arXiv:2611.32134v2 [cs.CV] 25 Nov 2026

EditText：结构感知生成与选择性视觉传输的细粒度文本编辑

ZhaobinChu[1, 2, *] Guandong Li[2, *] Xin Lv[1,2] Zhiwei Zeng[2]1中国科学技术大学，合肥，中国2科大讯飞，合肥，中国

zbchu@mail.ustc.edu.cn gdli7@iflytek.com, xinlv@iflytek.com, zwzeng@iflytek.com

*这些作者对本项工作贡献相同。



图 1. **跨三类文本的细粒度文本编辑**。从上到下，两行分别展示了字母文字、表意文字和阿拉伯数字，共 17 个示例。每对图像左侧为带有红色编辑框的源图像，右侧为对应结果。完整图像按照其原始宽高比紧密排列。字母文字示例包括短语长度变化、标志字母、密集的店面文字以及非常小的英文文本。表意文字示例包括描边字幕、弯曲的队徽、1场景牌匾、密集的正文，以及非常小的手写文本和屏幕文字。阿拉伯数字示例包括屏幕数字、数字数量变化、海报上的小文本，以及安装示意图中非常小的尺寸标签。

摘要

细粒度文本编辑旨在准确替换指定区域中的文本，同时保留源文本的字体、颜色和纹理，并维持未编辑区域的视觉一致性。我们提出 *EditText*，这是一种统一的细粒度文本编辑框架，将目标字符生成与源图像中的视觉传输相结合。该模型采用双条件填充输入：掩码图像和源文本风格参考，二者与目标图像共享空间参照。掩码图像遮盖原始文本，并将未编辑内容保留为源画布；风格参考则提供外观线索，与目标文本共同引导新字符生成。我们提出注意力传输框架，其中，源权重描述分配给某一条件的注意力，空间聚合核则描述其内部的空间注意力分布。准确迁移源细节取决于注意力强度、条件特征中细节的保留情况，以及空间聚合的准确性。因此，对于选定的非编辑查询向量，我们保留分配给掩码图像的总注意力权重，同时将其注意力重新分配到空间上对应的邻域，从而实现选择性视觉传输。通过多阶段训练，并结合针对局部目标字符结构的字形感知结构损失，模型能够生成准确文本、保留源文本外观并保留未编辑细节。我们还推出 *TBench*，这是一个涵盖字母文字、标志文字和阿拉伯数字的细粒度文本编辑基准，用于评估字符更新、外观保持和未编辑细节的保留效果。在 *TBench* 上，*EditText* 在非编辑区域结构相似性（*0.9769*）和全图学习感知图像块相似度（*0.0460*）方面均达到最先进水平。

1. 引言

图像中的文本呈现出多样的字符系统和视觉形式。字母文字常见于产品标签、描述性文本和场景标识。以汉字为例的表意文字具有丰富的笔画和部件结构。阿拉伯数字则传达时间、价格、数量和尺寸等精确信息。

这三类文本带来了各不相同的编辑挑战。对于字母文字，单词或短语长度的变化需要调整字符间距和空间范围

同时保留原始字体和布局样式。对于 Logo 图形文本，笔画与组件之间的复杂关系对局部字符结构提出了更高要求。对于阿拉伯数字，值和数字位数的变化要求在保留字体大小、对齐方式和视觉风格的同时，确保数值内容准确无误。密集布局和极小文本进一步加剧了这些挑战：模型必须更新细微的



图 2. **通过生成与视觉迁移进行文本编辑**。在橙色虚线框内，模型以原有字样风格将“0000”替换为“699”。绿色框标出了模型通过复用掩码源图像中的视觉特征而保留的屋顶和树叶细节。

字符结构，同时在有限区域内保留相邻文本和背景细节。

因此，细粒度文本编辑需要准确更新字符、保留源文本外观，并在不同文本类别、长度和尺度下保留未编辑细节。图1 展示了我们在这三个类别上的结果，包括长度变化、密集布局和极小文本。该方法在更新目标内容的同时，保留源文本外观及周边细节。

扩散模型和多模态生成式 Transformer 推动了图像合成与编辑的发展 [4, 16, 19]。文本感知方法进一步利用布局、字形和源文本提示来改进文本生成 [2, 11, 24, 25]。要在这三类文本中实现细粒度编辑，就需要有选择地使用源信息：未编辑内容应保持不变，源文本外观应予以保留，原有字符结构则应根据目标文本进行更新。

为应对这些挑战，我们提出了 EditText，这是一种通过字符生成和视觉传输实现文本编辑的统一框架（图 2）。双条件填充组织字符生成和源信息复用：掩码图像隐藏原始文本，同时保留未编辑内容；源文本风格参考则从掩码区域提供外观信息。两种视觉条件与目标图像共享空间参照，并结合目标文本共同引导生成。在编辑区域内，目标文本驱动新字符的生成，而

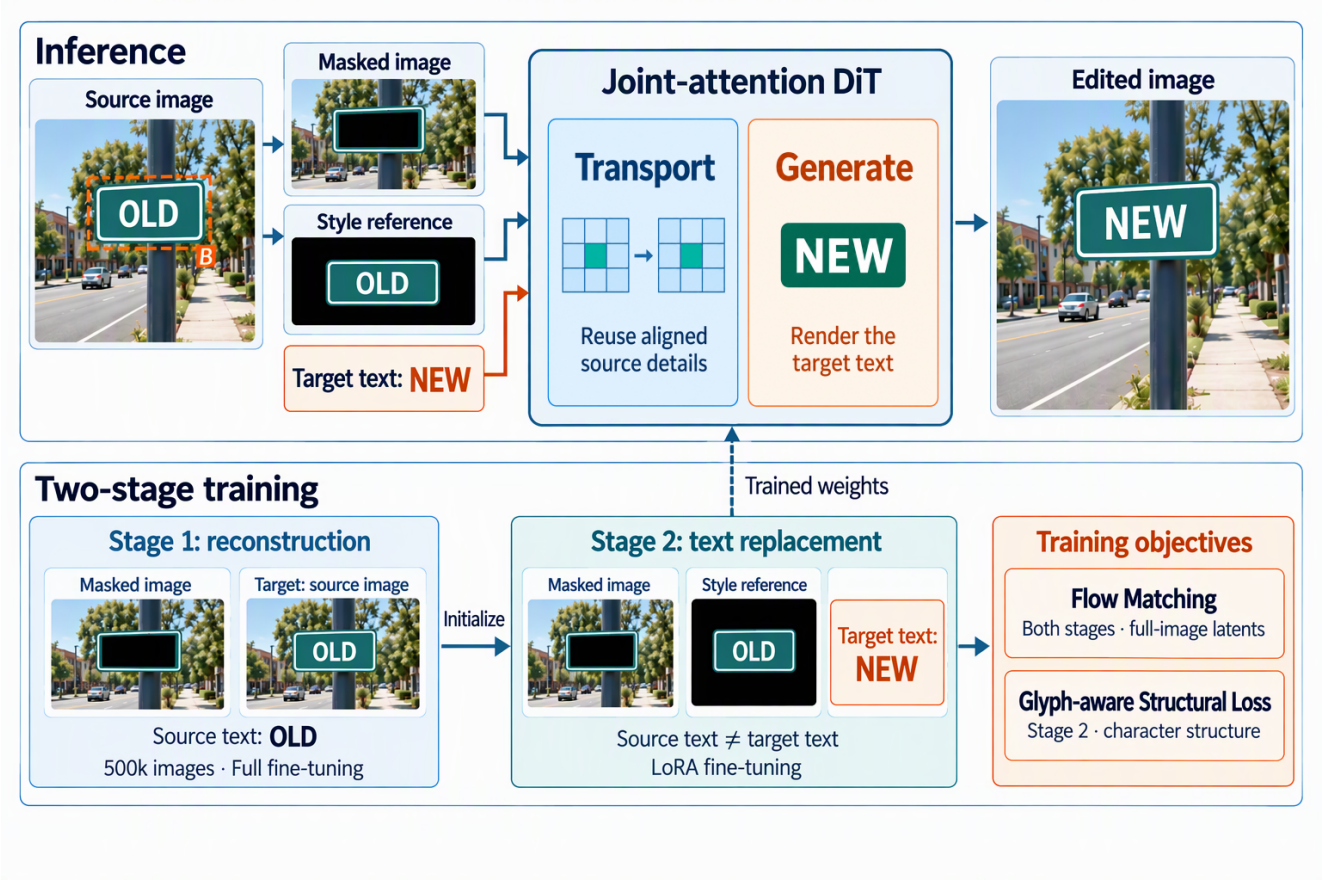

图3. **EditText 概览**。上：双条件 Fill 保留掩码图像中的未编辑内容，并通过源文本风格参考提供外观线索。两种视觉条件与目标图像共享空间参照，并与目标文本共同引导联合注意力 DiT。对于选定的非编辑查询，空间重新分配会保留分配给掩码图像的注意力权重总量 μ_i，同时将其内部空间分布从 π_{ij} 改为 π'_{ij}。下：阶段1在 50 万张真实图像上进行自监督掩码文本重建和全参数微调。阶段2加入风格参考，并通过 LoRA 学习文本替换。两个阶段都使用流匹配监督；阶段2还通过字形感知结构损失约束目标字符结构。

风格参考引导外观。在未编辑区域，模型从掩码图像中读取并复用源信息。这两项功能在同一个生成式主干网络中运行。

仅将条件作为语义或外观参考，并不一定能保留可按位置复用的局部信息。强烈的下采样会削弱精细结构和纹理，而缺失空间对应关系则会限制在匹配位置进行特征检索。我们将掩码图像视为源画布，强调保留源细节并与目标图像进行空间对齐，以支持在未编辑区域复用局部信息。为分析如何读取源画布，我们引入注意力传输框架：源权重衡量分配给某一条件的注意力，空间聚合核描述其中的空间注意力分布。仅凭注意力强度无法判断源细节是否被准确迁移；条件特征必须保留这些细节，且聚合必须发生在正确的空间位置。

基于这一观察，对于选定的非编辑查询向量，我们保留分配给掩码图像的总注意力权重，同时将其内部注意力重新分配至对应的空间邻域，从而实现选择性视觉传输。

编辑区域中的字符结构必须根据目标文本重新生成，同时保留源文本外观。为学习字符生成和未编辑内容的保留效果，我们首先在 500,000 张真实图像上进行自监督 Fill 重建和全参数微调，将每张原始图像作为目标，以恢复被掩码的文本并保留其周围内容。随后，我们加入源文本风格参考，并使用源字符串和目标字符串不同的示例进行文本替换监督微调（SFT）。这一阶段学习新字符

生成和外观保持，并通过字形感知结构损失，在全图目标之外强化对局部字符结构的监督。

我们的主要贡献如下：

• **我们提出了用于细粒度文本编辑的 EditText**。该框架将目标字符生成与源图像中的视觉传输相结合，并利用源画布和源文本风格参考来协调内容更新与外观保持。它通过两个训练阶段，将自监督重建与文本替换相结合，并采用字形感知结构损失来强化局部字符结构的学习。

• **我们构建了注意力传输框架，并提出了选择性视觉传输**。将针对源条件的注意力分解为源权重和空间聚合核后可以发现，源细节的迁移同时取决于条件特征中的细节保留以及空间检索的准确性。对于选定的非编辑查询向量，空间重新分配可强化从对应邻域进行的检索，同时保留分配给源画布的注意力权重总量，从而实现选择性视觉传输。

• **我们实现了兼顾文本更新、外观保持和全图细节保留的细粒度编辑**。该方法支持字母文字、表意文字和阿拉伯数字，包括长度变化、密集布局、极小文本以及复杂的表意文字结构。在全图生成过程中，该方法能够保留未编辑细节，并在 TBench 上取得了最先进的非编辑区域结构相似性指数（0.9769）和全图学习感知图像块相似度（0.0460）。

• **我们介绍了 TBench，这是首个用于细粒度文本编辑的基准**。它涵盖字母文字、表意文字和阿拉伯数字，以及长度变化、密集布局和极小文本，并系统评估字符更新、文本外观保持以及未编辑细节的保留效果。

2. 相关工作

文本生成与编辑。 TextDiffuser 将文本布局与基于扩散的生成相结合 [2]。AnyText 使用字形、位置和掩码图像进行多语言文本生成与编辑，而 AnyText2 则增加了字体和颜色控制 [24, 25]。Qwen-Image 和 Seedrea m 3.0 等基础模型支持中文文本生成，并提升了复杂字形渲染和文本布局的效果 [5,27]。Qwen-Image-Edit 将这些能力扩展到了中文文本编辑 [17]。SelfPromptSTE 根据源文本构建风格提示，通过渲染的字形提供目标结构，并借助上下文学习实现开放词汇编辑 [11]。EditText 采用双条件填充：掩码图像和源文本风格参考共享空间参考，分别提供源画布和文本外观线索。目标文本和文本替换监督信号引导新字符生成。

图像修复与基于指令的编辑。扩散概率模型和基于分数的建模为图像生成提供了基础 [7, 21], ， 而潜在扩散则支持高分辨率合成与编辑 [19]。Palette 研究了基于扩散的图像到图像转换 [20]。SDEdit、DiffEdit 和 RePaint 分别通过随机微分方程、掩码引导和已知图像内容实现编辑或图像修复 [3, 13, 14]。从 pix2pix 到 InstructPix2Pix，成对监督也被广泛用于学习图像转换 [1, 9]。细粒度文本编辑还要求准确生成离散字符，并保持文本外观。TextStyleBrush 从单个示例中迁移视觉文本风格 [10]。EditText 保留未编辑内容，并引入源文本风格参考，为掩码区域提供直接的外观信息。

注意力交互与视觉保持。 DiT 使用 Transformer 处理图像潜在标记，[16], 而多模态流模型则通过联合注意力实现图像和文本标记之间的双向交互 [4]。ControlNet 和 T2I-Adapter 通过条件分支或适配器提升空间控制能力 [15, 28]。即插即用扩散特征利用中间特征，在图像编辑过程中保持一致性 [23], ；Prompt-to-Prompt 则展示了注意力控制对编辑的影响 [6]。EditText 从注意力强度和空间注意力分布的角度分析条件特征聚合，考察空间对应关系和条件表示质量如何影响视觉保持，并在掩码图像内重新分配空间注意力。

