100，32，768) 的情况下，对 $w \in \{4, 8, 16, 32, 64, 128\}$ 进行遍历。我们的分层抽样包含

表7：**我们的主要评测套件**。评估指标与主文中原生分辨率对比所报告的指标一致。这35项任务涵盖自然图像、文档、图表、示意图、信息图、场景文本、图形用户界面截图、驾驶场景、自我中心视角、外部视角、立体视频帧，以及室内和室外RGB-D。

数据集	指标	Task
		图像问答
A-OKVQA（直接回答）	视觉问答得分	开放式知识感知视觉问题（直接作答）。
A-OKVQA（多项选择）	准确率	多项选择式知识感知视觉问题。
AI2D	准确率（透明）	带多项选择答案的图表科学问答。
OK-VQA	视觉问答得分	需要外部知识的开放式视觉问题。
ScienceQA	准确率	带图像的多模态科学问题（图像子集）。
VQAv2	视觉问答得分	开放式自然图像视觉问答。
ChartQA	宽松准确率	关于图表和绘图的问答。
DocVQA	ANLS	关于文档图像/扫描页面的问答。
InfoVQA	ANLS	关于信息图的问答。
TextVQA	视觉问答得分	需要读取自然图像中文本的问题。
		指点
OSWorld-G	总体准确率	GUI 定位：根据自然语言指令指向 UI 元素。
PixMo-Points	F1	对描述对象的所有实例进行开放词汇指点。
PointBench	平均准确率	多技能指点（空间、计数、推理、可操控等）。
SA-Co Gold	全频段 F1	SA-Co Gold 上的长尾概念指点（指点划分）。
ScreenSpot Pro	总体准确率	跨应用类别的专业桌面 GUI 定位。
ScreenSpot v2	平均准确率	移动设备 / 桌面设备 / 网页用户界面元素定位。
		计数
CountBenchQA	精确计数准确率	根据自然语言问题描述的物体进行计数。
PixMo-Count	精确计数准确率	统计自然图像中查询类别的实例数量。
		分割
PixMo-Seg	CGF1	开放词汇指代分割（掩码预测）。
RefCOCO	总体交并比	COCO 上的指代表达分割（RefCOCO）。
RefCOCO+	总体交并比	不含位置词的指代表达分割。
RefCOCOg	总体交并比	针对较长表达式的指代表达分割。
SA-Co Gold	平均 CGF1（组平均）	SA-Co Gold 上的长尾概念分割（分割划分）。
		点到深度
Argoverse 2	尺度不变 $\delta_{1.25}$	估计自动驾驶场景中查询的二维点处的深度。
Aria AEA	尺度不变 $\delta_{1.25}$	Aria Everyday Activities 第一人称视角视频上的点到深度。
Aria Nymeria	尺度不变 $\delta_{1.25}$	Aria Nymeria 第一人称视角录制内容上的点到深度。
DDAD	尺度不变 $\delta_{1.25}$	自动驾驶密集深度上的点到深度。
DIODE	尺度不变 $\delta_{1.25}$	密集室内/室外深度上的点到深度。
EgoExo4D（自我）	尺度不变 $\delta_{1.25}$	来自 EgoExo4D 自我中心相机的点到深度。
EgoExo4D（Exo）	尺度不变 $\delta_{1.25}$	来自 EgoExo4D 外部视角相机的点到深度。
nuScenes	尺度不变 $\delta_{1.25}$	nuScenes 驾驶场景中的点到深度。
NYU Depth v2	尺度不变 $\delta_{1.25}$	室内 RGB-D 场景中的点到深度。
Stereo4D	尺度不变 $\delta_{1.25}$	互联网立体视觉／视频推导深度中的点到深度。
SUN RGB-D	尺度不变 $\delta_{1.25}$	SUN RGB-D 室内场景中的点到深度。
Waymo	尺度不变 $\delta_{1.25}$	Waymo开放数据集驾驶场景中的点到深度。

来自 A-OKVQA 直接回答、InfoVQA、PixMo-Point、PixMo-Count、RefCOCO 和 NYU Depth v2 的 1，480 张图像。表 9 总结了我们的结构分析。

