

PR #47099 完整报告

vllm-project/vllm

[Bugfix] Align OpenCV video metadata timeline

合并时间: 2026-07-01 04:43

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/47099>

执行摘要

- 一句话: 对齐 OpenCV 视频元数据时间线
- 推荐动作: 建议所有使用 OpenCV 视频后端的团队成员精读此 PR, 理解元数据时间线与实际采样逻辑对齐的重要性。特别关注 `_sample_frame_indices` 的提取模式, 可作为后续相似 bug 修复的参考。

功能与动机

修复 Issue #46988: Gemma4 视频提示扩展中的时间戳与 Transformers 官方 `Gemma4Processor` 不一致。根源是 `video_get_metadata()` 报告的 `fps` 为 `duration / num_frames` (采样后的平均帧率), 而 `frames_indices` 为 `list(range(num_frames))` (采样后的位置), 与 `video_to_ndarrays()` 通过 `np.linspace(0, total_frames - 1, num_frames, dtype=int)` 实际采样的源视频帧索引不匹配。

实现拆解

1. 提取共享采样函数: 在 `vllm/assets/video.py` 中新增 `_sample_frame_indices(total_frames: int, num_frames: int)` 函数, 基于 `np.linspace` 返回均匀分布的帧索引数组。
2. 统一调用点: 将 `video_to_ndarrays()` 中内联的 `np.linspace` 替换为调用 `_sample_frame_indices`, 确保帧加载和元数据使用完全相同的采样逻辑。
3. 修正元数据字段: 在 `video_get_metadata()` 中增加对 `_sample_frame_indices` 的调用, 并将 `fps` 从 `duration / num_frames` (错误) 改为直接使用原始 OpenCV 读取的 `fps` (`cap.get(cv2.CAP_PROP_FPS)`), `frames_indices` 从 `list(range(num_frames))` 改为 `frame_indices.tolist()`。
4. 添加回归测试: 在 `tests/multimodal/media/test_video.py` 中新增 `test_opencv_video_metadata_matches_sampled_frame_timeline`, 创建一个 10 帧、5 FPS 的视频, 请求 4 个采样帧, 验证元数据中 `fps` 为 5.0、`duration` 为 2.0、`frames_indices` 为 `[0, 3, 6, 9]`、`total_num_frames` 为 4。

关键文件:

- `vllm/assets/video.py` (模块 视频资产; 类别 `source`; 类型 `core-logic`; 符号 `_sample_frame_indices`): 核心变更文件, 提取了共享采样函数并修正了视频元数据字段。

- tests/multimodal/media/test_video.py (模块 视频测试; 类别 test; 类型 test-coverage; 符号 test_opencv_video_metadata_matches_sampled_frame_timeline) : 新增回归测试, 确保元数据与采样逻辑一致, 避免后续回归。

关键符号: _sample_frame_indices, video_get_metadata

关键源码片段

vllm/assets/video.py

核心变更文件, 提取了共享采样函数并修正了视频元数据字段。

```
def _sample_frame_indices(total_frames: int, num_frames: int) -> npt.NDArray:
    # 返回均匀分布在 [0, total_frames-1] 区间内的 num_frames 个帧索引
    # 使用 int 向下取整, 保证与 video_to_ndarrays 中的采样行为完全一致
    return np.linspace(0, total_frames - 1, num_frames, dtype=int)

def video_get_metadata(path: str, num_frames: int = -1) -> dict[str, Any]:
    import cv2

    cap = cv2.VideoCapture(path)
    if not cap.isOpened():
        raise ValueError(f"Could not open video file {path}")

    total_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
    fps = cap.get(cv2.CAP_PROP_FPS) # 保持原始视频 FPS
    duration = total_frames / fps if fps > 0 else 0

    if num_frames == -1 or num_frames > total_frames:
        num_frames = total_frames
    # 使用共享采样函数获取实际的帧索引, 而非 range(num_frames)
    frame_indices = _sample_frame_indices(total_frames, num_frames)

    metadata = {
        "total_num_frames": num_frames,
        "fps": fps, # 此前错误地为 duration / num_frames
        "duration": duration,
        "video_backend": "opencv",
        "frames_indices": frame_indices.tolist(), # 此前错误地为 list(range(num_frames))
        "do_sample_frames": num_frames == total_frames,
    }
    return metadata
```

tests/multimodal/media/test_video.py

新增回归测试, 确保元数据与采样逻辑一致, 避免后续回归。

```
def test_opencv_video_metadata_matches_sampled_frame_timeline(tmp_path):
    # 构造一个 10 帧、5 FPS 的测试视频
    image_path = f"{tmp_path}/test_metadata_image.png"
```

```
Image.new("RGB", (8, 8), color=(255, 0, 0)).save(image_path)
video_path = f"{tmp_path}/test_metadata_video.mp4"
create_video_from_image(image_path, video_path, num_frames=10, fps=5.0)

# 请求 4 个采样帧
metadata = video_get_metadata(video_path, num_frames=4)

# 验证元数据：原始 FPS 应保持 5.0，时长 2.0 秒
assert metadata["fps"] == pytest.approx(5.0)
assert metadata["duration"] == pytest.approx(2.0)
# 帧索引应为均匀分布的 [0, 3, 6, 9]，而非 [0, 1, 2, 3]
assert metadata["frames_indices"] == [0, 3, 6, 9]
assert metadata["total_num_frames"] == 4
```

评论区精华

无 review 评论。Isotr0py 批准了 PR，仅评论 "Thanks for catching this!"。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。变更集中在 vllm/assets/video.py 中两个函数，提取出的 `_sample_frame_indices` 逻辑与原有的 `video_to_ndarrays` 内联 `np.linspace` 调用完全一致；`video_get_metadata` 的 FPS 改为使用原始 fps，更准确且不向下游引入新错误。测试覆盖了关键场景。
- 影响：直接影响 `video_get_metadata()` 返回的 `fps` 和 `frames_indices` 字段，从而修复了 Gemma4 等依赖帧时间戳的多模态模型的提示扩展准确性。对未采样的全帧视频无影响。
- 风险标记：低风险

关联脉络

- PR #36396 fix: release VideoCapture resources and guard div-by-zero in video utils: 关联 Issue #46988 提及的同一文件修改，但该 PR 关注资源释放而非元数据时间线对齐。
- PR #40764 [CI/Build] Make opencv-python-headless an optional dependency: 相关的 Opencv 依赖变更，但与本 PR 功能无关。