Countify Project Documentation

Overall Approach:

The main approach to solving the problem involved using a Convolutional Neural Network (CNN) to detect and count the number of sheets in a stack from provided images. The CNN model was chosen due to its ability to handle the complexities of image recognition and pattern detection. The preprocessing stage included techniques like edge detection and contrast adjustment to improve model accuracy.

Frameworks/Libraries/Tools:

- **TensorFlow/Keras**: Used for defining and training the CNN model.
- **Flask**: Used to build the web application for user interaction.
- **OpenCV**: **Employed** for image preprocessing tasks such as edge detection.
- **NumPy**: Utilised for numerical operations and data handling.
- **Pillow**: Used for image manipulation and processing.

Challenges and Solutions:

- **Challenge:** Handling varying lighting conditions in images.
  - **Solution:** Applied histogram equalisation during preprocessing to normalise image contrast.
- **Challenge:** Accurately counting sheets with overlapping edges.
  - **Solution:** Implemented edge detection combined with deep learning to differentiate closely stacked sheets.
- **Challenge:** Ensuring fast processing times for real-time usability.
  - **Solution:** Optimised the CNN model by reducing its complexity and using efficient preprocessing techniques.

Future Scope:

- **Improvement in Counting Accuracy:** Implementing more sophisticated deep learning models like ResNet or DenseNet could improve the counting accuracy.
- **Real-time Video Analysis:** Expanding the application to support real-time video analysis for continuous monitoring of sheet stacks.
- **Cross-platform Application:** Developing a mobile version of the application to enable on-site use in the manufacturing plant.

This document serves as the project's comprehensive guide, detailing the approach, tools, challenges, and future potential.