大多数区域都能容纳在 w=16内。在 w=4 时，31.1% 的 FH 区域需要拆分，而在 w=16 时这一比例仅为 7.8%（拆分情况见表9）。因此，默认宽度可使 92.2% 的区域无需拆分，为进一步减少视觉令牌留下的空间有限。

更宽的连接器会带来大量填充。额外节省的令牌是以更大的填充后连接器输入为代价的。将 w 从 16 增至 128，视觉令牌仅减少 19.3%，而填充比例则从 62.2% 升至 94.1%。因此，估算的 bf16 连接器输入内存从 68.9 MiB 增至 444.8 MiB，增加了 $6.5\times$ 。这一权衡促使我们将 $w = 16$ 作为默认连接器宽度。

视觉词元缩减在 w=16 附近趋于饱和。表 9 显示，将 w 从 4 增至 16，可将视觉令牌平均数从 1，814 降至 980，降幅为 46.0%。这相当于 $w = 4$ 与 $w = 128$之间观测到的令牌总缩减量的 81.5%。将宽度增至 16 以上，收益递减。

表8：**FH 参数消融**。默认设置以蓝色突出显示。Token 数量是六项任务的平均视觉 Token 数量。NYUv2 报告采用逐图像中位数尺度对齐的尺度不变 δ_1 。

设置	A-OKVQA	InfoVQA	PixMo-Point	PixMo-Count	RefCOCO	NYUv2	令牌数量
尺度 k （最小尺寸 $=32$，最长边 $=768$）							
k=50	75.5	74.3	86.6	91.9	82.4	97.6	1171
k=100	74.8	74.3	86.4	90.4	82.5	97.5	963
k=200	75.4	72.8	85.8	90.9	82.2	97.5	794
k=400	75.1	72.1	85.1	89.6	82.0	97.4	686
最小区域尺寸（$k=100$，最长边 $=768$）							
最小尺寸=16	74.9	74.9	86.5	90.9	82.4	97.4	1032
最小尺寸=32	74.8	74.3	86.4	90.4	82.5	97.5	963
最小尺寸=64	75.5	72.2	85.8	90.6	82.4	97.4	860
最小尺寸=128	75.2	70.7	85.9	91.3	82.3	97.4	736
FH 工作分辨率（$k=100$，最小尺寸 $=32$）							
最长边=384	75.1	71.0	85.1	90.4	82.4	97.4	746
最大边长=512	75.0	71.6	85.9	91.1	82.5	97.4	830
最大边长=768	74.8	74.3	86.4	90.4	82.5	97.5	963
最大边长=1024	74.9	75.2	86.7	90.7	82.5	97.5	1060

表9：**连接器宽度的结构效应 w**。统计数据汇总了六项任务中的 1,480 张图像。输入内存估算单张图像填充后的 bf16 连接器输入所需的存储空间；不包括模型参数和中间激活。

w	视觉令牌	每个标记的图像块	拆分比例	填充	输入内存（MiB）
4	1,814	3.21	31.1%	18.4%	31.9
8	1,243	4.96	16.5%	40.4%	43.7
16	980	6.63	7.8%	62.2%	68.9
32	863	7.79	3.5%	78.5%	121.3
64	812	8.43	1.5%	88.6%	228.4
128	791	8.73	0.7%	94.1%	444.8

B.3 分辨率缩放

不同图像分辨率下的性能。表10 将原生分辨率评估细分为五个图像尺寸范围。在较低分辨率下，RVLM 和 PVLM 的表现相当，但随着图像尺寸增大，差距也随之扩大。在最小范围（0.14-0.41 MP）内，两个模型的按样例加权得分均为 85.3，RVLM 使用的视觉令牌减少了 4.5 %。在最大范围（2.08-2.63 MP）内，RVLM 的性能提升了 5.3 个点，同时平均令牌数减少了 51.8%。按类别平衡的增益也从这些范围内的 0.3 个点增至 9.0 个点。

