

# PR #32999 完整报告

sgl-project/sglang

[diffusion] Fix GLM-Image resolution alignment

合并时间: 2026-08-06 20:41

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/32999>

## 执行摘要

- 一句话: 修复 GLM-Image 非对齐分辨率统一向上取整到 32 倍数并新增 resize 字段
- 推荐动作: 值得精读。该 PR 展示了一个跨阶段一致性 bug 的系统性修复思路: 把对齐逻辑收敛到单一工具函数, 在入口处提前归一化并增加校验防线, 同时通过响应字段向用户透明化行为变更。关注点包括 glmimage.py 的对齐实现、AR 阶段的校验、以及 \_get\_response\_resize 对真实尺寸的读取回退逻辑。

## 功能与动机

issue #32996 报告: GLM-Image 生成非对齐分辨率时内部张量形状不匹配 (`RuntimeError: The size of tensor a (961) must match the size of tensor b (900)`), OpenAI 兼容端点返回 HTTP 500。根因是 AR 阶段把 500x500 向下取整为 480 生成 30x30 token 网格, 而扩散阶段从原始 500 推导出 31x31 网格。官方文档要求目标分辨率必须能被 32 整除, 因此需要统一各阶段的有效分辨率, 并在响应中告知用户实际生成尺寸。

## 实现拆解

1. 统一对齐策略 (核心逻辑): 在 `python/sglang/multimodal_gen/configs/sample/glmimage.py` 中新增 `GLM_IMAGE_RESOLUTION_ALIGNMENT = 32` 常量、`align_glm_image_dimension()` (向上取整到 32 的倍数, 最小为 32) 和 `align_glm_image_resolution()`。GlmImageSamplingParams.\_adjust() 在调度前把显式 width/height 对齐, 并在发生变化时输出 warning 日志, 告知用户请求尺寸到实际尺寸的调整。
2. AR 阶段同步对齐与校验: 在 `python/sglang/multimodal_gen/runtime/pipelines_core/stages/model_specific_stages/glm_image.py` 中, 新增 `resize_glm_image_to_alignment()` 将编辑条件图用 LANCZOS 放大到 D32 网格; `forward()` 中对运行时推断出的宽高调用 `align_glm_image_resolution()`, 并删除原先独立的向下取整 `((height // factor) * factor)`; 新增 `_validate_glm_image_resolution_alignment()`, 在 `generate_prior_tokens()` 与 `generate_prior_tokens_batch()` 入口校验维度必须已对齐, 否则抛 `ValueError`, 确保错误在到达模型前被拦截。
3. OpenAI 响应数据契约扩展: 在 `python/sglang/multimodal_gen/runtime/entrypoints/openai/protocol.py` 的 `ImageResponseData` 中新增可选字段 `resize`; 在 `image_api.py` 中新增 `_get_response_resize()`, 优先读取落盘图片的真实尺寸 (覆盖生成后超分场景), 读取失败或无法读取时回退到采样画布的对齐尺寸; `_build_image_response_kwargs()` 接受

resize 参数并写入每个 data 项, /generations 与 /edits 端点均接入。

4. 测试配套: 新增 / 扩展 4 个单测文件覆盖: 采样参数向上取整与最小对齐、告警触发条件、运行时维度对齐、编辑图 resize 传播、未对齐输入的內部校验拒绝、响应 resize 字段的填充逻辑与非 GLM-Image 场景留空, 以及 size/width/height 优先级解析。

关键文件:

- python/sglang/multimodal\_gen/configs/sample/glmimage.py (模块 采样参数; 类别 source; 类型 core-logic; 符号 \_adjust, align\_glm\_image\_dimension, align\_glm\_image\_resolution) : 新增统一的 D32 对齐核心逻辑, 定义对齐常量和向上取整函数, 并在采样参数调整阶段提前对齐与告警, 是本次修复的策略源头。
- python/sglang/multimodal\_gen/runtime/pipelines\_core/stages/model\_specific\_stages/glm\_image.py (模块 AR 阶段; 类别 source; 类型 data-contract; 符号 resize\_glm\_image\_to\_alignment, \_validate\_glm\_image\_resolution\_alignment) : AR 阶段在此应用对齐策略: 运行时尺寸向上取整、编辑条件图统一放大、并增加进入 token 生成前的强校验, 是消除 500 错误的关键现场。
- python/sglang/multimodal\_gen/runtime/entrypoints/openai/image\_api.py (模块 图像接口; 类别 source; 类型 endpoint; 符号 \_get\_response\_resize) : OpenAI 兼容入口新增 data[].resize 字段, 负责向用户透明报告实际返回图像尺寸, 是本次讨论中确定的用户可见补偿机制。
- python/sglang/multimodal\_gen/runtime/entrypoints/openai/protocol.py (模块 响应协议; 类别 source; 类型 core-logic) : 响应数据模型 ImageResponseData 新增可选的 resize 字段, 是本次 API 契约变更的直接落点。
- python/sglang/multimodal\_gen/test/unit/test\_glm\_image\_ar.py (模块 AR 测试; 类别 test; 类型 test-coverage; 符号 test\_forward\_aligns\_runtime\_dimensions\_before\_ar\_generation, test\_forward\_resizes\_edit\_image\_up\_to\_d32\_grid, test\_generate\_prior\_tokens\_rejects\_unaligned\_internal\_dimensions, test\_generate\_prior\_tokens\_batch\_rejects\_unaligned\_internal\_dimensions) : 覆盖 AR 阶段运行时维度对齐、编辑图 resize 传播以及未对齐输入被拒绝的校验逻辑, 是验证核心修复行为的直接测试。
- python/sglang/multimodal\_gen/test/unit/test\_sampling\_params.py (模块 参数测试; 类别 test; 类型 test-coverage; 符号 test\_glm\_image\_rounds\_resolution\_up\_to\_multiple\_of\_32, test\_glm\_image\_resolution\_rounds\_up, test\_glm\_image\_resolution\_keeps\_minimum\_alignment, test\_glm\_image\_does\_not\_warn\_for\_aligned\_resolution) : 验证采样参数阶段向上取整、最小对齐、告警触发条件等行为。
- python/sglang/multimodal\_gen/test/unit/test\_openai\_image\_api.py (模块 接口测试; 类别 test; 类型 test-coverage; 符号 test\_image\_response\_includes\_resize\_for\_every\_output, test\_response\_resize\_is\_only\_populated\_for\_glm\_image, test\_response\_resize\_uses\_actual\_generated\_image\_size, test\_response\_resize\_prefers\_final\_output\_over\_sampling\_canvas) : 验证响应 resize 字段的填充逻辑: 仅 GLM-Image 填充、使用实际输出尺寸而非采样画布。

- python/sglang/multimodal\_gen/test/unit/test\_openai\_utils.py (模块 工具测试; 类别 test ; 类型 test-coverage; 符号 test\_build\_sampling\_params\_resolves\_size\_and\_explicit\_dimensions, fake\_from\_user\_sampling\_params\_args) : 补充 size/width/height 优先级解析测试, 确保显式尺寸与 size 字段组合时的行为稳定。

关键符号: align\_glm\_image\_dimension, align\_glm\_image\_resolution, GlmImageSamplingParams.\_adjust, resize\_glm\_image\_to\_alignment, \_validate\_glm\_image\_resolution\_alignment, \_get\_response\_resize

