Problem Statement:
In industries such as e-commerce, healthcare, and supply chain management, product information is often stored in images, making it difficult to extract important numeric values and units for structured data processing. Manually extracting this information is time-consuming and prone to errors.
Solution:
This project provides an automated solution to extract numeric values with corresponding units from images, filter the extracted values based on the context (entity types like voltage, weight, etc.), and store them for further analysis. The project leverages OCR (Optical Character Recognition) technology, specifically EasyOCR, to handle text extraction from images, and uses a predefined map of allowed units for different entities to filter the relevant information.

Technologies Used

- **Programming Language**: Python
- **Libraries**:
    - **EasyOCR**: For extracting text from images.
    - **OpenCV**: For image processing and loading.
    - **Pandas**: For data handling (input and output in CSV format).
    - **Requests**: For downloading images from URLs.
    - **TQDM**: For progress bar during processing.
    - **Regex (re)**: For extracting numeric values with specific units.

Approach

1. **Dataset Loading and Preparation**: The dataset is loaded using Pandas, and images are downloaded from specified URLs to a local folder if they don't already exist.
2. **Image Downloading**: Images are fetched using the requests library, and checks are made to avoid redundant downloads.
3. **Text Extraction**: EasyOCR is used to extract text from the downloaded images, supporting multiple languages and providing efficient extraction.
4. **Value and Unit Filtering**: A regular expression is applied to the extracted text to identify numeric values followed by allowed units based on a predefined entity-unit map.
5. **Entity-Based Filtering**: The program filters the extracted values based on the entity_name from the dataset, matching only the relevant units for each entity.
6. **Output Predictions**: The filtered values and their corresponding entity names are saved to a CSV, with each row containing the index and extracted numeric values matched to the entity.

Entity-Unit Mapping

A detailed map is used to identify valid units for different entities. For example:

- **Voltage**: {'V', 'kV', 'mV', 'volt', 'kilovolt', 'millivolt'}

- **Width/Height**: {‘cm’, ‘mm’, ‘m’, ‘inch’, etc.}

Code Flow

- **Main Components**:
  - **Download Function**: Downloads images from the given URLs.
  - **Text Extraction Function**: Uses EasyOCR to extract text from the images.
  - **Filtering Function**: Uses regex to extract and filter numeric values with valid units for the specified entities.
  - **Data Handling**: Pandas is used to read the input CSV and write the output CSV.

Example Walkthrough

Input Example:

- Image: An image containing the text “87inch, 24cm, 5V”
- Entity: voltage

Output:

- The program will extract the value “5V” from the image, as it matches the voltage entity’s valid units.

Challenges Faced

- **OCR Accuracy**: Text extraction is highly dependent on image quality and text alignment. EasyOCR performs well in most cases, but improvements in image preprocessing could increase accuracy.
- **Contextual Filtering**: Mapping text to the correct entity required precise unit definitions to ensure that only relevant values were extracted.

Future Enhancements

- **Improved Image Preprocessing**: Implement additional steps like image resizing, contrast enhancement, or noise reduction to improve OCR accuracy.
- **Support for Multiple Languages**: Expand the language capabilities beyond English for more versatile usage.

Conclusion

This project automates the extraction of structured numeric data from images, a task that is often tedious and prone to human error. By leveraging OCR technology, regular expressions, and entity-unit mapping, the project delivers a solution that can significantly enhance data processing workflows across multiple industries.