AI Based Product Sorting System

Ms.Nishigandha N.Shevkar, Rohit Dhage , Chetan shewale , Kartik Nikam, Yash Kapase

1. Ms.Nishigandha N.Shevkar, lecturer, Computer Engineering, Mahavir Polytechnic, Nashik
2.Mr. Rohit Dhage, Student, Information technology, Mahavir Polytechnic, Nashik
3.Mr. Chetan shewale , Student, Information technology, Mahavir Polytechnic, Nashik
4.Mr. kartik Nikam, Student, Information technology, Mahavir Polytechnic, Nashik
5.Mr.Yash Kapase, Student , Information technology, Mahavir Polytechnic, Nashik

---

Abstract – This project presents a deep learning-based image processing system for product sorting applications. A real-time camera captures images of various products, including candies, chocolates, chairs, and mobile phones. The system processes these images using deep learning algorithms to detect and identify components within the frame. Each detected component is labeled with its name, and bounding boxes are drawn to highlight their positions. A user-friendly web interface displays the real-time camera feed along with detection results, ensuring an interactive and informative experience. The system's architecture prioritizes accuracy and efficiency, optimizing detection reliability across diverse product categories

Key Words: Deep Learning, Image Processing ,Object Detection ,Real-Time Camera ,Feed Bounding, Box Annotation Neural Networks, Product Classification ,Web Interface

1. INTRODUCTION

In the modern era of automation and artificial intelligence, industries are increasingly leveraging deep learning and computer vision technologies to enhance efficiency and accuracy in various applications. One such application is automated product sorting, which plays a critical role in manufacturing, retail, logistics, and quality control. This project aims to design a deep learning-based image processing system for product sorting, focusing on real-time detection and classification of products such as candies, chocolates, chairs, and mobile phones.

The system will utilize a camera to capture live images, process them using advanced deep learning algorithms, and identify the components within the frame. Each detected object will be labeled with its name, and a bounding box will be drawn around it to highlight its position. The integration of such an automated solution ensures streamlined operations, reduced manual labor, and enhanced accuracy.

Traditional product sorting techniques often rely on manual labor or rule-based algorithms, which may not be efficient in handling a large variety of objects with different shapes, sizes, and colors. Manual sorting is prone to human errors, fatigue, and inefficiency, whereas rule-based image processing systems may struggle with variations in lighting, background, and object orientation.

To overcome these limitations, deep learning-based image processing offers a robust and scalable solution capable of learning complex patterns from vast datasets, making real-time product sorting highly accurate and reliable.

2.Objectives

Here are the objectives in concise, clear English:

1. **Real-Time Image Capture** – Use a camera to capture images of products dynamically.
2. **Deep Learning-Based Detection** – Implement deep learning algorithms to identify and classify products.
3. **Component Identification** – Label detected products with their names and highlight them with bounding boxes.
4. **User-Friendly Web Interface** – Display the real-time camera feed and detection results interactively.

5. **High Accuracy & Efficiency** – Optimize the system for reliable and fast detection across multiple product categories.

2. System Components and Functionality

2.1 Camera Module

The camera module serves as the primary input device for capturing real-time images of products for the sorting system. It is responsible for continuously streaming high-resolution images, ensuring clarity and accuracy in detecting various objects, including candies, chocolates, chairs, and mobile phones. The camera must be capable of capturing images with minimal distortion and sufficient lighting to enhance detection performance

To achieve real-time processing, the camera should have a high frame rate and low latency. A suitable choice would be an industrial-grade camera or a high-resolution USB/web camera with autofocus and adjustable exposure settings. The module must also support seamless integration with the deep learning model and web interface, allowing instant image transmission and processing.

The camera module interfaces with the processing unit via USB, Ethernet, or wireless connectivity, depending on the system's design. It continuously captures images and transmits them to the deep learning model for analysis. The system may also implement pre-processing techniques, such as noise reduction, contrast enhancement, and background removal, to improve detection accuracy.