3. 方法

3.1. 任务定义

给定一张源图像 $I_{\text{src}} \in \mathbb{R}^{H\times W\times 3}$、一个编辑区域 B 和一个目标字符串 y，我们将细粒度文本编辑形式化为

$$\hat{I} = \mathcal{F}_\theta(I_{\text{src}}, B, y). \tag{1}$$

编辑区域 B 由标注或用户绘制的框指定。模型必须生成目标字符串 y，保留源文本外观，并保持未编辑区域的视觉一致性。在单次生成过程中，EditText 在编辑区域生成目标字符，同时保留源文本外观，并传递源视觉信息，以保留未编辑区域中的内容。

设 Ω 表示完整图像域。编辑区域和非编辑区域为

$$\Omega_e = B, \qquad \Omega_n = \Omega \setminus B. \tag{2}$$

Ω_e 由编辑框或掩码定义，包含待替换的文本以及该区域内的背景像素。Ω_n 包含图像的其余部分，包括其他文本、对象和背景。模型在 Ω_e 中生成目标字符并保留源文本外观，同时在 Ω_n 中保留源内容。联合生成维持了两个区域之间的视觉连续性。

我们首先介绍用于字符生成的条件设计和字符结构监督，然后介绍用于视觉传输的源信息检索机制，最后介绍将二者结合起来的分阶段训练方案。

3.2. 字符生成与外观保留

字符生成在编辑区域 Ω_e 上进行：根据目标文本更新字符结构，同时使用源文本的字体、颜色和纹理。双条件 Fill 提供互补的内容和外观信息，而字形感知结构损失则强化了局部字符结构监督。

双条件 Fill 与源文本风格参考。
目标文本指定新的字符内容，而源文本风格参考则提供外观线索。我们将输入组织为 Fill 表述。令 M 为 B的二值掩码。对文本区域进行掩码处理后即可得到掩码图像。为提供来自掩码源文本的外观信息，我们构建风格参考，从而形成双条件 Fill:

$$\begin{aligned} I_m &= (1 - M) \odot I_{\text{src}} + M \odot c, \\ I_s &= M \odot I_{\text{src}} + (1 - M) \odot c, \\ \hat{I} &= F_\theta(I_m, I_s, y), \qquad c = 0. \end{aligned} \tag{3}$$

I_m 保留未编辑区域的内容和位置，作为视觉信息复用的源画布。 I_s在黑色背景上保留位于原始位置的源文本区域，为新字符生成提供字体、颜色、纹理和透视参考。提供给变分自编码器的两种视觉条件具有与目标图像相同的尺寸和空间布局，并在潜在网格上对齐。目标字符串通过自然语言指令提供，不额外输入目标字形。条件接口和坐标处理分别见附录 B.4 和 C。

风格参考 I_s 还包含原始字符的笔画和部件，无法直接用作目标字符结构。我们使用源文本和目标字符串各不相同的文本替换示例进行训练，以从 I_s 中学习外观线索，并根据 y生成新的笔画、部件和字符排列。两种视觉条件与目标文本通过联合注意力进行交互。单一主干网络生成完整图像，在生成过程中将字符更新与周围内容联系起来。

全图生成与局部结构监督。模型采用流匹配 [12]。令 $z_0 = E(I_{\text{tgt}})$ 表示目标图像潜变量。噪声插值和目标速度为

$$z_\sigma = (1 - \sigma)z_0 + \sigma\epsilon, \qquad v^* = \epsilon - z_0, \tag{4}$$

其中 $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$。基础流匹配目标函数为

$$\mathcal{L}_{\text{FM}} = \mathbb{E}\left[w(\sigma)\,\text{mean}[(v_\theta - v^*)^2]\right], \tag{5}$$

其中 $w(\sigma)$遵循基础调度器。

将式（2）中的区域映射到潜在网格后，每个样本的均方项可写为

$$\text{mean}[(v_\theta - v^*)^2] = \sum_{u \in \{e,n\}} \rho_u \ell_u, \quad \rho_u = |\Omega_u|/|\Omega|. \tag{6}$$

其中， ℓ_u 是区域均方误差， ρ_u 是面积占比。训练不采用区域重加权。全图均方目标函数按照区域面积的比例汇总误差。当编辑区域较小时，其中的字符误差对整体目标函数的贡献也与该区域面积成比例。因此，我们在文本替换训练期间引入字形感知结构损失，为目标字符提供额外的局部结构监督。

字形感知结构损失。 结构监督作用于预测的干净潜变量，而不是速度场的梯度或解码后的 RGB 图像。根据式（4），

$$\hat{z}_0 = z_\sigma - \sigma v_\theta. \tag{7}$$

我们将目标文本区域映射到潜在网格，并将其裁剪算子记为 C_B。令 D 表示逐通道水平和垂直 Scharr 梯度。我们定义

$$\begin{aligned} G_\theta &= D(C_B\hat{z}_0), \quad G_0 = D(C_B z_0), \\ s_B &= \text{sg}\left[\max\left(\sqrt{\text{mean}(G_0^2)}, s_{\min}\right)\right]. \end{aligned} \tag{8}$$

这里，sg 表示停止梯度，均值涵盖区域内的空间位置、潜在通道以及两个梯度方向。在训练期间，栅格化目标字形图会经历与目标图像相同的变换，并被调整为非负区域支持权重 $S(p)$。这

支持图仅用于损失计算，并非模型条件。每个样本的结构误差为

$$e_g(p) = \sqrt{\|(G_\theta(p) - G_0(p))/s_B\|_2^2 + \delta^2} - \delta,$$
$$R_g = \frac{\sum_p S(p) e_g(p)}{\max(\sum_p S(p), \varepsilon_S)}. \tag{9}$$

该范数涵盖通道和梯度方向，其中 δ 是 Charbonnier 平滑常数。目标图像潜变量提供结构监督，而字形支持图定义空间权重。因此，监督信号保留的是目标图像的实际外观，而不是渲染字体的像素形状。

为限制辅助项对全图重建的影响，令 ℓ_{FM} 表示每个样本的流匹配损失。我们采用噪声门控和自适应幅度系数：

$$g(\sigma) = 4\sigma(1-\sigma),$$
$$c_g = \text{sg}\left[\min\left(1, \frac{\text{sg}[\ell_{\text{FM}}]}{\text{sg}[R_g] + \varepsilon_R}\right)\right], \tag{10}$$
$$\mathcal{L}_{\text{glyph}} = \mathbb{E}[g(\sigma) c_g R_g].$$

该门控在中等噪声水平下较大，并在两个端点处趋近于零。令 $\lambda_g \geq 0$ 表示结构损失系数；完整的训练目标见第 3.4节。对于非负的 ℓ_{FM}，辅助项满足$0 \leq \lambda_g g(\sigma) c_g R_g \leq \lambda_g \text{sg}[\ell_{\text{FM}}]$，从而限制其损失幅度，但不限制其梯度范数。实现细节见附录 B.3。

3.3. 视觉传输与细节保留

视觉传输作用于非编辑区域 Ω_n ，以便在全图生成过程中保留周围文本、纹理和背景结构。尽管 Fill 输入保留了这些内容，网络仍必须在正确的输出位置检索源信息；仅提供源图像条件并不能确保细节得到准确重建。

观察真实图像中的空间检索。图 4比较了未编辑文本区域中的输出。使用单条件参考模型时，我们固定模型权重、其他输入和随机种子，仅在水平方向循环移位条件潜变量。这样既保留了源词元值集合，又改变了它们与输出位置之间的对应关系。对齐和移位后的输出整体布局相似，但文本笔画和边缘出现局部变化；差异图使用相同色阶标出了这些变化的位置。在评估的所有图像中，移位 1、2 和 4 个词元都会增加未编辑区域相对于源图像的误差。因此，仅有源内容并不足以保留细节；模型对空间位置的使用同样重要。循环移位还会引入边界环绕。完整设置和解释范围见第 6.2 节及附录 E。

从空间对应关系到注意力传输。这些观察引出了两个不同的问题：输出位置会将多少注意力分配给源图像，以及该注意力会在源图像的何处聚合信息。前者描述源权重；后者决定空间检索分布。即使检索位置正确，在条件编码过程中丢失的细节也无法直接复用。因此，注意力传输分析源权重、空间聚合位置和特征表示质量，并由此推动采用保留源权重的选择性视觉传输。空间错位测试对应关系的作用；完整方法的组件消融实验则评估由此带来的编辑增益（表 3）。

源画布与空间对应关系。我们将掩码图像视为源画布，利用共享的空间参照，为非编辑查询向量提供对应的源邻域。注意力传输描述了这一检索过程中的注意力分配和特征聚合。令 p_i， p_j 表示输出与条件的图像词元坐标，并令 $R(p)$表示旋转位置编码。对于旋转位置编码[22], 的相对位置形式，注意力分数满足

$$\langle R(p_i)q_i, R(p_j)k_j \rangle = q_i^\top R(p_j - p_i)k_j. \tag{11}$$

相对位移参与查询–键交互，为从对应邻域聚合源特征提供位置参考。多图像序列还包含图像身份等非空间分量，因此注意力取决于位置和内容。

EditText 对条件和目标图像应用一致的空间变换，以保持坐标对应关系。空间对齐方式指定检索位置，而细节复用还取决于条件特征是否保留源信息。风格参考保留源文本位置和外观；原始字符与目标字符的笔画不必一一对应。

源权重与空间聚合核。考虑任意层中一个注意力头的输出查询 i 。设 a_{ij} 为对所有可见键进行归一化后的注意力

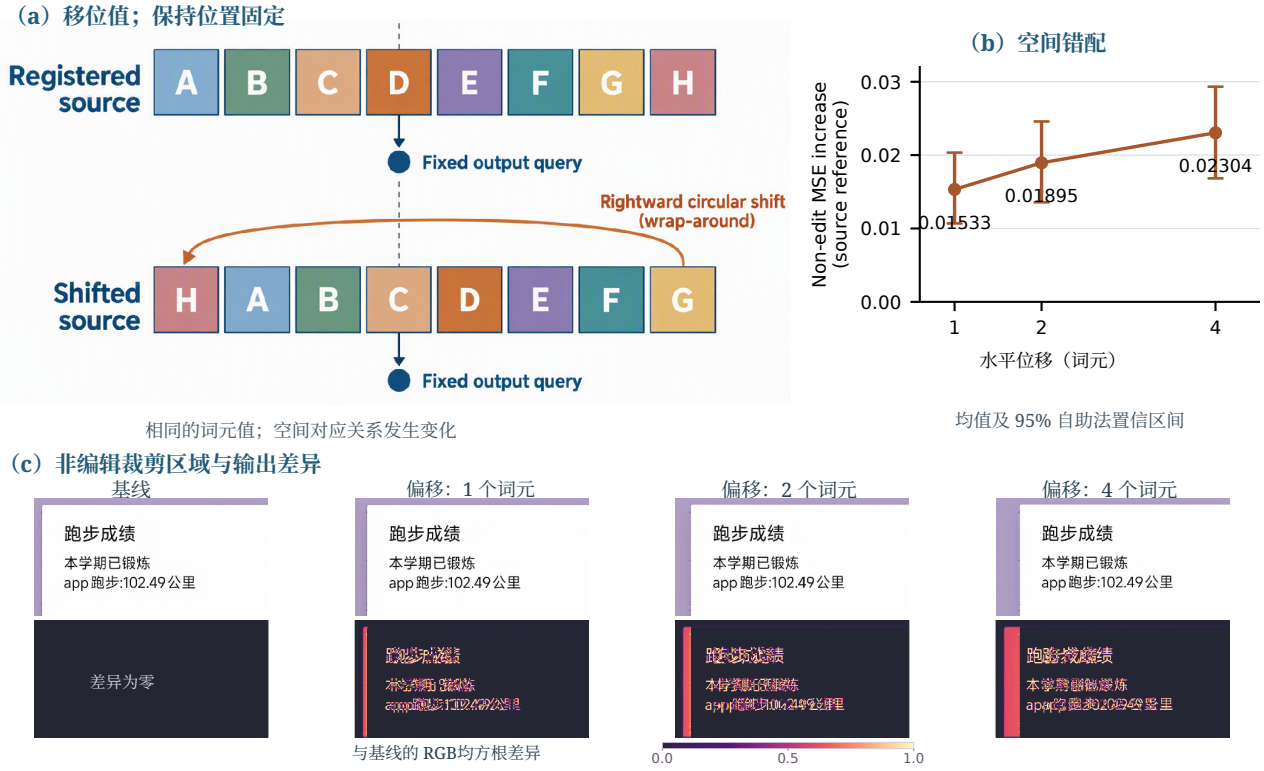

图 4. **空间对应关系在保留未编辑细节中的作用**。对于单条件参考模型，模型权重、其他输入和随机种子均固定。(a) 水平循环移位源词元值，同时固定位置分配和输出查询，使同一查询关联到不同的源内容。(b) 条件潜变量偏移下，相对于源图像的非编辑区域均方误差的增幅；误差条表示逐图像自助法95%区间。所有三种偏移都会降低保留效果。(c) 四种输出中的同一未编辑区域，屏幕截图边缘以淡紫色着色以便显示。第二行显示相对于基线的 RGB 均方根差异，在着色前计算，并以统一刻度绘制。曲线衡量源内容保留误差，差异图则显示干预引起的输出变化。

以及 $\mathcal{K}_m$ 掩码图像键集合。我们定义

$$\mu_i = \sum_{j \in \mathcal{K}_m} a_{ij}, \qquad \pi_{ij} = a_{ij}/\mu_i,$$
$$h_i^{(m)} = \mu_i \sum_{j \in \mathcal{K}_m} \pi_{ij} v_j. \tag{12}$$

其中 $\mu_i > 0$ 和 $\sum_j \pi_{ij} = 1$。源权重 μ_i 对查询 i 分配给掩码图像的注意力求和，用以表征该条件下的注意力强度。空间聚合核 π_{ij} 是该条件内经过归一化的空间注意力分布。相同的分解方式也适用于风格参考，且无需额外的网络分支。

对于 Ω_n 中的查询，令理想源特征场 $f(p)$ 满足 $\|f(p) - f(p')\| \le L\|p - p'\|$，且处于所分析的范围内。令条件值为 $v_j = f(p_j) + e_j$，其中 e_j 表示编码和表征残差。接下来

$$\left\| \sum_j \pi_{ij} v_j - f(p_i) \right\| \le L\, d_i + \bar{e}_i,$$
$$d_i = \sum_j \pi_{ij} \|p_j - p_i\|, \tag{13}$$
$$\bar{e}_i = \sum_j \pi_{ij} \|e_j\|.$$

