PR #27084 完整报告

sgl-project/sglang

[diffusion] Optimize Cosmos3 i2v latent prep

合并时间：2026-06-03 10:23

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27084

执行摘要

- 一句话：优化 Cosmos3 I2V 潜变量预处理，减少 70% 阶段耗时
- 推荐动作：建议合并。这是一个干净的微小优化，改动明确、性能数据扎实、风险极低。代码库维护者可关注是否存在类似潜在冗余操作（例如其他 diffusion 模型的 I2V 预处理）。

功能与动机

PR body 中提到："cosmos3 I2V only uses the first encoded latent frame for image conditioning, but the pipeline was expanding the input image to the full requested video length before VAE encode, which made I2V latent preparation do unnecessary VAE work and allocate a repeated video tensor." 即发现管线将输入图像扩展到完整视频长度后才进行 VAE 编码，而实际上 I2V 仅使用第一个编码后的潜变量帧，导致额外计算和内存浪费。

实现拆解

1. 识别冗余操作：在 Cosmos3LatentPreparationStage 的 forward 方法中，原始代码在 I2V 分支中先将 batch.preprocessed_image 通过 unsqueeze(2) 增加帧维度，然后通过 .expand(-1, -1, batch.num_frames, -1, -1).contiguous() 将单帧图像扩展为完整视频帧数（batch.num_frames），最后才进行 VAE 编码。
2. 移除不必要的扩展：将上述操作替换为直接对 unsqueeze(2) 后的单帧张量进行 VAE 编码。改动集中在 python/sglang/multimodal_gen/runtime/pipelines_core/stages/model_specific_stages/cosmos3.py 文件的第 324-328 行。
3. 保持后续逻辑不变：编码得到的 cond_latent 仍然是多帧潜变量，但后续仅使用第 0 帧（cond_latent[:, :, 0:1, :, :].clone()），因此编码结果在语义上等价。

关键文件：

- python/sglang/multimodal_gen/runtime/pipelines_core/stages/model_specific_stages/cosmos3.py（模块 扩散模型；类别 source；类型 core-logic）：核心改动文件，包含 Cosmos3 I2V 潜变量准备阶段的优化，是最关键的变更。

关键符号：Cosmos3LatentPreparationStage.forward

关键源码片段

python/sglang/multimodal_gen/runtime/pipelines_core/stages/model_specific_stages/cosmos3.py

核心改动文件，包含 Cosmos3 I2V 潜变量准备阶段的优化，是最关键的变更。

```
# 文件：python/sglang/multimodal_gen/runtime/pipelines_core/stages/model_specific_stages/cosmos3.py
# 位于 Cosmos3LatentPreparationStage.forward() 方法内

if is_i2v:
    vae_dtype = next(self.vae.parameters()).dtype
    # 优化前：将单帧图像扩展为 batch.num_frames 帧后再编码
    # pixel_video = (
    # batch.preprocessed_image.unsqueeze(2)
    # .expand(-1, -1, batch.num_frames, -1, -1)
    # .contiguous()
    # .to(device=device, dtype=vae_dtype)
    # )
    # 优化后：直接编码单帧图像（unsqueeze(2) 增加帧维度为 1）
    pixel_video = batch.preprocessed_image.unsqueeze(2).to(
        device=device, dtype=vae_dtype
    )
    with torch.no_grad():
        cond_latent = self._vae_encode(pixel_video).to(dtype)

    # condition_mask 和后续逻辑保持不变，仅使用 cond_latent 的第 0 帧
    condition_mask = torch.zeros(
        1, 1, num_latent_frames, 1, 1, device=device, dtype=dtype
    )
    condition_mask[:, :, 0, :, :] = 1.0
    latents = condition_mask * cond_latent + (1.0 - condition_mask) * noise
    batch.image_latent = cond_latent[:, :, 0:1, :, :].clone() # 只取第 0 帧
    batch.extra["velocity_mask"] = 1.0 - condition_mask
    self.log_info("Prepared I2V latents with frame-0 conditioning")
else:
    latents = noise
```

评论区精华

该 PR 的 review 由仓库维护者 mickqian 直接批准，无 review 评论。PR 作者 qimcis 在 body 中提供了详细的性能基准测试数据。

- 暂无高价值评论线程

风险与影响

- 风险：低风险。改动范围极小（仅 2 行增加、5 行删除），且逻辑清晰：VAE 编码前不再扩展帧维度，而后续逻辑只取编码结果的第 0 帧，因此语义等价。唯一潜在风险是如果未来代码修改了 cond_latent 的使用方式（例如改为使用多帧），但当前代码没有这种迹象。

- 影响：影响面窄：仅影响 Cosmos3 I2V 管线的 Cosmos3LatentPreparationStage，不影响其他模型或管线。性能提升显著：I2V 潜变量准备阶段延迟降低 70.9%，端到端延迟降低约 5.1%。内存消耗也相应减少，因为不再分配完整的视频张量。
- 风险标记：低风险

关联脉络

- PR #27041 [diffusion] Optimize Cosmos3 lossless hot paths: 同为 Cosmos3 优化系列，对同一文件进行过重构，关注性能提升。
- PR #27026 [diffusion] Add realtime WebUI super resolution controls: 均为 diffusion 管线改进，体现了项目持续优化 diffusion 功能的趋势。