这一优势在空间任务上尤为明显。在最高分辨率范围内，RVLM 在指点、计数、分割和深度估计任务上的平均分分别提高了 5.6、25.9、8.7 和 5.5 分。图像问答的表现总体上仍大致相当。这些结果表明，区域令牌能够更充分地利用高分辨率视觉信息，尤其是在预测依赖精细空间细节时。由于每个分辨率范围包含不同的任务样例，因此不应直接比较各范围之间的绝对分数；有意义的比较是每个范围内 PVLM 与 RVLM 之间的对比。

不同图像分辨率下的推理成本。表11衡量了高分辨率裁剪图像数量增加时的推理成本。在裁剪图像数量为 1 和 4 时，PVLM 在批量大小为 1 时略快，因为较短视觉序列带来的节省尚不足以抵消区域池化的开销。在批量大小为 8 时，裁剪图像数量为 4 时两种模型的延迟几乎相同。从 8 张裁剪图像起，RVLM 在两种批量大小下都能实现更低的延迟和峰值内存。

表10：**PVLM 与 RVLM 在五个原生分辨率范围内的对比**。任务指标根据各分辨率范围内的所有示例计算。短横线（–）表示该数据集在此范围内不包含图像。

Task	0.14–0.41 MP		0.55–0.76 MP		0.83–1.38 MP		1.55–2.01 MP		2.08–2.63 MP	
	PVLM	RVLM	PVLM	RVLM	PVLM	RVLM	PVLM	RVLM	PVLM	RVLM
所有任务										
按示例加权平均	85.3	85.3 (0.0)	69.8	70.2 (↑0.4)	72.6	73.8 (↑1.2)	69.0	73.6 (↑4.6)	68.4	73.7 (↑5.3)
家族均衡平均值	85.0	85.3 (↑0.3)	78.5	79.4 (↑0.9)	76.9	77.9 (↑1.0)	76.5	81.9 (↑5.4)	70.4	79.4 (↑9.0)
令牌数量	691	660 (↓4.5%)	987	797 (↓19.3%)	1,608	1,076 (↓33.1%)	2,499	1,358 (↓45.6%)	3,277	1,581 (↓51.8%)
图像问答										
A-OKVQA 直接回答	75.5	74.8	–	–	–	–	–	–	–	–
A-OKVQA 多项选择	88.6	88.2	–	–	–	–	–	–	–	–
AI2D	94.9	94.5	96.7	94.8	95.0	98.3	96.2	97.4	96.0	96.0
ChartQA	67.8	61.9	84.7	84.2	85.0	84.4	100.0	100.0	100.0	100.0
DocVQA	97.2	99.3	95.8	92.6	89.8	92.3	94.4	90.4	93.2	92.9
OK-VQA	65.8	66.0	–	–	–	–	–	–	–	–
ScienceQA	98.4	98.0	97.6	97.3	–	–	–	–	–	–
TextVQA	94.1	96.1	85.4	83.0	85.5	85.1	–	–	80.0	80.0
VQAv2	87.2	87.1	–	–	–	–	–	–	–	–
任务平均值	86.8	86.7	88.7	87.8	86.0	85.8	95.1	92.9	93.2	92.9
令牌数量	689	662	984	612	1,614	1,098	2,483	1,260	3,359	1,636
指点										
OSWorld-G	–	–	–	–	71.7	73.8	–	–	43.6	43.3
PixMo Point	81.2	81.6	90.6	89.5	86.0	86.6	83.1	87.3	88.1	94.0