该界限由三角不等式推出；其推导见附录A。它包含空间混合项 Ld_i 和条件表征残差 $\bar{e}_i$，分别对应聚合距离和特征表示质量。在给定假设下，该界限刻画了归一化条件特征聚合误差。完整输出还取决于源权重、其他条件、后续网络层以及解码器。

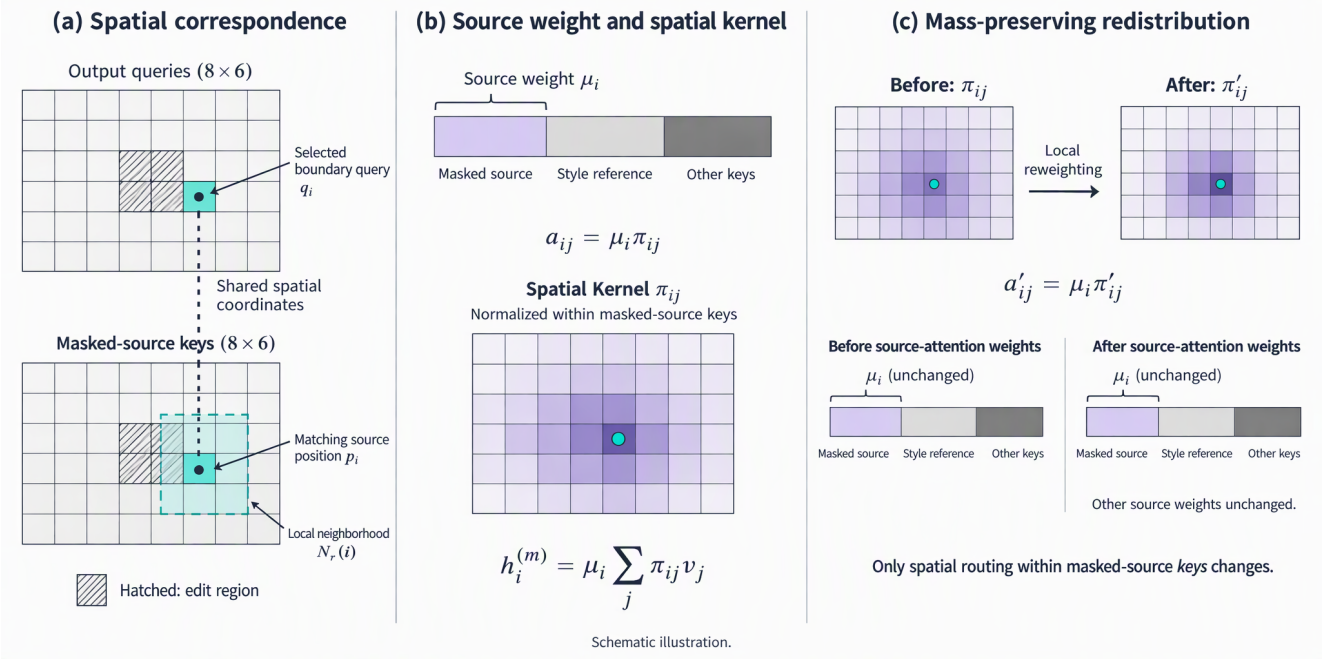

图 5. **注意力传输分解与空间重新分配**。(a) 掩码图像与目标图像共享空间参照，使每个输出查询向量都与同一网格位置处的条件邻域 $\mathcal{N}_r(i)$ 相关联。(b) 对掩码图像的注意力分解为其总权重 μ_i 和内部归一化的空间分布 π_{ij}。(c) 重分配会调整掩码图像内部的空间注意力分布，同时保留 μ_i 以及所有其他键的注意力权重。网格示意表示单个查询向量在掩码图像上的空间注意力分布。

选择性视觉传输。 为加强从对应邻域中的检索，同时保留源画布的整体贡献，我们应用保质量空间重分配。它修复 μ_i ，并将掩码图像中的注意力集中到空间上对应的邻域。

对于选定的非编辑查询向量，令 $\mathcal{N}_r(i)$ 表示对应位置处的局部条件键邻域，并设 $b_{ij} = \beta\,\mathbf{1}[j \in \mathcal{N}_r(i)]$, , 其中 $\beta \geq 0$。重分配为

$$\pi'_{ij} = \frac{\pi_{ij}\exp(b_{ij})}{\sum_{k\in\mathcal{K}_m}\pi_{ik}\exp(b_{ik})}, \qquad a'_{ij} = \mu_i \pi'_{ij}, \quad j \in \mathcal{K}_m. \tag{14}$$

因此， $\sum_{j\in\mathcal{K}_m} a'_{ij} = \mu_i$，所有其他键的注意力权重保持不变，而 $\beta = 0$ 则恢复原始注意力。局部指数加权与重新归一化使空间聚合核能够独立调整。

传输范围与职责划分。空间重分布会调整选定的非编辑查询向量如何读取掩码图像，从而支持复用源画布中的内容。在编辑区域内，模型根据 y 生成新的笔画与布局，并通过风格参考保留源文本外观。在同一主干网络中，这些功能协同完成文本替换并保留未编辑内容。

这带来了两个可检验的预期：破坏条件与输出之间的空间对应关系会影响图像保持效果，而抑制局部对应的键则会改变输出。局部注意力聚合可通过分配给掩码图像的注意力权重总和 μ_i 、均值聚合距离 d_i 以及局部注意力质量来表征

$$A_r = \frac{1}{|\mathcal{Q}_n|}\sum_{i\in\mathcal{Q}_n}\sum_{j\in\mathcal{N}_r(i)} a_{ij}, \tag{15}$$

其中 $\mathcal{Q}_n$ 是非编辑查询向量的集合。 A_r 取决于掩码图像注意力权重总和，以及分配给对应邻域的比例，并结合 μ_i 和 d_i进行联合解读。第6.2 节通过条件聚合特征、输入比较和空间干预来检验这些预期，并分别评估路由变化和图像保持效果的提升。

3.4. 多阶段训练

两个训练阶段都使用 Fill 联合学习字符生成和未编辑内容保留（见图 6）。阶段1仅将掩码图像用作视觉条件，并使用原始文本进行自监督重建。

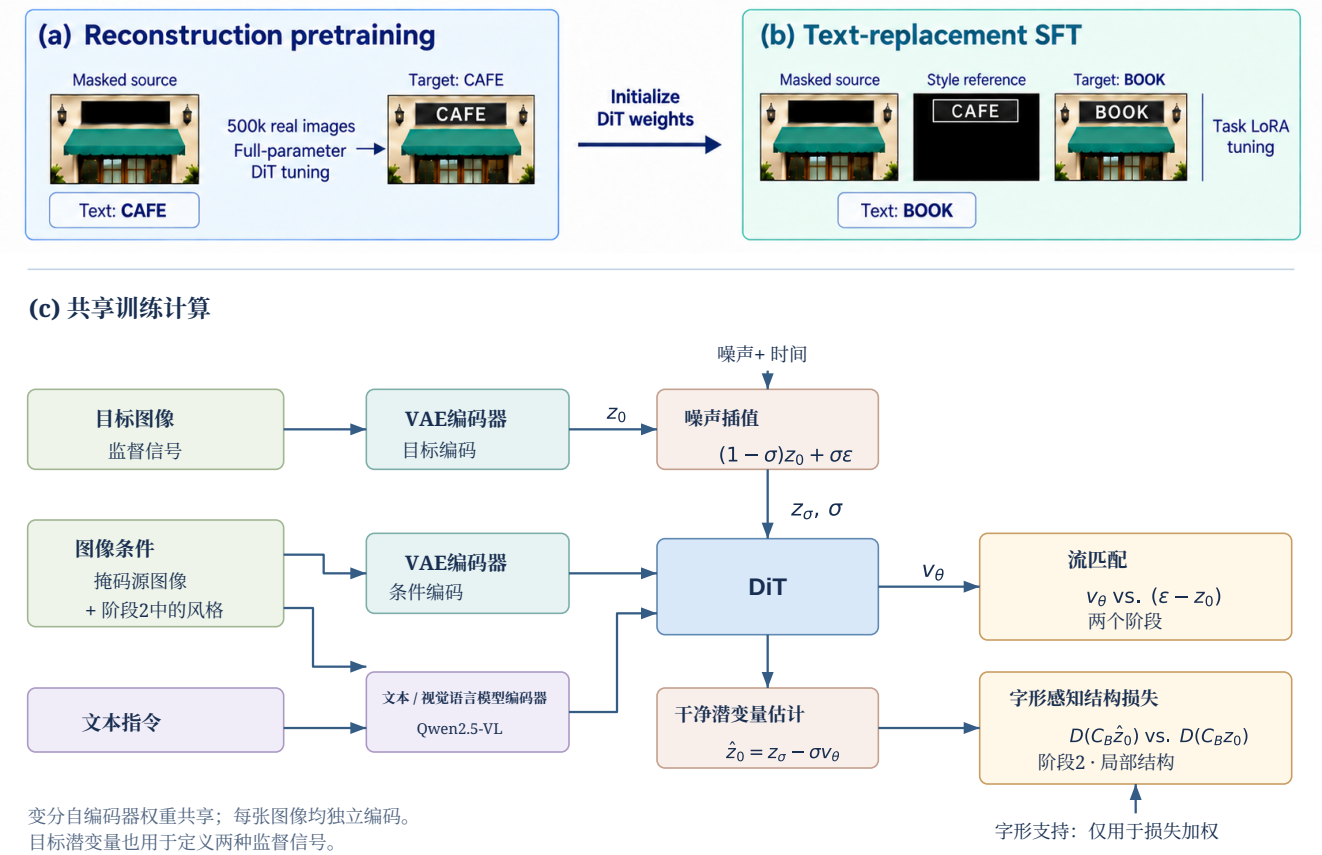

图6. **两阶段训练与共享计算**。(a) 阶段1通过在 500,000 张真实图像上进行掩码文本重建，对 DiT 进行全量微调。(b) 阶段2继承这些权重，加入源文本风格参考，并通过任务LoRA学习文本替换。(c) 目标图像经变分自编码器编码后，与噪声插值，形成 DiT 的训练输入。图像条件由共享的变分自编码器单独编码，并与文本指令一起由 Qwen2.5-VL 联合编码。两个阶段均使用流匹配对速度预测进行监督。阶段2还会恢复预测的干净潜变量，并将其在编辑区域内的局部结构与目标潜变量的局部结构进行比较。 z_0 表示目标图像潜变量； C_B 和 D 分别表示编辑区域裁剪和 Scharr 梯度。字形支持仅用于结构损失加权。

阶段2加入源文本风格参考，并通过双条件填充进行文本替换 SFT，以学习字符更新和外观保持。

阶段1：自监督文本重建。我们从 500,000 张真实图像中构建重建样例，将每张原始图像作为目标，并将其文本掩码版本作为唯一的视觉条件。通过光学字符识别获取的原始文本 y_{src} 包含在指令中。该任务是

$$\hat{I}^{(1)} = F_\theta(I_m, y_{\text{src}}), \qquad I_{\text{tgt}} = I_{\text{src}}. \tag{16}$$

该阶段更新所有 DiT 参数，以学习对掩码字符进行文本引导的恢复，并保留未编辑区域中的源内容。在没有源文本风格参考的情况下，模型根据文本和周围图像恢复缺失内容。所得权重将用于初始化后续的文本替换训练。

阶段2：文本替换监督微调。阶段2继承重建权重，并在公式（3）中加入源文本风格参考，将重建扩展至文本替换。包含不同源文本和目标字符串的训练样本要求模型在利用外观参考的同时更新字符内容。条件、目标图像和编辑区域会进行同步的空间变换，以在训练期间保留坐标对应关系。

阶段1从 Qwen-Image-Edit- 初始化2511 [18,27], , 阶段2通过 LoRA [8]进行文本替换监督微调。两个阶段均使用第 3.2 节（公式（5））中定义的全图流匹配目标函数。阶段1设置 $I_{\text{tgt}} = I_{\text{src}}$以学习文本恢复和未编辑内容保留。阶段2以编辑后图像为目标，并在全图流10匹配监督中加入字形感知结构损失（公式（

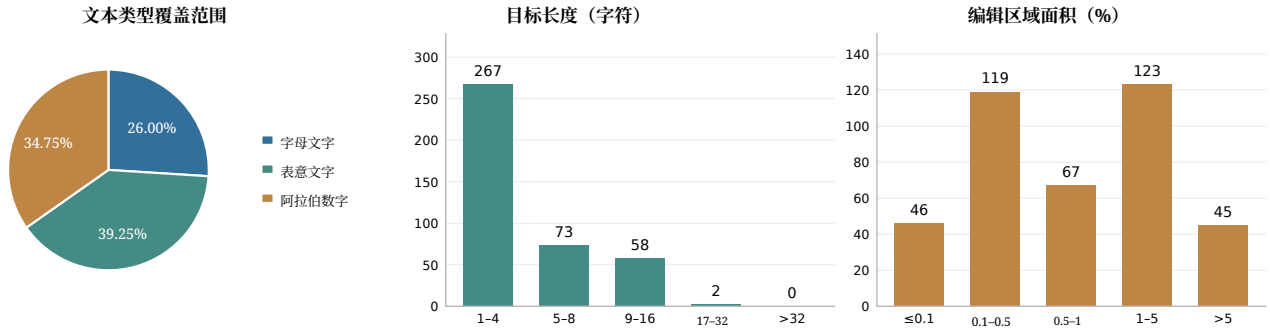

图7. **TBench 数据分布**。左：400 个任务中三种文本类别各自所占的比例；每个任务恰好属于一个类别。中：移除空格后的目标字符串字符数。右：编辑区域面积占源图像面积的百分比。

匹配监督：

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda_g \mathcal{L}_{\text{glyph}}. \tag{17}$$

我们在阶段2中设置 $\lambda_g = 0.05$ 。

两个损失函数均在阶段2的样本上取平均。

4. TBench：细粒度文本编辑基准

4.1. 设计目标与任务定义

评估细粒度文本编辑，需要综合考察目标文本准确率、源文本外观保留效果以及未编辑内容保留效果。文本识别指标无法充分反映字体、颜色和纹理的一致性，而全图相似度可能会掩盖小区域中的字符误差。因此，我们构建了 TBench，以统一的基准评估文本准确率、视觉保持效果和整体编辑质量。每项任务都提供一张源图像、一个编辑区域和一个目标字符串，并要求输出完整的编辑后图像。评估既考察指定区域内的文本更新，也考察周围文本和背景细节的保留效果。

