

PR #43586 完整报告

vllm-project/vllm

[MM][Perf][CG] Support dual-path ViT full CUDA graph for DeepSeek-OCR

合并时间: 2026-06-16 19:35

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/43586>

执行摘要

- 一句话: DeepSeek-OCR 视觉编码器双路径 CUDA 图形支持
- 推荐动作: 该 PR 值得精读, 尤其是双路径图的设计模式: 如何通过独立预算和路径回退避免动态形状问题。EncoderCudaGraphManager 的泛化支持 (dict[str, dict]) 为未来多模型多路径图提供了扩展基础。建议关注 CI 测试的重新开启和显存优化。

功能与动机

DeepSeek-OCR 使用双塔 ViT (SAM+CLIP) 和动态平铺机制, 单个 CUDA 图无法适应可变 crop_shape 和 patch 数量。传统单图会导致大量零填充或图失效。双路径设计允许独立捕获全局和局部图, 最大化图命中率, 提升推理效率。PR body 提到依赖 #41234 和 #42288, 并引用 DeepSeek-OCR 技术报告。

实现拆解

1. 模型侧实现 (deepseek_ocr.py): 在 DeepseekOCRForCausalLM 中添加 _get_num_input_output_tokens、get_encoder_cudagraph_config、prepare_encoder_cudagraph_capture_inputs、encoder_cudagraph_forward、prepare_encoder_cudagraph_replay_buffers、encoder_eager_forward 及 postprocess_encoder_output, 实现 SupportsEncoderCudaGraph 协议。新增属性 image_side、global_image_output_token、patch_side、single_patch_output_token 用于计算固定 token 预算。get_encoder_cudagraph_config 返回配置时启用 enable_dual_path_graph=True, 并注册 images_crop 为缓冲区键。
2. 管理器扩展 (encoder_cudagraph.py): 将 budget_graphs 类型从 dict[int, BudgetGraphMetadata] 改为 dict[str, dict[int, BudgetGraphMetadata]], 新增 global_token_budgets 和 local_token_budgets 独立预算列表。capture() 方法按路径 ('global'/'local') 分别调用 _capture_budget_graph, 新增 _get_graph_set 辅助方法安全获取图集合。新增 _execute_local_single_path 和 _execute_local_dual_path 进行推理时的路径选择。
3. 数据契约扩展 (encoder_cudagraph_defs.py): 在 EncoderCudaGraphConfig 中添加 enable_dual_path_graph、global_token_per_image、local_token_per_patch 字段; 在 EncoderItemSpec 中添加 global_output_tokens 和 local_output_tokens。
4. 接口同步 (interfaces.py + 7 个模型文件): 在 SupportsEncoderCudaGraph 协议方法中添加 path: str = 'default' 和 local_output: Optional[torch.Tensor] = None 参数, 所

有现有模型实现按默认值向后兼容。

5. 测试和文档：在 `test_vit_cudagraph.py` 添加 `deepseek_ocr` 配置项（启用 `dummy weight`，跳过执行）；更新 `docs/design/cuda_graphs_multimodal.md` 在支持列表中增加 DeepSeek-OCR。

关键文件：

- `vllm/model_executor/models/deepseek_ocr.py`（模块 模型层；类别 `source`；类型 `data-contract`；符号 `DeepseekOCRForCausalLM`, `_assemble_patch_grid`, `image_side`, `global_image_output_token`）：模型主实现文件，新增 `SupportsEncoderCudaGraph` 协议全部方法，实现双路径图形配置和 `token` 计算。
- `vllm/v1/worker/encoder_cudagraph.py`（模块 CUDA 图管理器；类别 `source`；类型 `core-logic`；符号 `_capture_budget_graph`, `_get_graph_set`, `_execute_local_single_path`, `_execute_local_dual_path`）：CUDA 图管理器核心逻辑，扩展支持双路径模式：`budget_graphs` 改为 `dict[str, dict[int, ...]]`，新增单独 `global/local` 预算列表，`capture/execute` 分流。
- `vllm/v1/worker/encoder_cudagraph_defs.py`（模块 图配置定义；类别 `source`；类型 `data-contract`；符号 `global_output_tokens`, `local_output_tokens`, `enable_dual_path_graph`, `global_token_per_image`）：定义双路径相关配置字段和数据结构，是双路径图的数据契约基础。
- `vllm/model_executor/models/interfaces.py`（模块 接口层；类别 `source`；类型 `data-contract`；符号 `postprocess_encoder_output`, `prepare_encoder_cudagraph_capture_inputs`, `prepare_encoder_cudagraph_replay_buffers`, `encoder_cudagraph_forward`）：`SupportsEncoderCudaGraph` 协议接口调整，为每个方法添加 `path` 参数和 `local_output`，以支持双路径传递。
- `tests/models/multimodal/generation/test_vit_cudagraph.py`（模块 CUDA 图测试；类别 `test`；类型 `test-coverage`）：集成测试新增 DeepSeek-OCR 配置，但当前被跳过（OOM）。

关键符号：`get_encoder_cudagraph_config`, `_get_num_input_output_tokens`, `_capture_budget_graph`, `_get_graph_set`, `postprocess_encoder_output`

