Overall Approach

Problem Statement: Automate the counting of sheet stacks in a manufacturing plant to replace the current manual and error-prone process.

Approach:

1. **Image Upload**: Allow users to upload images of sheet stacks via a user-friendly web interface.
2. **Image Processing**: Utilize computer vision techniques to preprocess the image, detect edges, and identify contours corresponding to sheet stacks.
3. **Counting Sheets**: Count the number of detected contours, which represent individual sheets.
4. **Output Results**: Display the number of sheets and provide an annotated image highlighting the detected contours.

Frameworks/Libraries/Tools

1. **Streamlit**: For creating the web interface, handling file uploads, and displaying results.
2. **OpenCV**: For image processing, including converting images to grayscale, applying Gaussian blur, detecting edges, and finding contours.
3. **NumPy**: For efficient handling of image data as arrays.
4. **PIL (Pillow)**: For reading the uploaded image files.
5. **Matplotlib**: For plotting and visualizing images (imported but not utilized in the current version).

Challenges and Solutions

1. **Challenge**: Handling diverse image qualities and formats.
    - **Solution**: Convert images to a consistent grayscale format and apply Gaussian blur to reduce noise, ensuring more robust edge detection.
2. **Challenge**: Accurate edge detection and contour identification.
    - **Solution**: Use Canny edge detection, which is effective for detecting a wide range of edges, followed by finding external contours to isolate individual sheets.
3. **Challenge**: Displaying processed images in a user-friendly manner.
    - **Solution**: Convert the processed image from BGR to RGB format (since OpenCV uses BGR by default) before displaying it with Streamlit.

Future Scope

1. **Enhanced Image Processing**: Implement advanced image processing techniques like adaptive thresholding or deep learning-based methods for more accurate sheet detection, especially in complex scenarios.

2. **Batch Processing**: Allow users to upload and process multiple images simultaneously, improving efficiency for larger datasets.
3. **Integration with Manufacturing Systems**: Connect the application to existing manufacturing systems for real-time data integration and automated workflow.
4. **Mobile Application**: Develop a mobile app version to enable on-the-go image capturing and processing directly from smartphones.
5. **User Feedback Mechanism**: Incorporate a feedback system where users can provide input on the accuracy of the detected sheet count, enabling continuous improvement through machine learning.