PointBench	71.0	71.6	56.7	56.9	68.1	65.2	62.5	64.9	59.8	58.7
SA-Co Gold	91.2	91.6	88.6	89.3	90.4	91.5	87.6	90.6	78.3	87.4
ScreenSpot Pro	–	–	–	–	–	–	–	–	29.3	26.0
ScreenSpot v2	33.3	33.3	30.4	30.7	31.7	31.7	30.6	28.0	79.8	82.8
任务平均值	90.4	90.8	86.9	87.5	88.9	89.8	83.7	86.7	68.5	74.1
令牌数量	719	596	990	838	1,620	1,080	2,497	1,380	3,279	1,544
计数										
CountBenchQA	90.8	91.8	87.2	87.2	94.9	94.9	100.0	100.0	61.5	92.3
PixMo Count	91.9	91.9	84.4	93.3	89.9	91.3	74.5	90.9	61.9	87.3
任务平均值	90.9	91.8	85.7	90.5	91.7	92.6	75.3	91.1	61.9	87.8
令牌数量	552	500	1,004	855	1,649	1,129	2,729	1,433	3,354	1,611
分割										
PixMo Seg	53.9	52.5	67.8	65.0	57.1	58.1	58.9	65.9	48.6	70.8
RefCOCO	82.0	82.5	–	–	–	–	–	–	–	–
RefCOCO+	78.5	78.7	–	–	–	–	–	–	–	–
RefCOCOg	80.3	81.1	–	–	–	–	–	–	–	–
SA-Co Gold	42.9	43.1	40.9	41.9	46.0	48.4	41.3	46.7	47.6	56.2
任务平均值	64.3	64.6	41.9	42.7	46.5	48.9	43.9	49.6	47.6	56.3
令牌数量	709	669	987	843	1,610	1,077	2,499	1,386	3,209	1,596
深度										
Argoverse 2	–	–	–	–	–	–	–	–	77.9	81.8
Aria AEA	–	–	–	–	68.5	69.5	–	–	–	–
Aria Nymeria	–	–	–	–	54.6	55.6	–	–	–	–
DDAD	–	–	–	–	–	–	–	–	81.7	87.7
DIODE	–	–	–	–	92.2	94.0	–	–	–	–
EgoExo4D 自我	89.1	88.9	–	–	–	–	–	–	–	–
EgoExo4D Exo	–	–	84.0	82.9	–	–	–	–	–	–
nuScenes	–	–	–	–	–	–	84.7	88.8	–	–
NYU Depth v2	97.8	97.5	–	–	–	–	–	–	–	–
Stereo4D	89.0	88.6	–	–	–	–	–	–	–	–
SUN RGB-D	97.5	97.7	99.0	98.8	–	–	–	–	–	–
Waymo	–	–	–	–	–	–	84.2	89.8	83.9	91.9
任务平均值	92.8	92.6	89.2	88.5	71.4	72.7	84.5	89.1	80.6	86.1
令牌数量	634	542	955	734	1,372	821	2,321	1,125	3,254	1,533