关键源码片段

`vllm/model_executor/models/deepseek_ocr.py`

模型主实现文件，新增 `SupportsEncoderCudaGraph` 协议全部方法，实现双路径图形配置和 `token` 计算。

```
@property
def image_side(self) -> int:
    # 每个全局图像每维的输出网格单元数
    return math.ceil((BASE_SIZE // 16) / 4) # 16

@property
def global_image_output_token(self) -> int:
    # 每个全局图像的 token 数：网格 + 每行一个换行
```

```

        return self.image_side * (self.image_side + 1) # 272

@property
def patch_side(self) -> int:
    # 每个局部补丁每维的网格单元数
    return math.ceil((IMAGE_SIZE // 16) // 4) # 10

@property
def single_patch_output_token(self) -> int:
    # 每个局部补丁的 token 数
    return self.patch_side * (self.patch_side + 1) # 110

def get_encoder_cudagraph_config(self) -> EncoderCudaGraphConfig:
    # 返回编码器 CUDA 图配置，启用双路径模式
    return EncoderCudaGraphConfig(
        modalities=['image'],
        input_key_by_modality={'image': 'pixel_values'},
        buffer_keys=['images_crop'],
        out_hidden_size=self.config.vision_config.hidden_size,
        enable_dual_path_graph=True,
        global_token_per_image=self.global_image_output_token,
        local_token_per_patch=self.single_patch_output_token,
    )

```

vllm/v1/worker/encoder_cudagraph.py

CUDA 图管理器核心逻辑，扩展支持双路径模式：budget_graphs 改为 dict[str, dict[int, ...]]，新增单独 global/local 预算列表，capture/execute 分流。

```

def capture(self, graph_pool: Any):
    # 按路径分别捕捉 CUDA 图
    self.graph_pool = graph_pool

    if self.config.enable_dual_path_graph:
        # 双路径模式：先捕捉所有全局图，再捕捉局部图
        for token_budget in sorted(self.global_token_budgets, reverse=True):
            self._capture_budget_graph(token_budget, path='global')
        for token_budget in sorted(self.local_token_budgets, reverse=True):
            if token_budget == 0: # 跳过零预算（表示无局部补丁）
                continue
            self._capture_budget_graph(token_budget, path='local')
        logger.info(
            'Encoder CUDA graph capture complete. '
            'Captured %d global + %d local budget graphs.',
            len(self.budget_graphs['global']),
            len(self.budget_graphs['local']),
        )
    return

# 默认单路径模式（保持向后兼容）

```

```

for token_budget in sorted(self.token_budgets, reverse=True):
    self._capture_budget_graph(token_budget, path='default')
logger.info(
    'Encoder CUDA graph capture complete. Captured %d budget graphs.',
    len(self.budget_graphs['default']),
)

```

评论区精华

gemini-code-assist[bot] 指出了四个关键问题：（1）_get_num_input_output_tokens 中缺少对 image_spatial_crop 为 None 的空检查，会导致 TypeError；（2）EncoderCudaGraphConfig 应使用 input_key_by_modality 而非 input_keys；（3）images_crop 张量必须注册到 buffer_keys 并包含在 capture/replay 缓冲区中；（4）需要实现 get_max_frames_per_video 方法以遵循协议。这些均被作者采纳。

Isotr0py 建议将 image_side 等计算改为 property 避免重复，以及将 budget_graphs 改为 dict[str, ...] 以支持多路径，作者采纳。

关于双路径图设计，Isotr0py 最初认为局部补丁是计算瓶颈，建议优先捕获局部图；shen-shanshan 回应并实现了双路径独立选择方案，使全局和局部可同时拥有图，并且支持路径级回退。

- images_crop 必须注册到 CUDA 图缓冲区 (correctness): 作者在 get_encoder_cudagraph_config 的 buffer_keys 中增加了 'images_crop'，并确保 capture_inputs 和 replay_buffers 中包含此字段。
- 双路径预算图存储结构 (design): 作者采纳建议，重构 budget_graphs 结构，并新增 _get_graph_set 辅助方法。
- 局部补丁计算效率讨论及双路径方案选择 (design): 双路径图设计允许每条路径独立选择最小合适预算，实现部分图回退，比仅捕获局部图更灵活。

风险与影响

- 风险：
 1. 显存增加：双路径模式需捕获两套图（全局 + 局部），CUDA 图捕获数量和总显存占用翻倍，可能对大 batch 或低显存环境造成压力。
 2. 测试覆盖不足：CI 测试因 OOM 被标记为 skip（见 commit skip deepseek-ocr ci），缺乏回归检测，可能遗漏边缘情况。
 3. 接口适配面广：interfaces.py 及 7 个模型文件同步修改协议方法签名，虽然向后兼容，但若其他分支未同步合并可能导致冲突。
 4. 动态形状交互：双路径图在 batch 内不同图像的分块状态可能不同（有 / 无局部补丁），图选择逻辑需要处理这种混合情况，但当前实现可能没覆盖所有组合。- 影响：对 DeepSeek-OCR 模型：端到端视觉编码延迟降低 5-17%（小图 224x224 达 13.93%，大图 1024x1024 约 4.75%）。对用户无功能影响，但需显存更多。对系统：CUDA 图管理器新增双路径模式，但仅当模型配置 enable_dual_path_graph=True 时激活，不影响其他 ViT 模型。对其他模型：接口扩展仅增加可选参数，向后兼容。- 风险标记：测试跳过，显存增加，核心路径变更，多模型接口适配

关联脉络

- 暂无明显关联 PR