---------- Requirements Specification Portion ----------

5.0 Requirements Specification (SRS) - Vision Guided Robotic Arm **1**
5.1 Introduction 1
5.2 CSCI Component Breakdown 2
- 5.2.1 Vision - Webcam capture, processing, detection/classification 2
- 5.2.2 Motion & Control 2
- 5.2.3 Logic 2
- 5.2.4 User Interface 2
- 5.2.5 Data 2
5.3 Functional Requirements by CSC 2
- 5.3.1 Vision 2
- 5.3.2 Hardware 2
- 5.3.3 Trajectory 3
- 5.3.4 Logic 3
- 5.3.5 User Interaction 3
5.4 Performance Requirements by CSC 3
- 5.4.1 Response Time 3
- 5.4.2 Pick and Place Time 3
- 5.4.3 Camera Performance 3
- 5.4.4 Accuracy 3
- 5.4.5 Safety 4
5.5 Project Environment Requirements 4
5.5.1 Development Environment 4
5.5.2 Execution 4
● 5.5.2.1 Hardware: 4

5.0 Requirements Specification (SRS) - Vision Guided Robotic Arm

5.1 Introduction

Description: We're creating a 3D-printed, servo-driven robotic arm powered by a Raspberry PI that uses a combination of a webcam and computer vision to detect and manipulate small objects. The system will ideally implement a pick-and-place movement action. Our arm is going to emphasize a CS-focused backend (Python + OpenCV + NumPy IK) with TensorFlow Lite. Users are going to be students, hobbyists, researchers, etc. Maintenance primarily consists of re-printing parts, replacing servos, and updating Python dependencies.

5.2 CSCI Component Breakdown

- 5.2.1 Vision - Webcam capture, processing, detection/classification
    - 5.2.1.1 Camera
    - 5.2.1.2 Detection Software

- 5.2.2 Motion & Control
    - 5.2.2.1 Trajectory Planner
    - 5.2.2.2 Servo Driver
    - 5.2.2.3 Safety Mechanisms (overheat protection, etc)

- 5.2.3 Logic
    - 5.2.3.1 Pick and Place
    - 5.2.3.2 Calibration (servos, ect)

- 5.2.4 User Interface
    - 5.2.4.1 CLI (start/stop, status, etc)

- 5.2.5 Data
    - Dataset (model training)

5.3 Functional Requirements

- 5.3.1 Vision
    - 5.3.1.1 The system shall acquire frames from a USB webcam at >= 15 FPS
    - 5.3.1.2 The system shall detect at least one simple object (i.e. ball, book, etc)
    - 5.3.1.3 The system shall provide the position of the detected object in the frame

- 5.3.2 Hardware
    - 5.3.2.1 The arm shall be able to move each joint when sent a basic command
    - 5.3.2.2 The system shall provide a way to stop all movement quickly (i.e. emergency stop)
    - 5.3.2.3 The system shall allow us to set a safe range of motion for each joint

- 5.3.3 Trajectory
    - 5.3.3.1 The system shall let us move the arm to a specific position
    - 5.3.3.2 The system shall move smoothly from one position to another without it being jumpy
    - 5.3.3.3 The system shall support a “rest” position command

- 5.3.4 Logic
    - 5.3.4.1 The system shall support a simple “pick and place” motion: detect -> move arm -> close hand -> move -> release
    - 5.3.4.2 The system shall allow switching between a manual mode (we control the arm with commands) and an automatic mode (self autonomy)

- 5.3.5 User Interaction
    - 5.3.5.1 The system shall provide a basic way to start and stop tasks from the terminal
    - 5.3.5.2 The system shall print simple status messages (e.g. “object found”, “no object found”, etc)

5.4 Performance Requirements

- 5.4.1 Response Time
    - 5.4.1.1 The system shall start moving the arm within a couple of seconds (1-4 seconds) after detecting an object.

- 5.4.2 Pick and Place Time
    - 5.4.2.1 The system shall be able to finish a simple pick and place motion in about 15 seconds or less.

- 5.4.3 Camera Performance
    - 5.4.3.1 The system shall show camera images at a reasonable speed (10 fps or more) so that the arm can react

- 5.4.4 Accuracy
    - 5.4.4.1 The arm shall get close enough to the object to grab it (within about 1-4 cm of the target)

- 5.4.5 Safety
    - 5.4.5.1 When the stop command is executed, the arm shall stop moving right away

5.5 Project Environment Requirements

5.5.1 Development Environment

- 5.5.1.1 Operating System: Raspberry Pi OS 64-bit
- 5.5.1.2 Language: Python 3.12 (possibly more)
- 5.5.1.3 Libraries: OpenCV, NumPy
- 5.5.1.4 Tools: Github

5.5.2 Execution

- 5.5.2.1 Hardware
    - 5.5.2.1.1 Raspberry Pi 4
    - 5.5.2.1.2 USB Webcam (preferably 1080p)
    - 5.5.2.1.3 6x STS3215 Servos
    - 5.5.2.1.4 Motor Control Board
    - 5.5.2.1.4 Power Supply (5V)
    - 5.5.2.1.5 3D printed arm