**Design and Implementation of an Autonomous Sorting System Using Mechatronics principles and AI**

**1.Problem statement:**

Develop an autonomous sorting system capable of identifying and sorting objects based on specific attributes (size, color, shape, etc.). The system should consist of the following components:

1. Object Detection: The system must utilize vision or proximity sensors (e.g., cameras, infrared sensors, or ultrasonic sensors) to detect and recognize objects on a conveyor belt or tray.
2. Control System: Develop an embedded system (e.g., Arduino, Raspberry Pi) to interface with sensors and actuators, making real-time decisions on sorting.
3. Sorting Mechanism: The mechanical part of the system should include actuators such as robotic arms, servos, or pneumatic systems to perform the sorting action.
4. AI Integration: Use AI/ML models to classify objects based on their features (shape, size, or color) and improve system performance over time by adapting to new objects.
5. Real-time Feedback and Adjustment: The system must be able to respond to dynamic changes, such as moving objects with varying speeds, or objects that are not within the original scope of the system.

**2.Abstract:**

Automated sorting systems play a crucial role in modern industries, enhancing efficiency and reducing human effort in material handling. This project presents the **design and implementation of an AI-powered robotic arm-based sorting system**, integrating **mechatronics principles and artificial intelligence (AI)** for intelligent object separation. The robotic arm utilizes **sensors (color, ultrasonic, inductive, and capacitive)** to classify objects based on **color, size, and material composition**, while an **AI-driven image processing model** enhances classification accuracy.

The system consists of a **servo motor-controlled robotic arm**, a **microcontroller (Arduino Uno/Mega) for real-time control**, and a **Raspberry Pi for AI-based object detection using machine learning (ML) models**. The robotic arm executes **predefined pick-and-place operations** based on sensor inputs and AI decisions, ensuring high-speed and accurate sorting.

**3.Introduction:**

In modern industries, automation plays a crucial role in enhancing efficiency, accuracy, and productivity. Sorting systems are widely used in manufacturing, logistics, and recycling industries to categorize objects based on specific attributes such as **size, color, and shape**. Traditional sorting methods rely on manual labor or simple mechanical processes, which can be inefficient and prone to errors. To address these challenges, this project presents an **AI-powered autonomous sorting system** that combines **mechatronics principles and artificial intelligence (AI)** for intelligent object identification and classification.

The system integrates **vision and proximity sensors** (cameras, infrared, ultrasonic) to detect objects on a conveyor belt or tray. A **microcontroller (Arduino) or microprocessor (Raspberry Pi)** processes the sensor data, while **AI-based object classification models** enhance decision-making. The robotic arm, controlled by actuators, picks and places objects into designated categories with high precision. Additionally, real-time feedback mechanisms allow the system to **adapt dynamically** to changing object conditions, such as varying speeds or unexpected variations.By leveraging **AI and automation**, this system significantly improves sorting efficiency, reduces human intervention, and ensures adaptability for future industrial applications.

**4.Key Components:**

|  |  |
| --- | --- |
| **Component** | **Purpose** |
| Robotic Arm | Picks and places objects |
| Microcontroller | Controls arm & sensors |
| Microprocessor (AI Processing) | Runs AI model |
| Sensors | Detects object properties |
| End Effector (Gripper) | Holds objects |
| Power Supply | Powers motors & electronics |

**5.Mechanical Design & Construction of the Robotic Arm:**

**Structural Design**

* **Material Selection: Use aluminum, acrylic, or 3D-printed plastic for lightweight durability.**
* **Joints and Degrees of Freedom (DOF):**
  + **Base Rotation (DOF 1) → Moves arm left & right.**
  + **Shoulder (DOF 2) → Lifts arm up & down.**
  + **Elbow (DOF 3) → Extends/retracts arm.**
  + **Wrist (DOF 4) → Rotates object orientation.**
  + **Gripper (DOF 5) → Opens/closes to grasp objects.**

**Assembly**

* **Use servo motors for controlled movement.**
* **Connect servos to Servo Driver (for multiple servo control).**
* **Mount sensors near the pickup zone for object detection.**

**6.Electrical & Sensor Integration:**

**Sensors for Object Classification**

The sorting system relies on multiple sensors to classify objects before the robotic arm moves them to their respective bins.

|  |  |
| --- | --- |
| **Sensor** | **Function** |
| Color Sensor | Detects object color |
| Ultrasonic Sensor | Measures object size |
| Inductive Sensor | Detects metallic objects |
| Capacitive Sensor | Detects non-metallic objects |
| Camera | AI-based object recognition |

**Actuators for Robotic Arm Control**

* **Servo Motors** → Controls base, shoulder, elbow, wrist, and gripper.
* **Stepper Motors** → Used for **precise movements** (if needed).

**7.Circuit Connections**

**Microcontroller**

* Controls **servo motors**.
* Reads **sensor data**.
* Sends classification data to **AI processor**

**Microprocessor**

* Runs AI-based object classification.
* Communicates sorting decisions to Arduino via **I2C or Serial Communication.**

**8.Software Development & Control Logic:**

**Sorting Algorithm Implementation**

1. **Read Sensor Data**
   1. Identify object **color, size, and material**.
   2. If AI is enabled, capture an image and classify using ML model.
2. **Decision Making**

Identifying the object and drop in respective bins.

1. **Move Robotic Arm to Pick Position**
   1. Calculate required **servo angles** for movement.
   2. Move arm to **pickup position**.
   3. Close gripper to **grasp object**.
2. **Move Object to Sorting Bin**
   1. Rotate base to correct bin location.
   2. Release object in correct bin.

**9.AI-Based Object Classification:**

**AI Model (Deep Learning)**

* **Train a Convolutional Neural Network (CNN)** for object recognition.
* **Pre-train models** with images of different objects.

**AI Classification Workflow**

1. **Capture Object Image** using **Camera**.
2. **Preprocess Image** (resize, grayscale conversion).
3. **Pass Image to Trained CNN Model** for classification.
4. **Send Sorting Decision to Arduino**.