随着分辨率提高，效率差距继续扩大。在裁剪图像数量为 32、批量大小为 1 时，RVLM 将生成延迟从 162.1 毫秒降至 105.0 毫秒。在批量大小为 8 时，延迟从 144.6 毫秒降至 83.7 毫秒，同时 GPU 峰值内存从 39.57 GiB 降至 30.15 GiB。这些趋势反映出，随着图像分辨率提高，减少大语言模型处理的视觉序列所带来的益处也不断增加，尽管视觉编码器仍会处理完整的图块网格。

表11：**不同图像分辨率下的推理成本**。PVLM 和 RVLM 在批次大小为 1 和 8 时的生成延迟、端到端（E2E）延迟及峰值 GPU 内存。测量使用单张 NVIDIA H100，对 512 个随机抽取的 PixMo-Cap 示例进行，包含 64 个预热批次和五次运行。生成上限为一个输出词元。延迟以均值 ± 标准差报告。裁剪图像数量较少时，PVLM 略快；但随着图像分辨率提高，RVLM 的延迟和内存节省幅度不断增加。

裁剪图像数量	生成延迟（毫秒）		E2E 延迟（毫秒）		峰值内存（GiB）	
	PVLM	RVLM	PVLM	RVLM	PVLM	RVLM
批次大小 1						
1	37.9±1.5	40.3±1.9	38.6±1.5	41.1±1.9	19.23	19.43
4	45.4±1.1	48.5±1.6	46.2±1.1	49.4±1.6	19.45	19.70
8	61.7±0.8	55.2±1.2	63.0±0.8	56.1±1.2	19.76	19.64
12	76.7±0.8	63.9±1.6	77.8±0.8	64.9±1.6	20.09	19.79
16	92.0±0.9	73.8±1.4	93.0±0.9	74.9±1.4	20.36	19.96
24	127.8±1.2	88.9±1.3	129.0±1.2	90.1±1.3	20.98	19.95
32	162.1±1.4	105.0±1.3	163.3±1.4	106.2±1.3	21.60	20.02
批次大小 8						
1	13.3±0.0	14.5±0.1	13.4±0.0	14.7±0.1	20.86	22.00
4	24.5±0.1	24.4±0.1	24.7±0.1	24.6±0.1	22.63	24.20
8	39.2±0.1	30.8±0.1	39.4±0.1	31.0±0.1	25.03	23.72
12	55.1±0.1	40.9±0.1	55.3±0.1	41.2±0.1	27.44	24.98
16	71.2±0.2	50.9±0.1	71.5±0.2	51.2±0.2	29.86	26.30
24	106.4±0.5	65.8±0.2	106.8±0.5	66.2±0.3	34.70	26.94
32	144.6±0.4	83.7±0.4	145.0±0.3	84.2±0.4	39.57	30.15

表 12：**展示区域拆分的重要性，并确保图块分配完整且互斥**。直接用区域令牌替换图像块令牌，在 PixMo-Point 上会导致灾难性失败（第 2 行）。确保区域令牌覆盖的图块数量不超过上限，且划分完整且互斥（第 3 行），使 RVLM 在指点任务上的表现优于基于图块的基线。结果是在 17 亿规模下报告的。

表征单元	直接坐标输出	由粗到细预测
图块	77.5	84.4
区域（朴素方法）	72.7	10.0
区域（RVLM）	n/a	86.6

B.4 用于定位的区域令牌要求

要支持精确的空间预测，区域令牌必须满足两个属性。每个图像块必须且只能属于一个区域，确保每个目标都可被找到且明确无歧义；同时，每个区域包含的图像块数量不得超过解码器可供选择的数量，确保每个图像块都可以被选中。我们在 1.7B 开发运行中测试了这些属性，在两种指点形式下用区域令牌替换图像块令牌（表 12）。采用直接坐标输出时，区域的得分比图像块低 4.8 分，因为大语言模型必须根据空间范围各异的令牌回归出精确位置。由粗到细的解码消除了这一负担，但使用未经修改的区域时，得分仅为 10.0，因为这些区域违反了上述两项属性。一些图像块不属于任何区域，或同时属于多个区域；而第二阶段会为每个区域考虑固定数量的图像块，因此无法选中超大区域中的图像块。让每个图像块都恰好属于一个区域，并将大型区域拆分为连续分组（第 3.1 节），即可恢复这两项属性，此时区域的表现超过图像块（86.6 对 84.4）。

B.5 向未见领域的泛化

我们在病理学和卫星基准测试上评估 PVLM 和 RVLM，而这两种模型均未在这些基准上训练（表13）。由于两种模型仅在令牌化方式上有所不同，这直接检验了 FH 区域能否迁移。RVLM 在 PathMMU（+1.7）、PathVQA（+0.6）和 LoveDA（+0.8）上表现更佳

表13：**在未见领域上的表现**。RVLM在病理学和卫星图像上的表现与PVLM不相上下，这些领域在训练期间均未见过。报告的令牌数量是每张图像的平均视觉令牌数（PVLM → RVLM）。

基准测试（指标）	PVLM	RVLM	令牌数量
病理学			
PathMMU（准确率）	45.3	47.0	1842 → 972
PathVQA（准确率）	63.1	63.7	1850 → 869
卫星影像			
RSVQA-HR（准确率）	47.9	47.3	1829 → 853
LoveDA（分割交并比）	62.6	63.4	1829 → 821

表14：**FH 在未见过的领域中进行分割，比 SAM 更可靠**。 FH 在每个数据集上的边界召回率（BR）都更高，并且从构造上保证覆盖每个像素，而 SAM 会留下 13–29% 的像素未被覆盖。

数据集	SAM BR	FH BR	SAM 覆盖率	FH 覆盖率
LoveDA	39.0	54.8	71.0%	100%
Total-Text（单词）	34.2	64.2	87.1%	100%
Total-Text（笔画）	60.7	83.5	87.1%	100%