Overall, the camera module plays a crucial role in ensuring the efficiency and reliability of the product sorting system by providing high-quality image data for deep learning-based classification and object detection.

2.2 Image Processing & Preprocessing

The image processing and preprocessing stage is crucial in ensuring accurate and efficient product detection within the deep learning-based sorting system. The process begins with capturing real-time images using a high-resolution camera. These images may contain various products, such as candies, chocolates, chairs, and mobile phones, positioned at different angles and lighting conditions. To enhance detection accuracy, the images undergo several preprocessing techniques before being passed into the deep learning model.

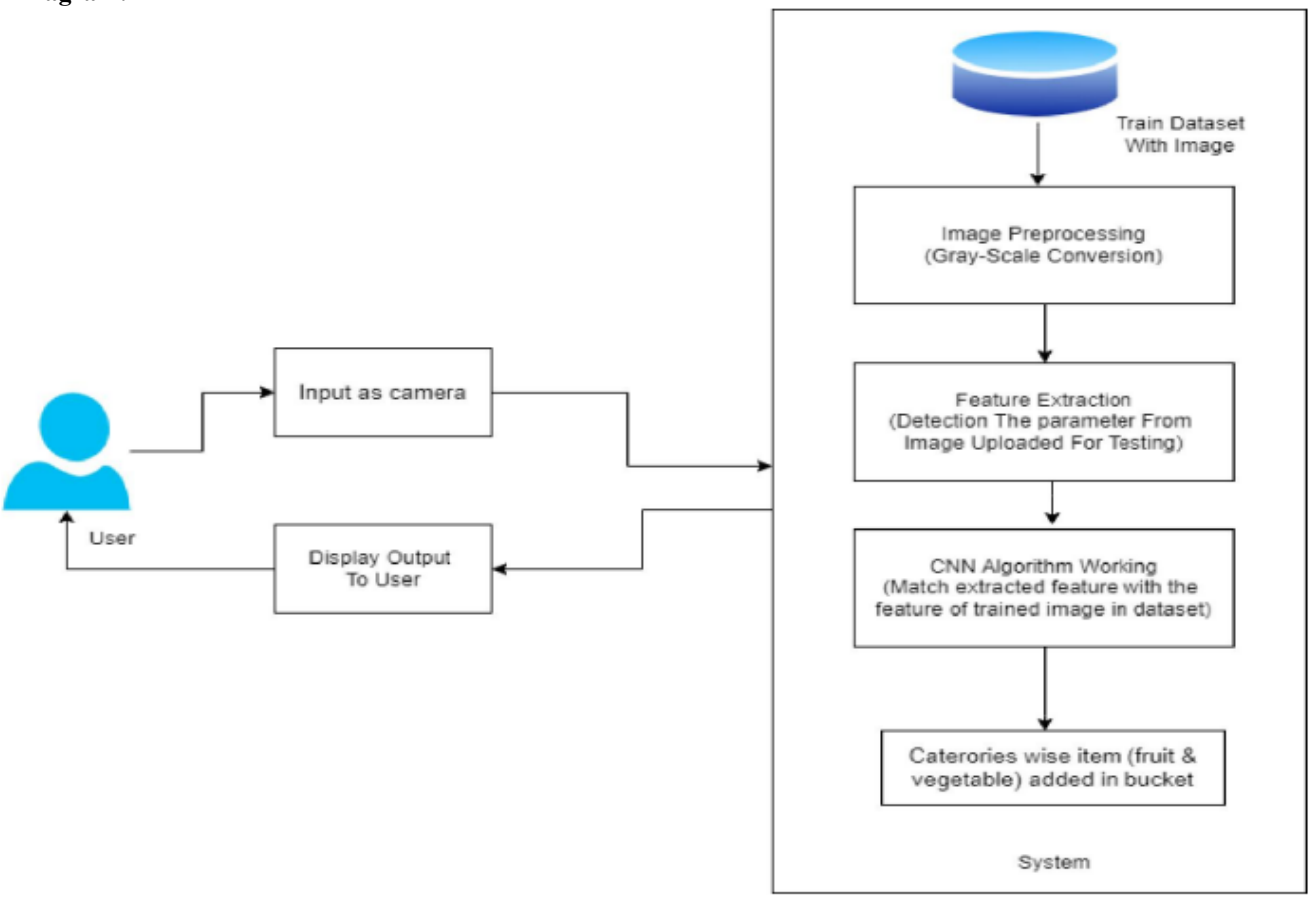
Initially, noise reduction techniques like Gaussian blur or median filtering are applied to remove unwanted distortions, ensuring a cleaner image for analysis. Next, image resizing is performed to standardize input dimensions, maintaining consistency across different product categories. This step helps optimize computational efficiency without compromising detection accuracy. Additionally, color normalization and contrast enhancement techniques, such as histogram equalization, improve visibility and feature extraction.