4.2. 数据集构建与覆盖范围

TBench 涵盖屏幕界面、场景文本、图形与文本结合的图像、收据及其他场景。每项任务都包含一张源图像、一个目标字符串、编辑区域坐标以及标注的图像尺寸。评估时，会根据图像尺寸映射编辑区域，使文本识别和非编辑区域评估使用对应的空间区域。图7展示了文本类型、目标字符串长度和编辑区域面积的分布。

TBench涵盖字母文字、表意文字和阿拉伯数字，分别包含104、157和139项任务，占全部任务的26.00%、39.25%和34.75%。这些类别彼此互斥，每项任务仅归入其中一个类别。

在这些文本类型中，TBench还涵盖长度变化、密集布局和极小文本。长度变化测试原有空间内的字符排列和外观保留效果；密集布局测试目标文本更新和邻近文本保持；极小文本测试有限像素区域内的字符结构和细节保留。表意文字复杂的笔画和部件对局部结构准确性提出了更高要求。

4.3. 评估协议

我们使用 PP-OCRv4 评估文本准确率。将标注框映射到输出图像，并向外扩展 10 个像素。拼接局部识别出的文本并移除空格后，计算目标字符串包含率（OCR）和归一化莱文斯坦距离（ED）。ED 按识别文本与目标字符串中较长者的长度进行归一化。图像保持效果通过非编辑区域结构相似性[26] 和全图 AlexNet 学习感知图像块相似度 [29], 进行衡量，分别评估局部结构保留效果和整体感知差异。整体编辑质量由 Gemini-3.0-pro 使用视觉语言模型评分（LLM）进行评估，分数范围为 1 至 5，分数越高表示质量越好。人工评估采用三个等级：好、中、差。OCR 包含率与完整字符串精确匹配率有所不同。参考图像和图像处理细节见附录C。文本指标评估内容更新，图像保持指标评估源图像细节的保留情况。整体视觉评分和人工评估还会进一步考察字体、颜色、布局和整体编辑质量。综合而言，这些指标能够全面刻画细粒度编辑，而不仅仅局限于文本识别或全图相似度。

5. 实验结果

5.1. 实验设置

阶段1全参数微调。从 Qwen- Image-Edit-2511 开始，对 DiT 进行 5 个训练轮次的全参数微调

表 1. TBench 上的定量比较。三组列分别评估文本准确率、图像保持和整体质量。SSIM 在非编辑区域上计算，LPIPS 在完整图像上计算。4步和8步后缀表示采用相应采样步数的蒸馏变体。

	文本准确率		图像保持		整体质量
模型	OCR↑	ED↓	SSIM↑	学习感知图像块相似度↓	LLM↑
FireRed Image Edit	.7200	.2824	.8968	.0765	4.005
FireRed Image Edit（4步）	.6970	.3034	.9432	.0560	3.970
Qwen-Image-Edit-2511	.7121	.3076	.9392	.0598	3.939
Qwen-Image-Edit-2511（8步）	.6633	.3447	.9494	.0520	3.874
Qwen Image Edit Max	.6492	.3598	.9072	.0869	3.806
Qwen Image Edit Plus	.6173	.3692	.8802	.0796	3.653
Qwen Image 2	.5990	.4137	.8916	.0893	3.279
Qwen Image 2 Pro	**.7277**	.3330	.8581	.0777	3.162
Qwen AIO v1.8	.3950	.5499	.6434	.3951	3.145
Seedream 4	.4774	.4275	.7628	.1757	2.598
Seedream 4.5	.7020	.2838	.8552	.1723	3.909
Wan Image 2.6	.6354	.3561	.8886	.0763	3.667
Nano Banana	.6450	.3518	.8596	.0906	**4.490**
Nano Banana 2	.6889	.3202	.8595	.0832	4.450
Nano Banana Flash	.3249	.5842	.7158	.1847	2.893
EditText（我们的方法）	.7093	**.2352**	**.9769**	**.0460**	4.1756

表 2。TBench 上的人工评估通过率。4步和 8步后缀表示采用相应采样步数的蒸馏变体。

模型	通过率↑	模型	通过率↑
FireRed Image Edit	45.50%	Seedream 4	36.22%
FireRed Image Edit（4步）	39.47%	Seedream 4.5	57.07%
Qwen-Image-Edit-2511（8步）	44.22%	Seedream 5 Lite	44.33%
Qwen Image Edit Max	36.61%	Seedream 5 Pro	64.21%
Qwen Image Edit Plus	47.74%	Wan Image 2.6	43.72%
Qwen Image 2	34.36%	Nano Banana Pro	75.00%
Qwen Image 2 Pro	32.98%	Nano Banana	24.87%
Qwen Image 3 Pro	44.95%	Nano Banana 2	70.95%
Qwen AIO v1.8	21.88%	SenseNova U1.5	47.00%
GPT-image-2	57.07%	gpt-image-2.5-flare	54.04%
gpt-image-2.5-sunburst	68.79%	**EditText（我们的）**	**76.68%**

使用 500,000 个自监督文本重建样本。训练使用 4 个节点，全局批次大小为 64，峰值学习率为 10^{-5}，并采用预热和余弦衰减，最大训练区域为 1024^2。所得权重用于初始化阶段2。完整配置见附录 B.1。

第二阶段文本替换监督微调。 阶段2继承重建权重，并在 21,061 个文本替换示例上为 DiT 训练秩为 128 的任务 LoRA。每个示例提供一个源图像、一个编辑区域和一个目标字符串，以及一张编辑后图像，其源字符串和目标字符串各不相同。视觉条件和目标图像均经过一致的空间变换。目标图像潜变量提供流匹配监督，离线字形支持图则用于加权结构损失。训练采用恒定学习率 2×10^{-4}，训练 5 轮，每轮进行 2 次数据集重复，全局批次大小为 64，最大训练面积为 3,500,000 像素，结构损失权重为 $\lambda_g = 0.05$。数据集构建、训练设置和接口详见附录 B.2。

推理。 推理结合了任务LoRA和四步蒸馏 LoRA，两者的强度均为 1。我们使用 4 个采样步骤、引导尺度 1 和随机种子 42。输出和变分自编码器条件的目标区域均约为 2048^2, , 在保留源宽高比的同时，将尺寸四舍五入到网格。

在推理阶段，公式（14）会在去噪步骤0、第25层，对编辑框外侧四个词元宽的带状区域内的边界查询向量，在所有注意力头上重新分配掩码图像中的空间注意力。层和步骤均从0开始索引。根据编辑框在条件网格上的较长边，我们使用 $(r, \beta) = (2, 0。15)$、$(4、0。25)$、$(6、0。35)$；阈值见附录 B.4。对于每个查询向量和注意力头，重分配会保留掩码图像注意力权重的总和，并保持其他所有键权重不变。实现中，会在共享输出投影之前，将重分配后的加权特征与原始加权特征之差加到原生注意力输出上。此操作不会引入可训练参数或条件词元，并且 $\beta = 0$ 恢复原始路径。

评估指标。我们使用第 4.3节定义的光学字符识别、编辑距离、结构相似性指数、学习感知图像块相似度、大语言模型评分和人工评估。

5.2. 定量比较

在 TBench 上，EditText 的局部 OCR 包含率为 70.93%，编辑距离为 0.2352，非编辑区域结构相似性为 0.9769，全图学习感知图像块相似度为 0.0460，大语言模型评分为 4.1756。如表 1 所示，EditText 获得了最低的编辑距离、较高的非编辑区域结构相似性以及最低的全图学习感知图像块相似度，表明其在更新文本的同时，能够有效保留未编辑的视觉内容。其人工评估通过率达到 76.68%，优于对比方法（表 2）。

5.3. 定性结果

图8 展示了字母文字、表意文字和阿拉伯数字的细粒度文本编辑结果，每个类别包含两行。示例分别展示了：照片铭牌和密集接口中的字母更新、长度变化以及极小文本；复杂汉字中描边风格和手写风格的保留，以及密集布局中的局部更新；数字数量变化、较小数字以及多样笔画的变化。全图对比还显示，未编辑区域中的相邻文本、纹理和背景细节均得到了保留。

6. 消融与机制分析

我们首先通过组件消融以及对重建检查点的评估，分析各设计组件和训练的贡献。随后，我们通过条件对比和空间干预，考察注意力传输中的信息检索。

表 3. TBench 上的组件消融。完整方法包含全部四个部件。

配置	OCR↑	ED↓	结构相似性指数↑	学习感知图像块相似度↓	LLM↑
完整方法	.7000	.2059	.9786	.0452	4.125
不使用阶段1	.6600	.2376	.9789	.0461	3.995
不使用风格参考	.6700	.2500	.9627	.0364	3.560
不含字形损失	.6950	.2006	.9753	.0477	3.945
无重分配	.6800	.2441	.9231	.0501	3.765

表4. 在 TBench 上使用单个掩码图像输入和通用四步加速 LoRA 时，首个训练周期重建检查点的结果。

模型	OCR↑	ED↓	结构相似性指数↑	学习感知图像块相似度↓	LLM↑
阶段1	.6650	.2757	.9634	.0353	3.4600

6.1. 组件与训练分析

组件消融。 我们在 TBench 上每次移除一个部件；所有消融实验使用相同的样本和评估设置，并保持其余配置不变。完整方法包含自监督重建初始化、源文本风格参考、字形感知结构损失以及空间注意力重分配。

移除自监督重建初始化后，光学字符识别从 .7000 降至 .6600，编辑距离和大语言模型评分也变差，表明重建训练有利于后续文本替换。移除风格参考会使光学字符识别、编辑距离、结构相似性指数和大语言模型评分变差，但学习感知图像块相似度有所改善。因此，源文本参考会影响编辑质量的多个方面，而单一的感知距离无法完全捕捉这些影响。移除结构损失会使光学字符识别、结构相似性指数、学习感知图像块相似度和大语言模型评分变差，同时编辑距离略有改善，说明它能在多个指标上带来益处，但并非对所有文本指标都有帮助。移除空间重新分配后，结构相似性指数从 .9786 降至 .9231，文本指标、学习感知图像块相似度和大语言模型评分也变差，这支持了路由调整对编辑质量的贡献。

使用重建权重进行文本替换。 我们评估第一个训练轮次结束时的重建检查点在文本替换任务上的表现（表 4）。训练会恢复原始文本，而评估则提供新的目标字符串，以评估从重建到文本替换的迁移能力。

首个训练周期重建检查点的局部 OCR 包含率为 66.50%，非编辑区域结构相似性指数为 0.9634，全图学习感知图像块相似度为 0.0353。在进行文本替换监督微调之前，模型已经能够生成新目标文本中的字符，同时保留部分源图像内容。阶段2进一步引入源文本外观参考，并学习目标内容替换。

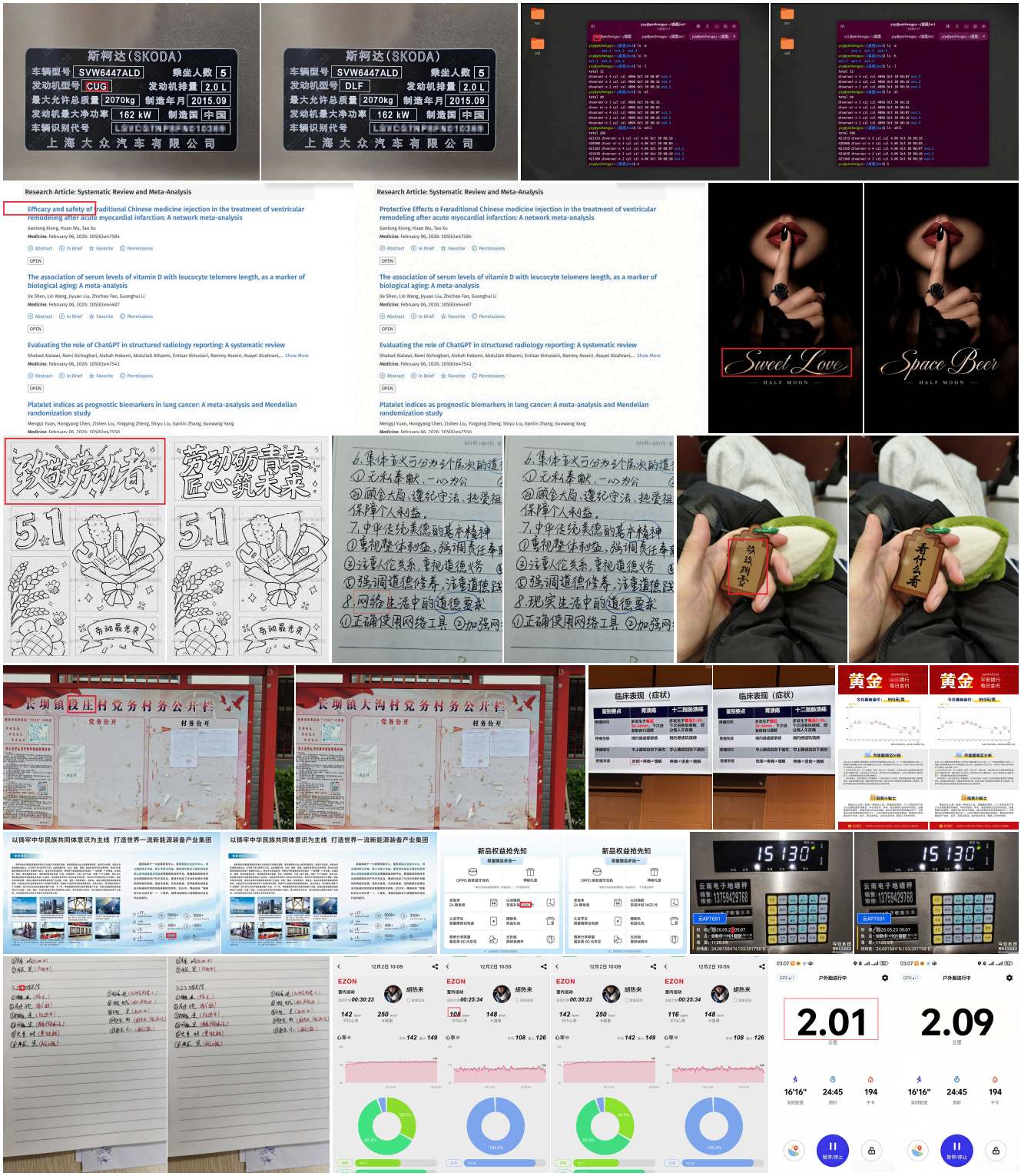

图 8。三**种文本类别的细粒度编辑结果**。TBench 评估结果，从上到下依次为字母文字、表意文字和阿拉伯数字，每种类别各占两行。每对图像中，左侧为源图像，右侧为对应的编辑结果；红色方框标示源图像中的编辑区域。完整图像在保持原始宽高比的同时紧密排列。

