Department of Computer Science & Engineering
The University of Texas at Arlington

Architectural Design Specification
CSE 4317: Senior Design II
Spring 2022

H.I.L Robot Team
H.A.F.R.A

Resha Adhikari
Katia Lopez
Nishan Pathak
Cesar Rea
Richard Tran

Revision History

Revision	Date	Author(s)	Description
0.1	11.11.2021	RA, KL, NP, CR, RT	document creation
1.0	11.15.2021	RA, KL, NP, CR, RT	official release
2.0	5.11.2022	RA, KL, NP, CR, RT	final release

CONTENTS

List of Figures

List of Tables

1 INTRODUCTION

HAFRA is an application that uses the UR5 robot and an Intel RealSense camera to detect ArUco markers and pick up envelopes and place them into a bin. The UR5 robot will be able to pick up an envelope after identifying the fiducial marker. With the help of the camera installed on the robot, and the assistance of computer vision system it will be able to identify the marker. After identifying the marker, the robot's arm will create the movement towards that envelope and pick it up with the suction gripper attached at the robot arm.

2 System Overview

The Human Assistance For Robot Arm (H.A.F.R.A) project has three major architectural layers which are the UR5 robot, the Movement Program and the Vision Program. There will be two programs (Movement and Vision) that will communicate with one another and send instructions to the robot to perform its task. The current task is designated as picking up envelopes and placing them into a bin using an AruCo tag. The robotâs camera will be used to detect the AruCo tag. In addition, the vision program will have human assistance so that if the robot detects something unique or a problem it does not have a solution to, it will notify a human. The user will then be able to access the robot, enter pixel values based on an image of the environment, and ultimately fix the problem. This section contains the high-level block diagram of the layers, as shown in the figure below, as well as detailed descriptions of the functions of each layer.

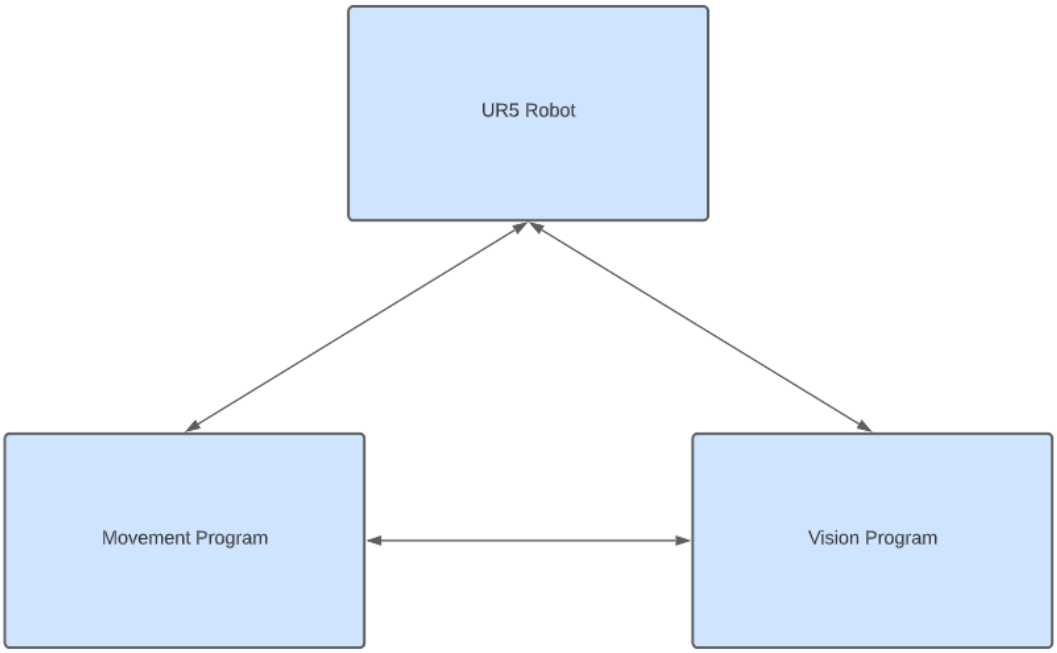

Figure 1: System Overview Diagram

2.1 UR5 Robot Layer Description

The UR5 robot layer will be interacting with both the Movement Program layer and Vision Program layer. The UR5 robot will receive information from the programs including movement instructions and picking instructions. This layer also contains a camera which will be used by the Vision Program layer.

2.2 Movement Program Layer Description

The Movement Program layer will be interacting with both the UR5 Robot layer and the Vision Program layer. This layer will receive world coordinates from the Vision Program AruCo tag detector subsystem

to locate the envelope. After receiving the information, picking instructions will be sent to the robot's suction gripper and movement instructions will be sent to the robot arm.

2.3 Vision Program Layer Description

The Vision Program layer will be interacting with both the UR5 Robot layer and the Movement Program layer. After taking an image from the camera, the World Coordinate subsystem will convert pixel coordinates into world coordinates and send this information to the Movement Program Layer. If the camera is not able to read an AruCo tag, it will send an image back to the vision program awaiting instruction from the user.