Segmentation is another vital step, where the system isolates the objects of interest from the background using techniques like thresholding, edge detection (Canny edge detection), or deep-learning-based instance segmentation models (e.g., Mask R-CNN). This ensures that only relevant product features are fed into the classification and detection model. Furthermore, data augmentation techniques, such as rotation, flipping, and scaling, are employed to enhance model robustness, allowing it to recognize products in various orientations and environments.

2.3 Deep Learning Model for Object Detection

The core of the product sorting application is a deep learning-based object detection model. This model is responsible for identifying and classifying products in real-time from the camera feed.:

DFD Diagram:



..

Algorithm:

- Data Preprocessing: Collection of product images, labeling, and augmentation.
- Model Training: Using CNN (Convolutional Neural Network) or YOLO (You Only Look Once) for real-time object detection.
- Classification: Products are classified based on predefined features such as shape, size, or labels.

2.4 Data Flow Diagram:

In Data Flow Diagram, we Show that flow of data in our system in DFD0 we show that base DFD in which rectangle present input as well as output and circle show our system, In DFD1 we show actual input and actual output of system input of our system is text or image and output is rumour detected likewise in DFD 2 we present operation of user as well as admin.

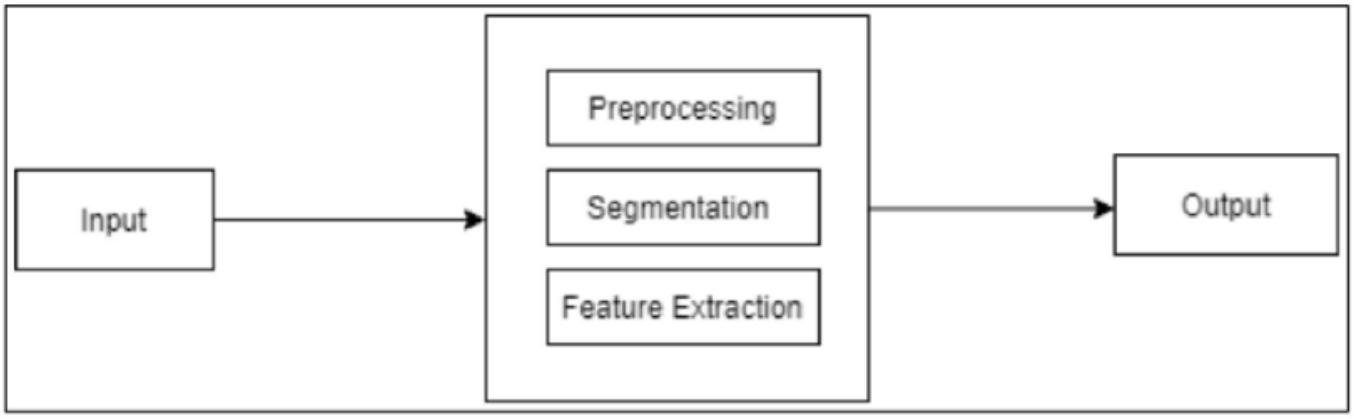


Fig. Data Flow Diagram

3. Advantages of the Proposed System

1. **Automated Product Sorting** – Reduces manual effort by classifying and labeling products in real-time.
2. **High Accuracy** – Utilizes deep learning algorithms for precise object detection and classification.
3. **Real-Time Processing** – Ensures quick and efficient identification of products using live camera feeds
4. **User-Friendly Interface** – Provides an interactive web interface displaying detection results clearly.
5. **Scalability** – Can be expanded to support additional product categories with minimal modifications.
6. **Efficiency** – Speeds up the sorting process, reducing human intervention and operational costs.

4. Applications

1. **Automated Product Sorting** – The system can be used in manufacturing and packaging industries to categorize and sort products efficiently.
2. **Retail and Inventory Management** – Stores and warehouses can leverage the system to automate inventory tracking by identifying products in real time.
3. **Quality Control** – Helps in detecting defective or misplaced products during production, ensuring quality assurance.
4. **E-commerce Warehousing** – Improves order fulfillment by recognizing and sorting items before shipment.
5. **Smart Vending Machines** – Enhances automated vending systems by identifying and dispensing the correct products.

5. Future Scope

1. Enhanced Product Categories – The system can be trained on a larger dataset to recognize a wider range of products beyond the initial categories.