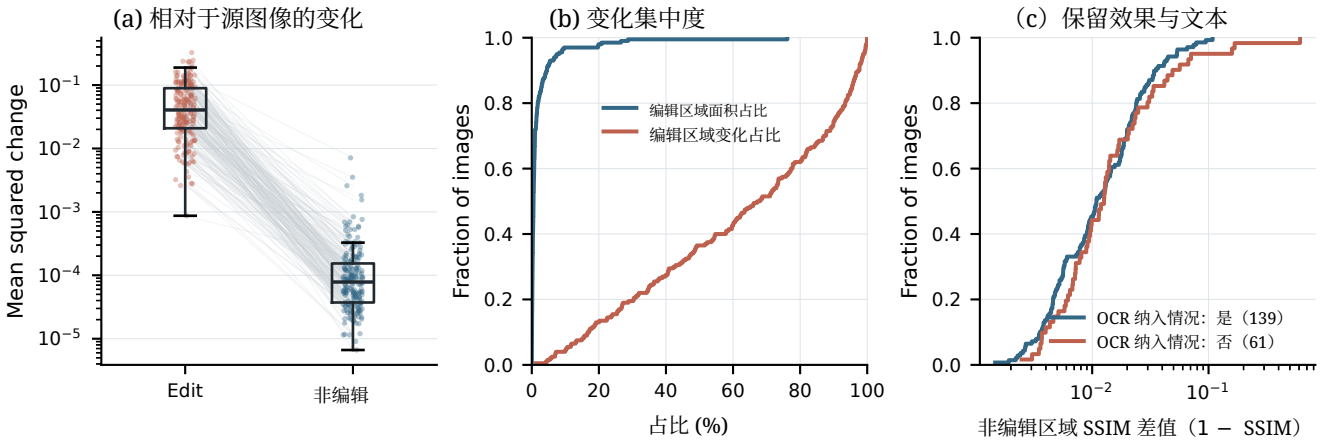

图 9。**双条件填充中的变化分布与文本正确性**。（a）逐图比较编辑区域和非编辑区域中相对于源图像的均方变化；相连的点对应同一图像。箱线图显示中位数、四分位数和 1.5 倍四分位距。（b）编辑区域面积占比及其在整幅图像平方变化中的占比。（c）按 OCR 纳入情况分组的非编辑区域 1 − 结构相似性指数累积分布。横轴采用对数刻度；值越小表示保留效果越好。平方变化衡量相对于源图像的修改幅度。

参考模型比较。我们在 TBench 上比较了两个使用相同主干网络的参考模型。第一个模型在原始骨干网络上训练 Fill LoRA，第二个模型在经过重建微调的骨干网络上训练单图编辑 LoRA。两者均使用一张以红框标记的图像进行评估。它们的非编辑区域 SSIM 分数分别为 0.9750 和 0.9018，全图学习感知图像块相似度分数分别为 0.0468 和 0.1286，OCR 纳入率分别为 0.6533 和 0.7588，LLM 平均分数分别为 3.9698 和 4.1658。第一种配置对图像的保留效果更好，而第二种配置具有更高的文本准确率和总体分数。这些模型在训练数据和分辨率方面也存在差异，该比较展示了不同配置之间的质量权衡。配对区间和配置差异见附录 D。

6.2. 注意力传输机制分析

章节 3.3 通过在真实图像上进行空间错位对比，引入了源检索。在此，我们提供完整的实验设置、定量统计和局部路由干预，以分析各条件的作用，以及空间对应关系对信息传输的影响。为研究未编辑内容保留、目标文本更新和源文本外观保留，我们采用一个通过文本替换 SFT 直接在原始骨干网络上训练的双条件 Fill 参考模型。将掩码图像和风格参考作为视觉输入后，我们更改目标文本、清空风格参考，并分析聚合特征，以考察模型如何利用每种输入。该参考模型既未采用阶段1重建初始化，也未使用结构损失。

为进一步检验空间对应关系，我们进行了局部键抑制和空间错位实验，使用的是一个独立训练的单条件模型。键抑制会降低选定条件位置的 Softmax 前注意力分数，从而削弱从这些位置进行的特征检索。单一视觉条件排除了第二张条件图像的影响，使局部检索得以单独考察。每组比较仅改变所研究的因素，同时固定模型权重、其他输入条件和随机种子。详细设置见附录B 和E。

变化分布与文本正确性。 图 9(a,b) 总结了生成图像与源图像之间逐像素平方变化的情况。对每幅图像的比例取平均后，编辑区域仅占图像面积的 2.12%，却贡献了整幅图像平方变化的 62.12%，表明修改集中在替换区域。这些变化包括移除原始文本并生成新字符。按局部光学字符识别纳入情况对输出分组后，通过和未通过的组，其非编辑区域结构相似性指数分别为 0.9829 和 0.9688，整幅图像学习感知图像块相似度分别为 0.0414 和 0.0538。未通过光学字符识别纳入情况的输出仍能很好地保留未编辑区域（图 9(c)），这支持将文本正确性与视觉保留效果分开评估。统计定义和坐标处理见附录 G。

Fill 中的条件检索与外观保持。为分析从两种视觉条件中进行的检索，我们记录掩码图像和风格参考的总注意力权重及实际加权特征聚合情况。图 10(b,c) 中，掩码图像的响应在编辑区域周围更强，而风格参考的响应则

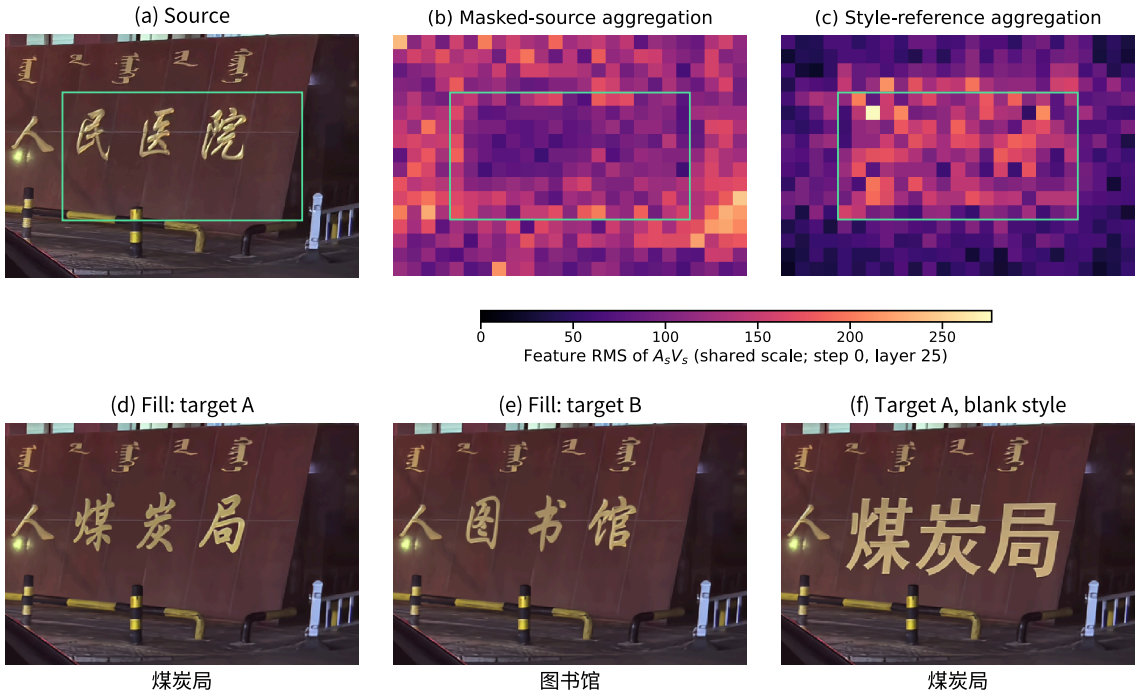

图10. **Fill 中的条件聚合与源文本外观复用**。(a) 源图像裁剪图；绿色方框标出标记网格上的编辑区域。(b,c) 去噪步骤0、第25层中，来自掩码图像和风格参考的实际聚合特征 A_sV_s 的均方根。两张图使用相同色阶；颜色越亮，特征幅值越大。特征在注意力输出投影和门控之前测量。(d,e) 固定源图像、视觉条件和随机种子，使用不同目标字符串生成的输出。(f) 与(d)相同的目标文本，但将风格参考替换为尺寸相同的黑色图像。生成内容会随目标文本变化。在此示例中，将风格参考置空后，文本变得更规整且尺寸更大。聚合图显示加权后的视觉条件特征响应；输出比较用于评估这些响应对文本外观的影响。

主要集中在编辑区域内。在第 0 步、第 25 层，分配给风格参考的平均总注意力权重在编辑区域为 10.37%–13.31%，在外边界为 5.91%–6.79%，表明注意力分配存在稳定的区域差异。各样本的数值和注意力收集设置见附录 B.4。掩码图像中的空间检索分布及相应的邻域统计见附录 F。

目标内容更新与源文本外观保留。图11 通过改变目标字符串并将参考图置空，比较目标文本和风格参考所产生的效果。在视觉条件固定而目标字符串改变时，输出内容会更新，同时保留源文本的主要外观。在目标文本固定而风格参考置空时，手绘标题会从空心变为实心字符，海报数字会从狭长变为宽圆，标牌文字会变得更加规整且尺寸更大，而印花布料文字则会从蓝色变为深灰绿色。这些对比展示了目标文本如何控制生成内容，以及风格参考如何影响新字符的外观。参考图的效果因示例而异，图像上下文也能提供风格线索。将参考图置空会改变变分自编码器和 VLM 的编码，因此这一对比评估的是风格参考作为完整输入时的效果。表3 评估了各个组件对整体质量的贡献。

局部条件路由的功能作用。为检验模型在生成未编辑区域时是否会使用空间对应的源信息，我们在单条件设置下比较了两种干预。一种抑制每个输出位置对应邻域中条件键的注意力分数；另一种以相同的抑制强度抑制数量相同的随机键。相较于未经修改的输出，它们在未编辑区域的平均扰动均方误差分别为 1.815×10^{-5} 和 1.028×10^{-5}。均方误差越大，表明削弱从这些键检索信息所造成的输出变化越大。



图11. **目标内容更新与源文本外观保留效果**。各列依次展示源图像、使用风格参考的目标 A、使用空白风格参考的目标 A，以及使用风格参考的目标 B。每行固定模型、源图像、编辑框和随机种子。裁剪区域采用相同的空间范围，保留各自的宽高比，并包含周围的图像上下文。第二列和第三列仅改变风格参考；第二列和第四列仅改变目标文本。目标 A/B 分别为：(a) 劳动砺青春匠心筑未来 / 匠心绘梦想奋 斗创未来；(b) 699 / 899；(c) 煤炭局 / 图书馆；(d) 交通科技 / 交通工程。

对应键抑制在 90% 的样本中造成了更大的变化（图 12（a））。因此，削弱从对应邻域检索信息通常比抑制随机位置对输出的影响更大，这支持在非编辑区域生成中使用空间对应的源特征。两组的被抑制注意力权重总量并未进行匹配。设置和局部输出比较见附录 E。

空间对应关系与层–步依赖性。将条件潜变量分别水平循环平移 1、2 和 4 个词元，相对于源图像的非编辑区域均方误差均值分别增加 0.01533、0.01895 和 0.02304。这三种设置都会降低保留效果（图12(b)）。图4展示了相应的局部变化。平移保留了源词元值的集合，同时改变了空间对应关系，这支持了空间对应关系在图像保持中的作用。循环平移也会引入边界环绕；其影响及

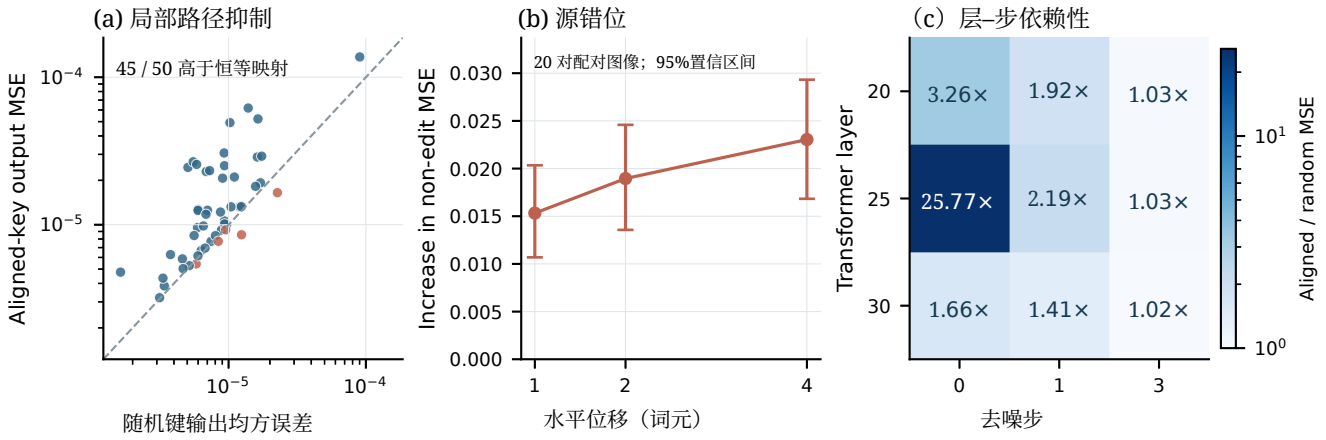

图12。**单一视觉条件下的注意力传输干预**。（a）在同一图像上进行对应键抑制和随机键抑制；均方误差相对于未修改的输出进行测量。（b）在条件潜变量移位下，相对于源图像的非编辑区域均方误差的增加，并附逐图像自助法95%区间。（c）九组实验中的层和去噪步比较。单元格数值表示对应键抑制的平均扰动均方误差与随机键抑制的平均扰动均方误差之比。