## 关键源码片段

### python/sglang/multimodal\_gen/configs/sample/glmimage.py

新增统一的 D32 对齐核心逻辑, 定义对齐常量和向上取整函数, 并在采样参数调整阶段提前对齐与告警, 是本次修复的策略源头。

```
# GLM-Image 要求分辨率两个维度都是 32 的倍数。
```

```
# 这里统一采用“向上取整”策略, 并保证最小对齐尺寸为 32。
```

```
from dataclasses import dataclass
```

```
from sglang.multimodal_gen.configs.sample.sampling_params import SamplingParams
```

```
from sglang.multimodal_gen.runtime.utils.logging_utils import init_logger
```

```
logger = init_logger(__name__)
```

```
GLM_IMAGE_RESOLUTION_ALIGNMENT = 32
```

```
def align_glm_image_dimension(value: int) -> int:
```

```
    """把单个维度向上取整到 32 的倍数, 最小值为 32, 避免非正输入产生非法尺寸。"""
```

```
    return max(
```

```
        GLM_IMAGE_RESOLUTION_ALIGNMENT,
```

```
        (value + GLM_IMAGE_RESOLUTION_ALIGNMENT - 1)
```

```
        // GLM_IMAGE_RESOLUTION_ALIGNMENT
```

```
        * GLM_IMAGE_RESOLUTION_ALIGNMENT,
```

```
    )
```

```
def align_glm_image_resolution(width: int, height: int) -> tuple[int, int]:
```

```
    """同时对齐宽高, 两个维度彼此独立。"""
```

```
    return align_glm_image_dimension(width), align_glm_image_dimension(height)
```

```
@dataclass
```

```
class GlmImageSamplingParams(SamplingParams):
```

```
    # 省略其它字段定义……
```

```
    num_frames: int = 1
```

```
    guidance_scale: float = 1.5
```

```
    num_inference_steps: int = 30
```

```

def _adjust(self, server_args):
    # 在调度前把显式请求的尺寸统一对齐，并打印一次告警，
    # 让用户知道实际生成尺寸与请求尺寸的差异。
    requested_width = self.width
    requested_height = self.height
    if self.width is not None and self.height is not None:
        self.width, self.height = align_glm_image_resolution(
            self.width, self.height
        )
    if (self.width, self.height) != (requested_width, requested_height):
        logger.warning(
            "GLM-Image requires dimensions divisible by %s; adjusted "
            "requested resolution from %sx%s to %sx%s",
            GLM_IMAGE_RESOLUTION_ALIGNMENT,
            requested_width,
            requested_height,
            self.width,
            self.height,
        )
    super()._adjust(server_args)

```

## python/sglang/multimodal\_gen/runtime/pipelines\_core/stages/model\_specific\_stages/glm\_image.py

AR 阶段在此应用对齐策略：运行时尺寸向上取整、编辑条件图统一放大、并增加进入 token 生成前的强校验，是消除 500 错误的关键现场。

```

# AR 阶段：运行期推断出的尺寸也统一走同一个 D32 对齐策略，
# 并把未对齐输入直接挡在校验层，避免再次出现 900 vs 961 的形状错配。
from PIL import Image

```

```

from sglang.multimodal_gen.configs.sample.glmimage import (
    GLM_IMAGE_RESOLUTION_ALIGNMENT,
    align_glm_image_resolution,
)

```

```

def resize_glm_image_to_alignment(image: Image.Image) -> Image.Image:
    """把编辑条件图放大到 D32 网格，保证 AR 编码与扩散条件看到同一张图。"""
    width, height = image.size
    aligned_width, aligned_height = align_glm_image_resolution(width, height)
    if (aligned_width, aligned_height) == (width, height):
        return image
    return image.resize((aligned_width, aligned_height), Image.Resampling.LANCZOS)

```

```

def _validate_glm_image_resolution_alignment(width: int, height: int) -> None:
    """内部防线：任何进入 AR token 生成的尺寸必须是 32 的倍数。"""
    if (
        height % GLM_IMAGE_RESOLUTION_ALIGNMENT != 0

```

```

        or width % GLM_IMAGE_RESOLUTION_ALIGNMENT != 0
    ):
        raise ValueError(
            "GLM-Image dimensions must be aligned before AR token generation, "
            f"got {width}x{height}"
        )

```

## python/sglang/multimodal\_gen/runtime/entrypoints/openai/image\_api.py

OpenAI 兼容入口新增 `data[].resize` 字段，负责向用户透明报告实际返回图像尺寸，是本次讨论中确定的用户可见补偿机制。

```

# OpenAI 兼容响应：新增 data[].resize 字段，报告实际返回图像的尺寸。
# 优先读取落盘后的真实尺寸，以便覆盖生成后的超分（upscaling）场景；
# 读取失败时回退到采样画布的对齐尺寸。
from PIL import Image

```

```

from sglang.multimodal_gen.configs.sample.glmimage import GlmImageSamplingParams
from sglang.multimodal_gen.configs.sample.sampling_params import SamplingParams