2. Integration with Robotics – The system can be integrated with robotic arms for automated product sorting and placement.
3. Edge AI Deployment – Optimizing the model for edge devices can reduce latency and improve real-time processing without relying on cloud-based computation.
4. Multi-Camera Support – Incorporating multiple cameras can improve detection accuracy and provide a 360-degree view for better classification.
5. Self-Learning Mechanism – Implementing reinforcement learning can allow the system to improve its detection performance over time.
6. Cloud-Based Data Storage & Analytics – Storing detected product data in the cloud can help with trend analysis,

inventory management, and predictive maintenance.

7. Integration with IoT – Connecting with IoT-enabled devices can enable automated decision-making and real-time notifications for inventory tracking.
8. Mobile & Web Accessibility – Expanding the interface to mobile applications and web dashboards for remote monitoring and management.
9. Multi-Language Support – Adding support for different languages can enhance usability across global markets

6. CONCLUSION

The deep learning-based image processing system for product sorting effectively automates the identification and classification of various products, such as candies, chocolates, chairs, and mobile phones. By leveraging a real-time camera feed and advanced deep learning algorithms, the system accurately detects objects, labels them, and highlights their positions with bounding boxes.

The user-friendly web interface enhances interaction by displaying live detection results, ensuring clarity and ease of use. The system is designed for high accuracy, efficiency, and reliability, making it a valuable solution for automated product sorting applications.

.

ACKNOWLEDGEMENT

We sincerely thank everyone who supported and guided us during this project. First, we are very grateful to our mentor, **Ms.Nishigandha N.Shevkar**, for his valuable guidance, encouragement, and helpful feedback. His support was very important for our project.

We also thank our institution, **Mahavir Polytechnic, Nashik**, for providing the resources and a good learning environment. Special thanks to our professors and faculty members for their support and motivation.

We appreciate our friends and classmates for their helpful suggestions and moral support. Their ideas and discussions helped us improve our project.

We thank our families for their patience, encouragement, and belief in us. Their constant support helped us complete this project successfully.

Thank you all for your help and inspiration.

REFERENCES

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
2. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
3. Chollet, F. (2021). *Deep Learning with Python*. Manning Publications.
4. Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. arXiv preprint arXiv:1804.02767.
5. He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition*. IEEE CVPR.
6. Ren, S., He, K., Girshick, R., & Sun, J. (2015). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. NeurIPS