解释范围将在附录E中讨论。

图 12(c) 比较了九组实验中第 20、25 和 30 层在去噪步 0、1 和 3 时的情况。相较于随机键抑制，对应键抑制的扰动效应因层和去噪步而异，并在最后一个去噪步显著减弱，这表明视觉传输存在层依赖性和时间依赖性。键抑制会改变检索位置，而空间错位会改变条件内容的位置，从两个角度检验对应关系的功能作用。

保质量空间重分配。在双条件 Fill 中，我们根据公式（14）对去噪步 0、第 25 层的掩码图像注意力进行重分配，且仅作用于编辑区域之外的边界查询向量。零强度对照与原始输出在像素级完全一致。强度非零时，掩码图像注意力权重总量的最大绝对偏差为 2.38×10^{-7}，而所有其他键权重保持不变。非编辑区域和边界均方误差的变化因样本而异，未呈现一致的保留效果提升，整体视觉变化也较小（表 9）。这验证了在保持总注意力强度不变的同时，可以独立调整掩码图像内部的空间注意力分布。完整方法中重分配的效果通过表 3中的组件消融进行评估。

机制分析与方法设计。 这些结果支持将文本编辑分解为可复用信息的传输和目标结构的生成。空间干预用于检验对应关系，条件对比用于检验内容更新和外观保留，而空间重新分配则提供了一种独立调节条件内注意力分布的方法。

这些实验使用单条件模型和直接通过文本替换监督微调训练的双条件参考模型，对选定样本进行局部检索分析。通过组件消融实验评估完整方法的编辑质量，考察重建初始化、风格参考、结构监督和空间路由（表3）。

7. 结论

我们提出了 EditText，这是一种细粒度文本编辑框架，将目标字符生成与源图像中的视觉传输相结合。注意力传输通过源权重、空间聚合和特征表示质量来分析源细节检索，并通过选择性视觉传输增强未编辑区域的信息复用。多阶段训练和字形感知结构损失支持字符更新和源文本外观保留。我们引入了 TBench，用于系统评估字母文字、表意文字和阿拉伯数字中的文本更新、外观保留效果及未编辑细节保留效果。实验和定性结果表明，该模型能够处理长度变化、密集布局、极小文本以及复杂的汉字结构。在 TBench 上，该模型在未编辑区域一致性和全图感知相似度方面达到了最先进水平。

局限性与未来工作。 细节复用仍受源图像编码质量的限制。风格参考中的原始字符结构也可能干扰目标字符生成，导致细微的笔画错误或残留原始文本。长度变化要求在给定的编辑区域内重新排列字符，而布局稳定性和外观保持效果仍有提升空间。

未来工作将探索细节编码、将外观与字符结构解耦，以及长度自适应布局，以提升编辑准确性和稳定性。

参考文献

[1] Tim Brooks、Aleksander Holynski 和 Alexei A. Efros。《InstructPix2Pix：学习遵循图像编辑指令》。发表于 CVPR，2023 年。4[2] Jingye Chen 、 Yupan Huang 、 Tengchao Lv、Lei Cui、Qifeng Chen 和 Furu Wei。《 TextDiffuser：将扩散模型用作文本绘制器》。发表于 NeurIPS，2023 年。2、4[3] Guillaume Couairon、Jakob Verbeek、Holger Schwenk 和 Matthieu Cord。《DiffEdit：基于扩散的语义图像编辑与掩码引导》。发表于 ICLR，2023 年。4[4] Patrick Esser、Sumith Kulal、Andreas Blattmann、Rahim Entezari、Jonas Müller、Harry Saini、Yam Levi、Dominik Lorenz、Axel Sauer、Frederic Boesel、Dustin Podell、Tim Dockhorn、Zion English 和 Robin Rombach。《扩展整流流 Transformer，实现高分辨率图像合成》。发表于 ICML，第 12606–12633 页，2024 年。2、4[5] Yu Gao、Lixue Gong、Qiushan Guo、Xiaoxia Hou、Zhichao Lai、Fanshi Li、Liang Li、Xiaochen Lian、Chao Liao、Liyang Liu 等。《Seedream 3.0 技术报告》。arXiv 预印本 arXiv:2504.11346，2025 年。4[6] Amir Hertz、Ron Mokady、Jay Tenenbaum、Kfir Aberman、Yael Pritch 和 Daniel Cohen-Or。《使用交叉注意力控制的 Prompt-to-Prompt 图像编辑》。发表于 ICLR，2023 年。4[7] Jonathan Ho、Ajay N. Jain 和 Pieter Abbeel。《去噪扩散概率模型》。发表于 NeurIPS，2020 年。4[8] Edward J. Hu、Yelong Shen、Phillip Wallis、Zeyuan Allen-Zhu、Yuanzhi Li、Shean Wang、Lu Wang 和 Weizhu Chen。《LoRA：大语言模型的低秩适配》。发表于 ICLR，2022 年。9[9] Phillip Isola、Jun-Yan Zhu、Tinghui Zhou 和 Alexei A. Efros。《使用条件生成对抗网络进行图像到图像转换》。发表于 CVPR，2017 年。4[10] Praveen Krishnan、Rama Kovvuri、Guan Pang、Boris Vassilev 和 Tal Hassner。《 TextStyleBrush：从单个示例迁移文本美学风格》。IEEE模式分析与机器智能汇刊，45(7)：9122–9134，2023 年。4[11] Hongxi Li、Tong Wang、Chengjing Wu、Tianbao Liu、Jiangtao Yao、Xiaochao Qu、Xinxiao Wu、Luoqi Liu 和 Ting Liu。《用于开放词汇场景文本编辑的自提示扩散变换器：通过上下文学习实现》。arXiv 预印本 arXiv:2605.15523，2026 年。2、4[12] Yaron Lipman、Ricky T. Q. Chen、Heli Ben-Hamu、Maximilian Nickel 和 Matt Le。《用于生成建模的流匹配》。发表于 ICLR，2023 年。5[13] Andreas Lugmayr、Martin Danelljan、Andrés Romero、Fisher Yu、Radu Timofte 和 Luc Van Gool。《RePaint：使用去噪扩散概率模型进行图像修复》。发表于 CVPR，2022 年。4[14] Chenlin Meng、Yutong He、Yang Song、Jiaming Song、Jia- jun Wu、Jun-Yan Zhu 和 Stefano Ermon。SDEdit：使用随机微分方程进行引导式图像合成与编辑。载于 ICLR，2022 年。4[15] Chong Mou、Xintao Wang、Liangbin Xie、Yanze Wu、Jian Zhang、Zhongang Qi 和 Ying Shan。T2I-Adapter：学习适配器，挖掘文本到图像扩散模型更强的可控能力。载于 AAAI，第 4296–4304 页，2024 年。4[16] William Peebles 和 Saining Xie。使用 Transformer 的可扩展扩散模型。载于 ICCV，2023 年。2、4[17] Qwen 团队。Qwen-Image-Edit：以更高质量和效率进行图像编辑。https://qwenlm.github.io/ blog/qwen-image-edit/，2025 年。4[18] Qwen 团队。Qwen-Image-Edit-2511 模型卡片。https: // huggingface.co/Qwen/Qwen- Image- Edit- 2511，2025 年。9[19]Robin Rombach、Andreas Blattmann、Dominik Lorenz、Patrick Esser 和 Björn Ommer。使用潜在扩散模型进行高分辨率图像合成。载于 CVPR，第 10684– 10695 页，2022 年。2、4[20] Chitwan Saharia、William Chan、Huiwen Chang、Chris A. Lee、Jonathan Ho、Tim Salimans、David J. Fleet 和 Mo- hammad Norouzi。Palette：图像到图像扩散模型。载于 ACM SIGGRAPH Conference Proceedings，第 1 10 页，2022 4 – 。[21] Yang Song、Jascha Sohl-Dickstein、Diederik P. Kingma、Abhishek Kumar、Stefano Ermon 和 Ben Poole。通过随机微分方程进行基于分数的生成建模。载于 ICLR，2021 年。4[22] Jianlin Su、Murtadha Ahmed、Yu Lu、Shengfeng Pan、Wen Bo 和 Yunfeng Liu。RoFormer：采用旋转位置嵌入的增强型 Transformer。Neurocomputing，568 127063，2024 6： 。[23] Narek Tumanyan、Michal Geyer、Shai Bagon 和 Tali Dekel。用于文本驱动图像到图像转换的 Plug-and-Play Diffusion Features。载于 CVPR，2023 年。4[24] Yuxiang Tuo、Yifeng Geng 和 Liefeng Bo。AnyText2：具有可定制属性的视觉文本生成与编辑。arXiv 预印本 arXiv:2411.15245，2024 年。2、4[25] Yuxiang Tuo、Wangmeng Xiang、Jun-Yan He、Yifeng Geng 和 Xuansong Xie。AnyText：多语言视觉文本生成与编辑。载于 ICLR，2024 年。2、4[26] Zhou Wang、Alan C. Bovik、Hamid R. Sheikh 和 Eero P. Simoncelli。图像质量评估：从错误可见性到结构相似性。IEEE Transactions on Image Processing，13(4):600–612，2004 年。10[27] Chenfei Wu、Jiahao Li、Jingren Zhou、Junyang Lin、Kaiyuan Gao、Kun Yan、Sheng ming Yin、Shuai Bai、Xiao Xu、Yilei Chen 等。Qwen-Image 技术报告。arXiv 预印本 arXiv:2508.02324，2025 年。4、9[28] Lvmin Zhang、Anyi Rao 和 Maneesh Agrawala。向文本到图像扩散模型添加条件控制。载于 ICCV，第 3836–3847 页，2023 年。4[29] Richard Zhang、Phillip Isola、Alexei A. Efros、Eli Shechtman 和 Oliver Wang。深层特征作为感知度量具有出乎意料的有效性。载于 CVPR，第 586 595 页，2018 10 – 。

A. 机制推导与假设

A.1. 局部特征聚合误差

公式（13）假设在所分析的位置存在可复用源特征 f（p_i），存在非负归一化权重π_{ij}，并且特征场在聚合范围内满足 Lipschitz 条件。由于 $v_j = f\ (p_j)\ + e_j$,

$$\begin{aligned}
&\left\| \sum_j \pi_{ij} v_j - f(p_i) \right\| \\
&= \left\| \sum_j \pi_{ij} [f(p_j) - f(p_i) + e_j] \right\| \\
&\le \sum_j \pi_{ij} \|f(p_j) - f(p_i)\| + \sum_j \pi_{ij} \|e_j\| \\
&\le L \sum_j \pi_{ij} \|p_j - p_i\| + \bar{e}_i.
\end{aligned} \tag{18}$$

锐利纹理和跨越语义边界的区域可能具有较大的 L，从而产生较松的界。残差e_j 包括表示误差和编码误差。该界将聚合位置与潜在特征空间中的条件表示分离开来。完整的生成过程还包括输出投影、残差连接以及后续解码。

多轴 RoPE 可以写成块旋转。设位置由非空间恒等映射 τ 和空间坐标 $(u,\ v)$ 组成：

$$R(p) = \mathrm{diag}\left(R_\tau(\tau), R_u(u), R_v(v)\right). \tag{19}$$

空间位置匹配时，后两个相对位移为零，而不同图像可能仍保留非零的恒等映射位移。注意力共同取决于内容相关的 $q,\ k$ 以及相对位置。空间变换、网格和位置分配共同决定了模型能够学习的对应关系。

B. 实现与可复现性细节

B.1. 阶段1全参数微调

阶段1使用 500,000 个真实图像重建样本，以每张原始图像作为目标，将其文本区域填充为黑色作为唯一视觉条件，并根据原始 OCR 文本构建指令。模型基于 Qwen-Image-Edit-2511 初始化，通过流匹配监督更新约 20.4B 个 DiT 参数，不使用风格参考或字形感知结构损失。

表5 给出了完整的重建训练配置，以及后续训练所用的检查点。第4 节6.1 另行评估了重建

表 5. 阶段1自监督重建的训练配置。

Item	设置
训练样本	500,000
更新的模块	所有 DiT 参数，≈20.4B
训练轮数 / 每轮重复次数	5 / 1
训练硬件	4 个节点
全局批次大小	64
最大训练面积	1024^2 像素
峰值学习率	10^{-5}
调度器	WarmupCosineLR
预热	500 个优化步数
精度 / 分布式策略	bf16 / DeepSpeed ZeRO-3
梯度裁剪 / 检查点保存	1.0 / 已启用
阶段2初始化	第5训练轮次结束

训练轮次 1 结束时的检查点，用于考察从重建到文本替换的早期迁移。此评估采用单一掩码图像条件、自适应图像尺寸，以及通用的 Lightning 四步加速 LoRA，不使用任务 LoRA。引导尺度为 1，随机种子为 42。本实验中使用的所有 TBench 任务均已完成生成和评分。光学字符识别使用 PaddleOCR 2.10.0 和 PP-OCRv4，并采用 10 像素的裁剪扩展；学习感知图像块相似度使用 AlexNet，大语言模型评分使用 Gemini-3.0-pro。这些结果对应于使用四步加速适配器进行的推理。

B.2. 阶段2数据、训练与推理设置

阶段2以（$I_m,\ I_s,\ y$）为输入，并使用编辑后图像I_{tgt}作为监督目标。目标字形支持图与目标图像进行相同的空间变换，且仅用于损失加权。推理仅需源图像、编辑区域和目标字符串。

训练数据。训练集包含 21,061 条编辑记录。所有样本均使用掩码图像、风格参考和目标文本输入，且条件与目标在空间上对齐。

优化和分辨率。训练继承完全微调的阶段1 DiT；设置列于表6中。LoRA 覆盖图像分支和文本分支中的注意力查询、键、值和输出投影，以及多层感知机输出投影和调制投影。相应的模块为 to_q、to_k、to_v、add_q_proj、add_k_proj、add_v_proj、to_out.0、to_add_out、img_mlp.net.2、img_mod.1、txt_mlp.net.2 和 txt_mod.1。目标图像和两个视觉条件均采用动态宽高比预处理，最大面积为 3,500,000 像素。条件图像的固定尺寸自动调整大小功能已禁用（edit_image_auto_resize_False）。

条件图像的固定尺寸自动调整大小功能已禁用（edit_image_auto_resize=False）。视觉条件使用零时间步（zero_cond_t=True）。

表6。阶段2文本替换监督微调的训练配置。

Item	设置
初始化	第一阶段微调的 DiT
训练记录	21,061
LoRA秩	128
优化器	bitsandbytes AdamW8bit
学习率 / 调度器	2×10^{-4} / ConstantLR，无预热
训练轮数 / 每轮重复次数	5 / 2
训练硬件	4 个节点
全局批次大小	64
最大训练面积	3,500,000 像素，保留宽高比
精度 / 检查点保存	bf16 / 已启用
结构损失权重	$\lambda_g = 0.05$