在 RSVQA-HR（−0.6）上则略逊一筹，同时使用的视觉令牌约为一半。我们还测量了 SAM ViT-H 和采用 RVLM 设置的 FH 相对于真实掩码的边界召回率（BR）（表14）。FH 在每个域外数据集上的 BR 都更高，并且从构造上覆盖每个像素，而 SAM 则有 13–29% 的像素未被覆盖。

B.6 替代位置编码

区域标记的内部形状在池化之前就已通过添加到每个图像块特征上的正弦位置嵌入进行编码（第 3.1 节）。因此，提供给大语言模型的位置只需将区域锚定在图像中，任何位置方案都能做到这一点。我们在 1.7B 规模下比较了两种此类方案（表 15）。我们的默认方案将大语言模型原生的一维 RoPE 扩展为交错的三维 M-RoPE，并将每个区域标记放置在距离该区域中心最近的图像块上。另一种方案保留一维 RoPE，并添加根据每个区域的掩码计算出的绝对位置嵌入，该嵌入通过学习型线性投影进行编码。在图像描述、指点和计数任务上，两者的差距均不超过 0.3 个百分点，因此 RVLM 并不需要原生支持多模态旋转位置编码的骨干网络。我们采用多模态旋转位置编码，因为它无需增加参数，而掩码嵌入则需要。

B.7 模型规模

我们使用 1.7B 大语言模型进行图像块与区域的对比（表 16 和表 17）。阶段 1 之后，RVLM-1.7B 的指点表现提高了 2.2 个点，而图像描述和计数表现与原先相差不超过 0.3 个点。阶段 2 之后，其在 COCO 和 A-OKVQA 上的表现提高了 0.7 至 1.0 个点，文本密集型任务的得分变化为 −1.1 至 +1.0 个点（DocVQA、ChartQA、InfoVQA）。

B.8 纯文本与视频评估

如表 18 所示，区域词元化基本不会改变纯文本能力（变化在 0.6 个点以内）。如表 19 所示，在视频任务上（两个模型均未针对该任务进行训练），RVLM 的表现与 PVLM 相差不超过 0.5 个点，同时使用的视觉令牌减少了 3× 。我们将基于区域的视频语言模型的进一步开发与探索留待未来工作。

C 局限性

细粒度视觉内容。 RVLM 在 InfoVQA（−4.9点）、ScreenSpot Pro（−3.3）和 ChartQA（−0.8）上的表现不如 PVLM。这些任务通常需要辨别极小的文本或面部元

表 15：**比较区域令牌的位置编码**。 报告的结果为 RVLM-1.7B 在第一阶段训练后的结果。

位置编码	新增参数	PixMo-Cap	PixMo-Point	PixMo-Count
区域掩码绝对位置编码 + 1D 旋转位置编码	Yes	53.6	86.7	84.7
三维 M-RoPE（我们的方案）	No	53.8	86.6	84.4

表 16：**1.7B 模型的第一阶段结果**。

模型	PixMo-Cap	PixMo-Point	PixMo-Count
PVLM-1.7B	54.1	84.4	84.6
RVLM-1.7B	53.8	86.6	84.4

素，而这些元素在分割或图块分配过程中可能会被吸收到周围的背景区域中。尽管保留了完整图块覆盖，池化仍可能模糊这些细节。我们的 FH 消融实验（附录 B.1）表明，文本密集型任务对区域粒度很敏感，这说明调整最小尺寸、工作分辨率或其他 FH 参数或许能缓解这些问题。我们在所有任务中采用固定配置，以开发一种兼顾性能与词元效率的通用视觉语言模型。根据单张图像或任务要求调整分割粒度，或许能进一步改善这种权衡，但尚未对此进行探索。

文本坐标定位。 RVLM 的空间精度依赖于通过保留的图像块到区域对应关系进行定位词元解码。若改为将坐标生成为文本，区域令牌在 PixMo-Point 上会比图像块令牌低 4.8 个百分点（1.7B 时为 72.7，对比 77.5；附录 B.4），因为单个区域令牌最多可覆盖 16 个图像块。因此，输出文本坐标的视觉语言模型若要从区域令牌中获益，就需要采用定位词元解码。然而，正如 Clark 等人（2026a）所示，这种解码方式本身就比文本坐标更准确，同时每个点仅使用 3 个输出词元，而基于文本的坐标预测则需要 6–8 个。

