If commenting out the OCR processing allows the frames to stream correctly, then it's likely that the OCR processing is either too slow or causing an error that prevents the frames from being encoded and sent properly.

Here are some potential solutions and optimizations you could consider:

1. **Asynchronous Processing:** OCR processing can be computationally expensive, especially if done on every frame of a video in real-time. Consider offloading the OCR work to a separate thread or process to avoid blocking the main thread that's streaming the frames. This way, you can still display the video while the OCR is being processed in the background.
2. **Frame Skipping:** You don't necessarily need to run OCR on every single frame, as video frames often have little difference between them over short periods of time. You could choose to run OCR on every nth frame or based on a certain time interval.
3. **Lower Resolution:** Before applying OCR, you can reduce the resolution of the frames. OCR often doesn't require high-resolution images to work effectively, and processing smaller images will be faster.
4. **Optimize pytesseract:** If you're using default settings for Tesseract, there might be some optimizations that can be applied, such as limiting the OCR to numbers only or specific character sets, depending on your use case.
5. **Profiling:** Use profiling tools to understand where the bottleneck is in your OCR processing. This information can help you to optimize the code more effectively.
6. **Caching Results:** If the video has static sections where text doesn't change much, you can cache the OCR results for certain frames and reuse them, rather than reprocessing the same image.
7. **Hardware Acceleration:** If you have access to a GPU, you can use it to accelerate the OCR processing. Some OCR tools can leverage GPU acceleration, which might require additional setup and configuration.
8. **Error Handling:** Ensure that you have proper error handling around the OCR code. Any exception thrown by **pytesseract** should be caught so that it doesn't interrupt the frame streaming.
9. **External OCR Services:** If local processing is too slow, you could consider using an external OCR service that may offer faster processing through optimized cloud-based solutions. This would involve sending frames to the service and receiving the OCR results.

Implementing these changes should help you find a balance between real-time video streaming and the computational demands of OCR processing.