表7. 阶段2的训练与推理接口。

Item	设置
初始化	第一阶段微调的 DiT
更新	DiT 任务 LoRA
视觉条件	掩码图像 I_m、源文本风格参考 I_s
文本条件	目标字符串 y
全图监督	流匹配目标速度 $v^* = \epsilon - z_0$
结构监督	预测的干净潜变量的局部结构潜变量 $\hat{z}_0$ 和目标潜变量 z_0
结构支持图	仅训练阶段的损失加权
推理适配器	任务与四步蒸馏两个 LoRA，强度均为 1
输出 / VAE条件区域	两者均为 $\approx 2048^2$，保持宽高比
步数 / 引导 / 随机种子	4 / 1 / 42
选择性视觉传输	步骤 0，第 25 层；四词元外边界；保留总计掩码图像注意力权重

B.3. 字形感知结构损失实现

本节详细介绍第 3.2 节中的结构监督。全图速度监督和局部干净潜变量结构监督分别对应公式（5）和（10）。

支持图和区域对齐方式。 离线目标字形栅格图经过与目标图像相同的缩放和裁剪处理，并将编辑框向外取整至 DiT标记网格。每个 DiT标记对应于 2×2 解包潜变量位置，损失在相应的潜变量区域内计算。字形区域通过面积插值下采样至该网格，再除以其最大值进行归一化，并将低于 0.08 的权重设为零以进行阈值化。如果归一化前的最大值不超过 10^{-6}，则使用全为 1 的区域支持图。支持图和目标特征均采用停止梯度处理，而预测潜变量中的结构误差则反向传播至可训练的DiT参数。

梯度与幅值归一化。对裁剪后的潜变量逐通道应用 Scharr 算子，并使用复制填充。两个卷积核为

$$K_x = \frac{1}{32}\begin{bmatrix} 3 & 0 & -3 \\ 10 & 0 & -10 \\ 3 & 0 & -3 \end{bmatrix}, \qquad K_y = K_x^\top. \tag{20}$$

公式（8）使用目标区域梯度的均方根对两个方向的残差进行归一化，以避免直接将绝对特征幅值视为结构差异。计算使用 float32，并采用 $s_{\min} = 0.02$、$\delta = 0.1$、$\varepsilon_S = 10^{-6}$ 和 $\varepsilon_R = 10^{-8}$。这些约束作用于潜空间中的局部梯度结构，并不直接定义 RGB 笔画方向误差。

辅助损失预算。在公式（10）中，ℓ_{FM} 已包含调度器的噪声权重。自适应系数按样本计算，并从梯度中分离。我们设定$\lambda_g = 0.05$，将辅助损失值限制在样本基础损失的 5% 以内；噪声门控还可进一步降低这一比例。结构支持权重仅影响辅助项。基础流匹配仍在完整潜变量上进行均匀平均。

B.4. 空间重新分配与机制分析设置

EditText 在第 0 步、第 25 层，对编辑框外侧四个词元宽的带状区域内的查询向量应用空间重新分配。对于较长边跨度不超过 4 个、超过 4 个但不超过 10 个，或超过 10 个词元的编辑框，我们分别采用$(r,\ \beta) = (2,\ 0.15)$，$(4,\ 0.25)$，$(6,\ 0.35)$。下表列出了第 6.2 节机制分析所用的双条件填充参考配置。该配置直接在原始骨干网络上执行文本替换监督微调，不使用第一阶段重建初始化或字形感知结构损失。其空间重新分配规则与完整方法一致。

条件构建。 掩码图像用黑色填充编辑框。风格参考则保留框内原位置的源内容，尺寸相同

表 8. 用于机制分析的 Fill 参考配置的训练和推理设置。

Item	设置
主干网络	Qwen-Image-Edit-2511
微调	DiT LoRA，秩 128
训练记录	21,061
最大训练面积	3,500,000 像素
学习率 / 调度器	2×10^{-4} / ConstantLR
训练轮数 / 每轮重复次数	5 / 2
训练硬件	4 个节点
全局批次大小	64
检查点	轮次结束 4
输出 / 条件目标区域	2048^2 / 2048^2
采样步数 / 引导	4 / 1
LoRA 强度 / 随机种子	两者均为 1 / 42

的黑色画布。条件图像保留动态宽高比，训练和推理区域由各自的配置设定。为将风格参考置空，我们使用尺寸相同的黑色图像，在保留图像数量和词元布局的同时，更改变分自编码器和 VLM 的编码。黑色参考可能偏离训练输入分布。图11 展示了在相同配置下对八张源图像进行的另一组配对实验。实验比较了目标文本固定时有无风格参考的输出，以及视觉条件固定时针对两个目标字符串的输出；其他输入、模型权重和随机种子均保持不变。这八张源图像构成定性配对集。注意力和路由统计采用下文所述的样本。

注意力收集。 注意力测量和空间重分配实验使用 TBench 样本，并为每个示例固定模型权重、源图像、编辑框和随机种子。对于原始 Fill模型，我们记录去噪步 0 和 3 与 Transformer层 6、25 和 54 的组合。目标替换、参考图置空和非零重分配均记录在第 25 层的第 0 和 3 个去噪步。层和去噪步均从零开始编号。每种组合均在整幅图像上采样固定的 24×24 查询网格，以及编辑区域和四词元宽外边界中的所有查询。图10 使用第 0 个去噪步、第 25 层的连续局部网格。图 14 重新使用保存在该位置、用于这些样本的原始查询和键。

我们使用公式（12）计算两种条件下的源权重和各注意力头的加权聚合特征。图 10 展示了拼接各注意力头特征后的通道均方根，两种条件共用同一色阶。测量在注意力输出投影和门控之前进行；均方根表示加权聚合特征的幅度。源权重先在注意力头之间取均值，再在区域查询向量之间取均值。在表 9中， $\bar{\mu}_e^{(s)}$ 和 $\bar{\mu}_b^{(s)}$ 分别表示编辑区域和外边界分配给风格参考的平均总注意力权重。

重分配实现与统计。 干预使用本节指定的层、去噪步和（r， β）设置，不针对测量样本重新调参。该实现将加权特征差异 $\Delta h_i = \sum_{j \in \mathcal{K}_m} (a'_{ij} - a_{ij}) v_j$ 添加到原生融合注意力输出中。在强度为零时， Δh_i 会被显式设为零，保存的图像在逐像素层面与原始输出完全相同。对于非零干预，掩码图像注意力权重总和的最大绝对偏差为2.38×10^{-7}，而所有其他键权重保持不变。

使用双三次插值将源图像映射到输出尺寸，并将 RGB 归一化为 $[0, 1]$。编辑框端点经过缩放并向下取整，使用不包含右下端点的半开区间。边界是框外宽度为 64像素的矩形环。设 $E_u^{(0)}$和 $E_u^{(r)}$分别表示原始输出和重新分配后输出的相对于源图像的区域均方误差。表 9 报告了$\delta_u = 100(E_u^{(r)}/E_u^{(0)} - 1)$，其中 $u = n$， b 分别表示非编辑区域和边界。

表9。**条件注意力权重与空间重新分配**。 每一行表示一个测量样本，所有值均以百分比表示。 $\bar{\mu}^{(s)}$ 表示原始 Fill模型分配给风格参考的平均总注意力权重。 δ_n 和 δ_b分别表示相对于原始输出，重新分配后非编辑区域和边界均方误差的变化；负值表示与源图像的一致性更高。

样本	$\bar{\mu}_e^{(s)}$	$\bar{\mu}_b^{(s)}$	δ_n	δ_b
样本 1	10.63	6.26	-0.004	-0.760
样本 2	10.37	5.91	+0.035	+0.210
样本 3	11.92	6.79	-0.159	+1.012
样本 4	13.31	6.07	+1.508	+1.405

目标文本与字形输入。 文本替换使用根据源图像构建的掩码图像 I_m 和风格参考 I_s ；阶段1重建仅使用 I_m。目标字符串 y 通过自然语言指令提供。训练期间，目标图像提供流匹配监督信号，而字形支持图仅为结构损失定义空间权重。推理时无需渲染目标字形。风格参考提供源文本外观，而目标文本与替换监督信号共同约束新字符。

C. 条件构建与评估实现

图像与坐标处理。 首先根据标注图像与源图像之间的尺寸比例映射标注坐标，然后通过图像缩放和裁剪将其转换至输出图像。掩码图像使用 Pillow 矩形填充，其中包含右下端点。风格裁剪图和评估掩码使用不包含该端点的半开区间，因此边界可能相差一个像素。如果标注缺少有效的图像尺寸，坐标映射将使用源图像的实际尺寸。参考图像根据已知的尺寸变换映射至输出图像。SSIM 首先计算源图像与输出图像之间的局部相似度图，然后对非编辑区域内各窗口中心的相似度取平均；窗口可能跨越编辑边界。LPIPS 在整幅图像上计算。

参考图像与大语言模型评估。 每项评估任务都使用固定的参考图像和评估模板。Gemini-3.0-pro 根据源图像和生成图像提供大语言模型评分；在编码前，两者都会调整尺寸，使最长边为 1024。大语言模型评估和人工评估结果分别报告于表 1 和 2中。

D. TBench上的配对比较

两种配置均使用一张标有红框的图像进行评估。按照第 6.1 节中参考模型比较的顺序，第一种配置在原始骨干网络上训练 Fill LoRA，第二种配置则在完全微调的骨干网络上训练单图像编辑 LoRA。两者使用不同的训练数据和分辨率设置。

TBench 结果按源图像配对，采用 10,000 次自助重采样，随机种子为 42。表10 中的差异是第 2 节中第一个配置减去第二个配置的结果。 6.1这些差异刻画了文本准确性与图像保持之间的权衡。区间反映图像重采样的变异性，但不包括生成随机种子、评分服务或训练随机性所带来的变异。

表10. TBench上的配对差异及95%自助法置信区间。OCR差异以比例表示。

指标	差异	95% 区间
OCR↑	$-.1055$	$[-.1658, -.0452]$
ED↓	$+.0933$	$[+.0526, +.1363]$
非编辑区域结构相似性指数↑	$+.0733$	$[+.0651, +.0820]$
全图学习感知图像块相似度↓	$-.0818$	$[-.0894, -.0742]$
LLM↑	$-.1960$	$[-.3920, .0000]$

E. 注意力传输干预设置

单条件编辑模型用于检验空间对应关系的功能作用。该模型采用与两阶段方法相同的主干网络，但使用独立训练的权重和单一视觉条件。所有干预均固定模型权重、条件和随机状态。

局部键抑制。 在 TBench 图像上，我们在第30层、去噪步骤 0 时抑制非编辑查询的条件键对数几率。对应键组采用半径为 8 个词元的邻域，每个查询抑制 $17 \times 17 = 289$ 个键，抑制强度为 50。随机键组的键数量、层、步骤和强度均与之匹配。

在图 12（a）中，90%的样本位于恒等线上方。逐样本均方误差比的均值为1.856，不同于两组均值之比。RGB 值归一化为 $[0, 1]$。随机键组与被抑制的注意力权重总量并不匹配，因此对高权重键的敏感性也可能影响结果。

图 13 提供了更多局部输出及其与未修改输出之间的差异。

条件潜变量错位。 在 TBench 图像上，错位实验在保持位置分配和其他推理设置不变的情况下，沿水平方向对条件潜在网格进行循环位移。图 12（b）报告了未编辑区域中升高的相对于源图像的误差；三种设置都会降低保留效果。

均值退化程度随位移增加而增加。循环位移保留了词元值集合，同时改变空间位置并引入边界环绕。循环位移与局部键抑制的干预强度不同；每项实验都将自身的干预与其对照进行比较。

不同层和去噪步之间的配对比较。图 12(c) 总结了九项实验中两组平均均方误差的比值。单元格值表示平均均方误差的比值，采用对数颜色刻度。图中仅显示经过评估的层和去噪步。

效应不确定性。 表 11 使用随机种子 42 进行 10,000 次成对图像自助重采样，并取均值分布的 2.5% 和 97.5% 分位数作为区间端点。每次重采样都保留干预–对照配对。区间反映当前图像集内的变异性，不包括重新训练或生成随机种子带来的变异性。