基于图块的编码。 RVLM 改变了大语言模型所处理的单位，但视觉编码器仍会处理完整的图块网格，因此其计算成本仍随像素面积增长（附录 B.3）。原生以区域为单位运行并跳过冗余图块的编码器，则可以将效率提升扩展到视觉编码器。

架构范围。 我们在 MolmoPoint 架构中开发并评估了区域词元化（Clark 等人，2026a）。尽管这种方法本身并不特定于该模型，但我们尚未验证其在其他视觉语言模型架构中的有效性。具体而言，沿特征维度拼接固定数量图块嵌入的连接器，未经修改无法直接适配可变尺寸区域。将区域词元化扩展到其他连接器设计和长视频仍是未来的工作。

对光度边界的依赖。 FH 根据局部颜色差异构建区域，而这些差异可能与语义边界并不一致。这一局限在低对比度或退化图像中可能尤为突出，例如雾、伪装和运动模糊场景，因为有意义的结构可能缺乏明显的视觉边界。我们没有对这些情况进行专门评估。我们将每个区域限制为最多$w = 16$ 个图块，以此限定池化到单个词元中的内容量，但这无法恢复区域构建过程中遗漏的边界。

连接器宽度权衡。 我们选择 w=16，以平衡视觉词元缩减与连接器填充。虽然更大的宽度能进一步缩短大语言模型序列，但我们的结构消融实验表明，这会显著增加填充后的连接器输入（附录 B.2）。因此，所选宽度代表的是一个实用的工作点，而非普遍最优的设置。更高效地处理可变尺寸区域或许能改善这一权衡。

表 17： **1.7B 模型的第二阶段结果。**

模型	COCO	A-OKVQA (MC)	A-OKVQA (DA)	InfoVQA	ChartQA	DocVQA
PVLM-1.7B	80.5	79.4	62.5	56.3	77.6	85.5
RVLM-1.7B	81.5	80.4	63.2	57.3	77.0	84.4

表 18： **纯文本基准测试上的表现。**

模型	MMLU	ARC-C	GSM8K
PVLM-4B	67.7	87.6	89.0
RVLM-4B	67.8	87.0	88.9

表 19：**视频基准测试上的性能**。RVLM 使用的视觉令牌比 PVLM 少 3× 。这两个模型都没有在视频上进行训练。

模型	MVBench	MLVU	NeXT-QA
PVLM-4B	50.9	44.0	69.9
RVLM-4B	50.6	44.2	69.4

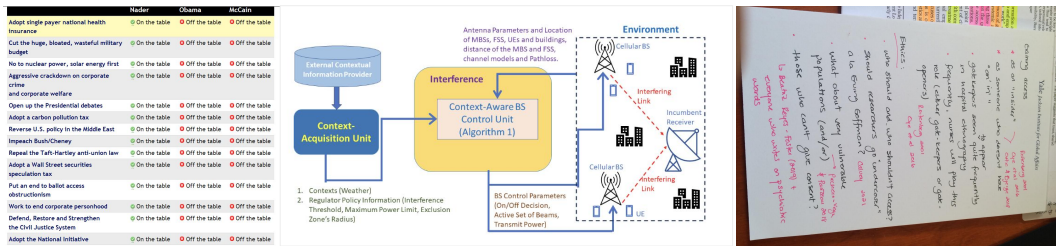


图5：**基于 T-REN 的 RVLM 的失败案例**。在视觉语言模型训练期间，区域令牌在空间上彼此脱节，导致难以将文本和视觉元素与其正确位置对应起来。模型能够识别单个标签，却会将其错误地归到相邻的行、列或物体上，从而错误描述图像结构。例如，在第一个示例中，尽管图像中没有列出任何举措，它却错误地描述说，有些举措是奥巴马“正在考虑的”。PVLM 和基于 FH 的 RVLM 没有出现这些问题。