Code Attribution Report

1. Percentage of Code Copied from the Internet

The following portions of the code were copied or adapted from official documentation:

- **SAM2**: The SAM2 model integration code is copied from the **official SAM2 documentation** and implementation examples.
- **YOLOv8**: Object detection logic is based on code examples provided in the **official YOLOv8 documentation**.

Percentage of Code Copied: Approximately 10%.

2. Percentage of Group Contributions (Not Copied from the Internet)

The remaining 90% of the code comprises our original contributions, which include:

Load Libraries and Models:

- Import necessary libraries (`cv2`, `YOLO`, `SAM2`, `supervision`, etc.).
- Load pre-trained models: `SAM2VideoPredictor` (for segmentation) and YOLO (for object detection).

Extract Video Information:

- Use `sv.VideoInfo.from_video_path(SOURCE_VIDEO)` to get video metadata (e.g., FPS, total frames).
- Set a scale factor to resize frames for processing.

Extract Frames from Video:

- Create a directory (`SOURCE_FRAMES`) to store extracted frames.
- Extract video frames between `START_IDX` and `END_IDX` (up to 600 frames).
- Resize each frame by the scale factor and save it as `.jpeg`.

Detect Objects with YOLO:

- Load the YOLO model (`yolov8n.pt`).
- Perform object detection on each frame and extract detected objects.

- Assign unique labels to each detected object (e.g., `person_1`, `car_2`).

Annotate Objects in Frames:

- Use `BBoxWidget` to annotate objects with bounding boxes (predefined or user-drawn).
- Extract bounding box coordinates for each detected object.
- Convert these coordinates into points and generate masks using the SAM2 model.

Store Annotated Frames:

- Create directories (`ORIGINAL_FRAMES_DIR` and `ANNOTATED_FRAMES_DIR`) to save frames.
- Annotate frames by adding detected object masks.
- Save both original frames (without annotation) and annotated frames.

Create Final Annotated Video:

- Collect all annotated frames from `ANNOTATED_FRAMES_DIR`.
- Combine them into a single video file (`final_annotated_video.mp4`) using OpenCV.
- Set the resolution and frame rate of the final video based on the extracted frames.

Output:

- Save the final annotated video with all frames compiled into it.

Detailed Explanation of Group Contributions:

Our team developed custom logic to:

- Integrate YOLOv8 and SAM2 predictions in a seamless workflow.
- Build a user-friendly interface using Streamlit, enabling easy interaction with the application.
- Implement video processing mechanisms (e.g., frame extraction, resizing, object annotations).
- Add the functionality for selecting specific objects to summarize in the video.

3. URLs of the Copied Elements

Below are the URLs of the resources from which code was referenced:

- **SAM2 Official Documentation**: https://github.com/facebookresearch/sam2
- **YOLOv8 Official Documentation**: https://github.com/ultralytics/ultralytics