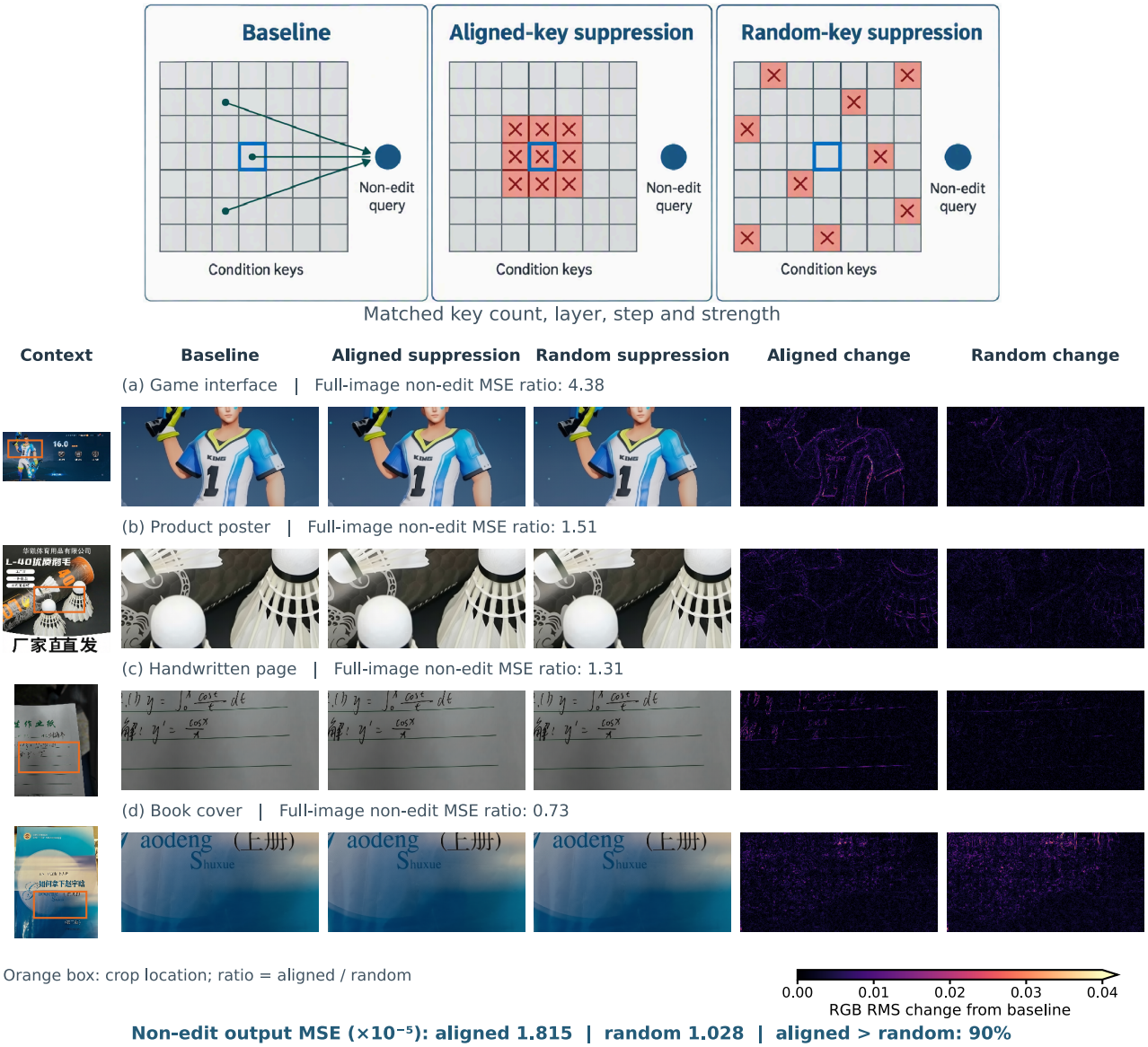

图 13. **局部键抑制的更多可视化结果。** 干预作用于第30层、去噪步骤0。上：对非编辑查询向量抑制局部对应键或随机键，并匹配键数量、层、步和抑制强度。下：游戏界面、产品海报、手写页面和书籍封面展示了强烈、接近中位数以及反转的干预效应。左侧缩略图中的橙色方框标出放大的非编辑区域。后续各列依次显示基线、对应键抑制、随机键抑制，以及两种干预相对于基线的 RGB均方根差异。每行报告对应键/随机键的扰动均方误差比，统计范围为整幅图像中的所有非编辑区域。差异在原生分辨率下使用统一色阶计算。底部统计结果显示，在90%的样本中，对应键抑制造成的扰动更大。层和步的统计数据见图 12（c）。

F. 条件内的空间注意力分布

为直接检查模型从条件中提取信息的位置，我们重新使用附录 B.4 中双条件 Fill 配置的原始 Q/K。我们分析去噪步骤0时，Transformer第25层中输出查询在掩码图像上的空间注意力分布。对于输出查询 i 和注意力头 h，在掩码图像键集合

表 11。单一视觉条件下的配对干预效应。第一行比较相对于未修改输出的扰动；其余各行比较相对于源图像的非编辑区域均方误差的增幅。两种误差参照不可直接比较。区间通过成对逐图自助法获得。

干预对比	平均差	95%置信区间	正差值
对应键抑制 − 随机键抑制	7.867×10^{-6}	$[4.662, 11.510] \times 10^{-6}$	90%
偏移 1 个词元的条件 − 无偏移	0.01533	[0.01069, 0.02034]	全部为正
偏移 2 个词元的条件 − 无偏移	0.01895	[0.01357, 0.02459]	全部为正
条件偏移4个词元 − 无偏移	0.02304	[0.01683, 0.02932]	全部为正

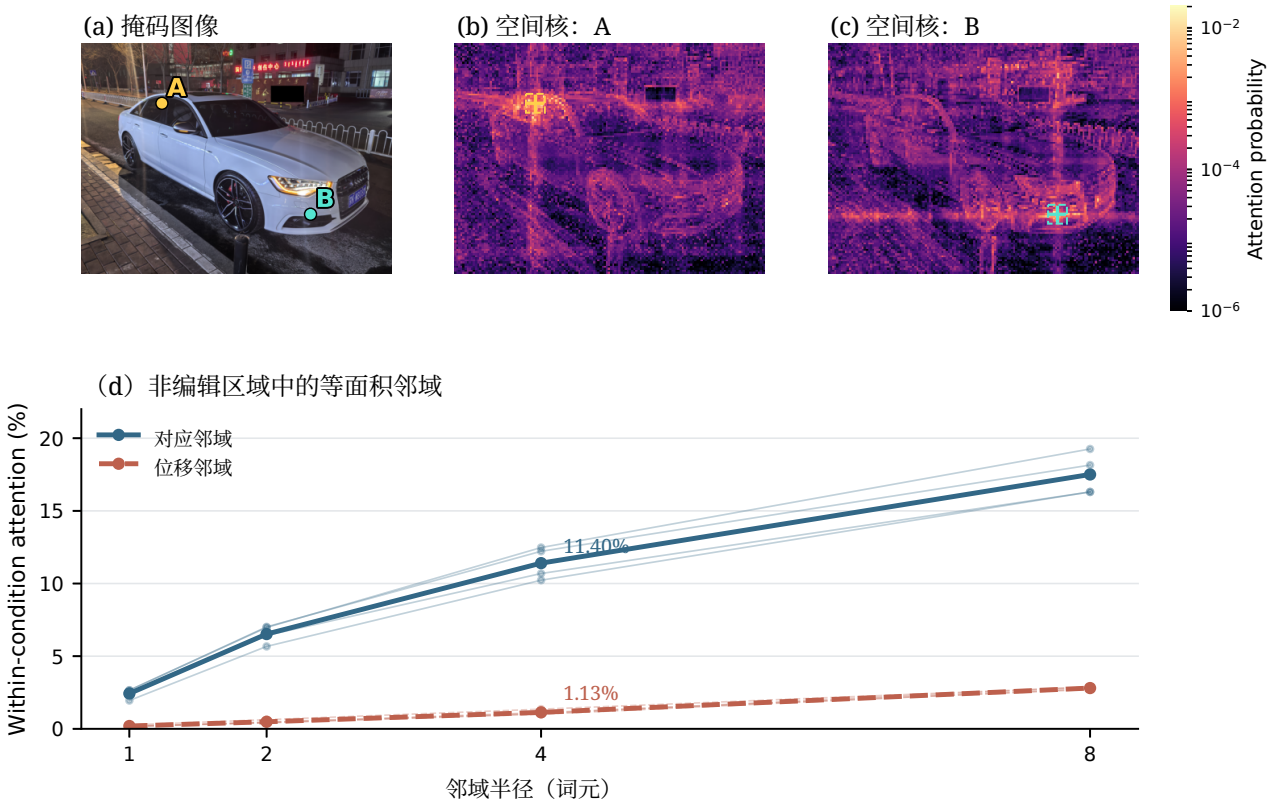

图14. **条件内的空间注意力分布**。(a) 标出了两个非编辑查询位置A和B的掩码图像，并显示其空间对应关系。(b,c) 两个查询在掩码图像上的空间聚合核。每个像素对应一个条件键位置；叉号标出对应位置，虚线框表示半径为4个词元的邻域。两幅图使用相同的对数颜色刻度。(d) 对应邻域与大小相同的偏移邻域中的注意力占比。细线表示单个样本，粗线表示样本均值。测量采用双条件Fill，在去噪步骤0、Transformer第25层进行。每个注意力头内按掩码图像对注意力进行归一化，然后跨注意力头取平均。层和步骤均从0开始编号。

$\mathcal{K}_m$ 内归一化后得到

$$\pi_{ij}^{(h,m)} = \frac{\exp(q_i^{(h)\top} k_j^{(h)}/\sqrt{d})}{\sum_{k\in\mathcal{K}_m} \exp(q_i^{(h)\top} k_k^{(h)}/\sqrt{d})}, \qquad \bar{\pi}_{ij}^{(m)} = \frac{1}{N_h}\sum_h \pi_{ij}^{(h,m)}. \tag{21}$$

对于固定的查询 i，我们根据条件键 j 的空间位置排列 $\bar{\pi}_{ij}^{(m)}$。该分布总和为 1，并描述

掩码图像中的检索位置，不包括其注意力权重总和。

查询位置与空间检索。图14（a–c）使用与图10相同的标志场景。我们在固定网格上，选择两个有效的非编辑查询向量，其位置最接近图像宽度和高度各自1/4 和3/4 处的位置。随着查询向量从 A 移动到 B，较强的注意力响应会移向对应的条件邻域，展示了输出位置与源检索位

置之间的空间对应关系。

对应邻域与位移邻域。 我们进一步在固定的24×24查询网格上统计非编辑区域。由于输出与掩码图像共享同一潜在网格，查询 i 对应的键位置具有相同的网格坐标。我们以该位置为中心，定义切比雪夫半径为 r 的正方形邻域$\mathcal{N}_r(i)$，其注意力占比为

$$C_i(r) = \sum_{j\in\mathcal{N}_r(i)} \bar{\pi}_{ij}^{(m)}, \qquad r \in \{1,2,4,8\}. \tag{22}$$

位移对照将邻域中心水平或垂直移动约网格宽度或高度的三分之一，仅保留完全位于非编辑区域内的有效邻域，并对其注意力占比取均值。两组中每个邻域均包含 $(2r+1)^2$ 个键，不进行边界裁剪或循环环绕。所有半径均使用同一组查询。在最大半径 8 处，排除与编辑区域或图像边缘相接的邻域后，剩余 1,463 个查询。我们先对每幅图像内的查询取均值，再对所有图像进行等权平均。

在图 14(d) 中，在每个测试半径下，对应邻域获得的注意力都多于位移邻域。对于 $r=4$，它们的注意力占比均值分别为 11.40% 和 1.13%，比值约为 10.06。在此配置下，掩码图像内的空间聚合更倾向于与输出位置对应的邻域，为局部源检索提供了分布层面的证据。其对生成的功能影响将结合图 12 和 4中的干预一并评估。

G. 输出变化分布的统计定义

ˆ ˜对于输出 I 和映射到输出尺寸的源图像 I_{src}，我们将逐像素平方变化定义为

$$D(p) = \frac{1}{3}\sum_{c=1}^{3}[\hat{I}(p,c) - \tilde{I}_{\text{src}}(p,c)]^2. \tag{23}$$

图 9 $C_u = |\ _u|^{-1}\sum_{p\in\ _u} D\ p$ $u\ \in$ (a) 报告了 的 ΩΩ()。面板 (b) 比较了编辑区域面积占比与其在总平方变化中的占比：

$$\eta_e = \frac{\sum_{p\in\Omega_e} D(p)}{\sum_{p\in\Omega} D(p)}. \tag{24}$$

C_e 衡量编辑区域内的内容变化，η_e 描述变化的空间集中度。目标字符的正确性通过文本准确率指标进行评估。

本分析使用附录中双条件 Fill参考 配置的 TBench 输出 B.4，其中

实际生成尺寸和 RGB 归一化为 $[0,1]$。源图像通过双三次插值映射到输出网格。输出尺寸遵循保持宽高比的面积缩放，并四舍五入为 32 的倍数。

编辑框使用固定评估标注列表中的原图坐标，并缩放至输出尺寸。左上端点向下取整，右下端点向上取整，采用不包含后者的半开区间。变化统计遵循本节中的定义。图 9(c) 使用逐图像光学字符识别判定和结构相似性指数分数。

H. 参数适配分析

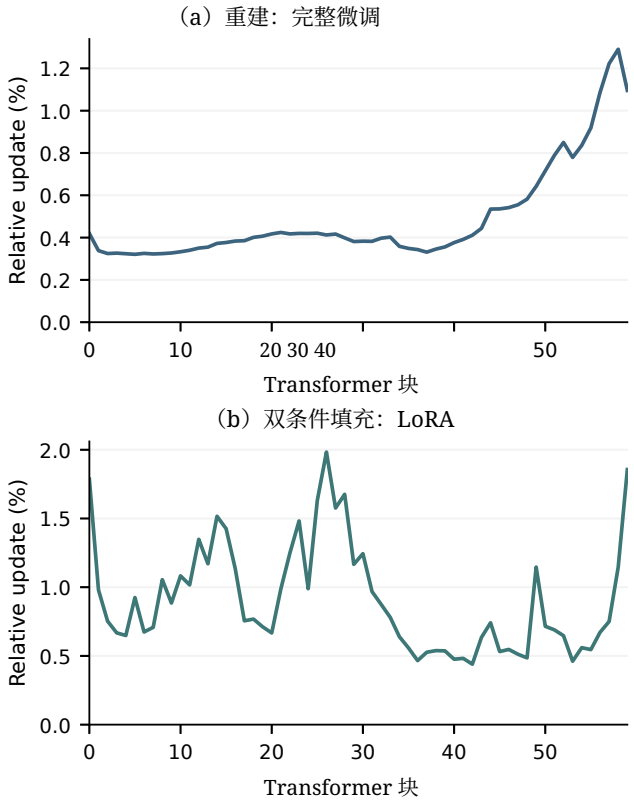

图 15. **逐层相对参数更新**。 重建包含每层中的所有参数，而 Fill 则包含 LoRA 覆盖的投影模块。更新量以相应初始参数的弗罗贝尼乌斯范数进行归一化。层索引从 0 开始。

我们分析在自监督文本区域重建和双条件 Fill 参考模型训练过程中，参数在哪些位置发生了适配。对于重建，我们使用第 1 轮训练结束时的全参数检查点；对于 Fill，我们使用第 4 轮训练结束时秩为 128 的 LoRA 权重，并采用附录 B.4中的参考配置。这两项分析分别刻画了各自配置下跨层的更新分布；这些配置在训练设置、检查点轮次和参数覆盖范围方面有所不同。

对于某一层内的参数组 G，我们将相对更新定义为

$$R_G = \left(\frac{\sum_{m \in G} \|\Delta W_m\|_F^2}{\sum_{m \in G} \|W_m^{(0)}\|_F^2} \right)^{1/2}. \tag{25}$$

此处，$W_m^{(0)}$ 表示初始参数。对于全参数微调，$\Delta W_m = W_m - W_m^{(0)}$。对于 LoRA，$\Delta W_m = (\alpha/r) B_m A_m$，其中在此配置中，$\alpha = r = 128$。

图 15 对于重建，后续层的相对更新幅度更大，在第 58 层达到 1.29%。对于 Fill，较大的更新出现在网络中间层以及两端。相对更新刻画了参数适配的位置和幅度，而干预和消融实验则评估模块的功能作用。