```

```

def _get_response_resize(
    sampling_params: SamplingParams, output_path: str | None = None
) -> str | None:
    """返回 GLM-Image 输出的实际尺寸字符串（WIDTHxHEIGHT），非 GLM-Image 返回 None。"""
    if not isinstance(sampling_params, GlmImageSamplingParams):
        return None

    if output_path is not None:
        try:
            with Image.open(output_path) as output_image:
                width, height = output_image.size
            return f"{width}x{height}"
        except (OSError, ValueError):
            # 自定义输出传输等场景下无法读取文件时，退回到采样画布尺寸
            pass

    if sampling_params.width is None or sampling_params.height is None:
        return None
    return sampling_params.output_size_str()

```

## 评论区精华

Issue 讨论中，用户 [Makcum888e](#) 提出“是否可以在对齐时通知用户”，[ChefWu551](#) 提议在响应中通过 `usage` 或 `data[].resize` 返回实际尺寸，[Makcum888e](#) 认可 `resize` 字段方案。另一个讨论点：是否应直接返回用户请求的精确尺寸（先按 32 倍数生成再缩放 / 放大），[AuFlow](#) 询问向上取整 + 中心裁剪还是向下取整 + 缩放的偏好，[Makcum888e](#) 作为用户偏好 `upscaling`，但同意作为后续 PR。评审结论是：本期采用统一向上取整并在 `data[].resize` 中告知实际尺寸，精确尺寸支持留待 follow-up。

- 是否告知用户分辨率对齐及方式 (question): 采用 `data[].resize` 字段报告实际返回图像尺寸，并在 PR 中落地。
- 是否直接返回用户请求的精确尺寸 (design): 当前 PR 采用统一向上取整并在响应中报告实际尺寸，精确尺寸支持留待 follow-up PR。

## 风险与影响

- 风险:
  1. 行为变更风险: 非对齐请求现在会被放大到 32 倍数 (如  $500 \times 500 \rightarrow 512 \times 512$ )，用户可能得到与请求不同的图像尺寸，可能影响依赖精确尺寸的调用方。已通过 `resize` 字段缓解，但属于可感知行为变化。
  2. i2i 编辑效果风险: `glm_image.py` 中对编辑条件图统一执行 LANCZOS 放大 (`resize_glm_image_to_alignment`)，可能轻微改变编辑图像的内容或画布比例，需要回归验证编辑质量。
  3. 内部校验收紧风险: `generate_prior_tokens()` 与 `generate_prior_tokens_batch()` 现在对未对齐输入直接抛 `ValueError`。若未来新增调用路径未先对齐，会由运行时错误替代原本的静默错误，但当前所有入口均已对齐，影响可控。
  4. 数据契约扩展风险: `ImageResponseData` 新增可选字段 `resize` 是向后兼容的，但下游若做严格 schema 校验 (如禁止未知字段) 需更新预期。
    - 影响: 受影响用户: 使用 GLM-Image 模型并通过 OpenAI 兼容接口 (`/v1/images/generations`、`/v1/images/edits`) 传入非 32 倍数分辨率的用户，原先会遇到 HTTP 500，现在可正常生成且响应中携带实际尺寸。
    - 系统层面, GLM-Image 管线各阶段 (采样参数、AR 阶段、扩散阶段) 首次统一了分辨率对齐策略，消除了同类形状不匹配隐患。
    - 团队层面, 新增了响应字段说明文档待补充，且后续需要跟进精确尺寸 (upscaling) 支持。
    - 风险标记: 核心路径变更: AR 阶段对齐与校验，OpenAI 响应数据契约新增字段，行为变更: 非对齐分辨率将增大渲染，i2i 条件图被 LANCZOS 放大

## 关联脉络

- PR #33734 [diffusion] ERNIE-Image bit-exact residual-gate fast path (H200  $1024^2 \times 2 \times 16.17 \rightarrow 15.75$  s): 同为 diffusion 模型管线优化，与 GLM-Image 共享 `multimodal_gen` 运行时基础设施，体现 diffusion 服务持续演进趋势。
- PR #33725 [diffusion] feat: data-parallel serving (`--dp-size`): 同属 diffusion 服务化能力增强，说明本 PR 所在的 diffusion 功能线正在系统性地补齐服务质量和可用性。