3 SUBSYSTEM DEFINITIONS & DATA FLOW

This section breaks down our layer abstraction to another level of detail. The graphical representation of the logical subsystems that compose each layer and the interactions/interfaces between those subsystems is shown in Figure 2. A subsystem is a programming unit that implements one of the major functions of the layer. Each subsystem, therefore, has data elements that serve as source/sinks for other subsystems. the UR5 robot has three subsystems which are the Camera, the Robot Arm, and the Suction Gripper. The Movement Program layer has two subsystems which is Picking and Position Instructions. Lastly, the Vision Program layer has three subsystems which are AruCo Detection, World Coordinates, and Human Assistance.

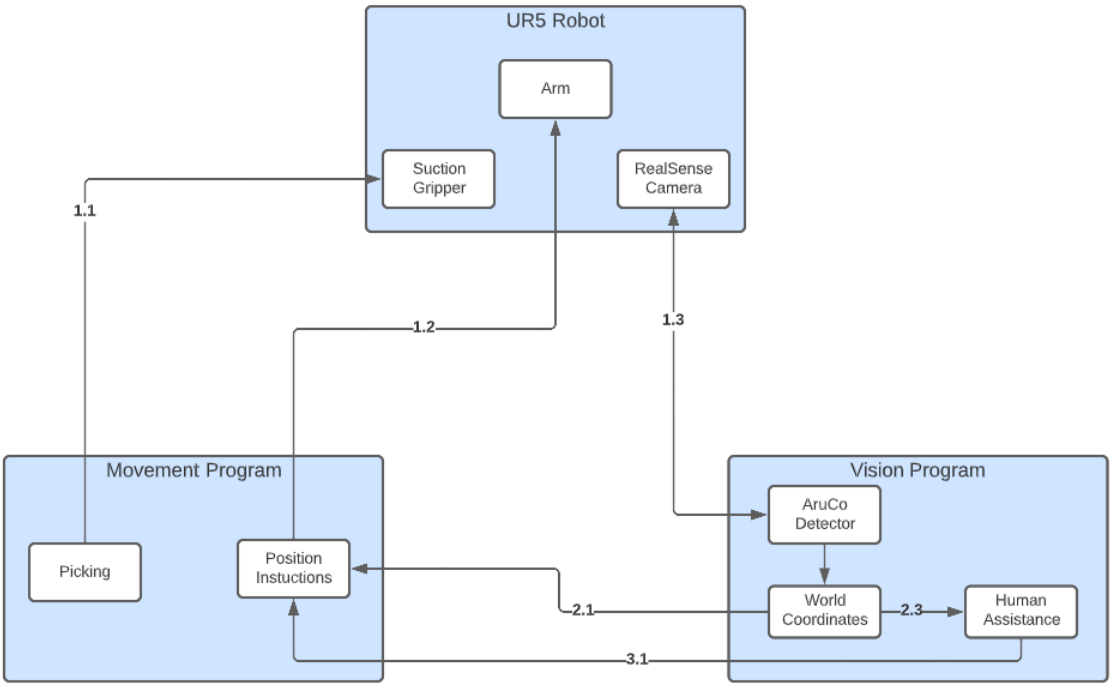

Figure 2: Data Flow Chart

4 UR5 Robot Layer Subsystems

This layer contains the hardware of the overall system. This layer directly responds to the Movement Program Layer and Vision Program Layer.

4.1 Arm System

The Arm subsystem is the robotic arm of the system. This subsystem responds to movements inputted by the program.

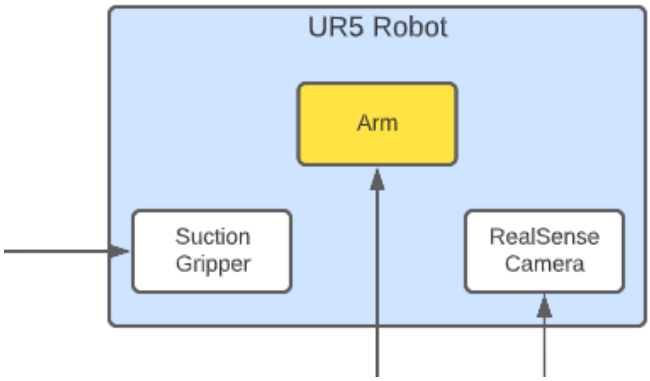

Figure 3: Arm Subsystem Description Diagram

4.1.1 Assumptions

Arm subsystem assumes that instructions sent are reliable and realizable.

4.1.2 Responsibilities

The Arm subsystem will be in charge of responding to the instructions given from the Movement Program layer. In this case, the arm will move around and get close the envelopes in the environment.

4.1.3 Subsystem Interfaces

Table 2: Arm Subsystem Interfaces

ID	Description	Inputs	Outputs
#1.2	New Instructions for Arm Movement	Instructions	Arm Movement

4.2 Suction Gripper Subsystem

The Suction Gripper is connected to a vacuum and will be used to allow the system to pick up the envelopes. This subsystem responds to Picking inputs by the Program layer

4.2.1 Assumptions

Vacuum will be strong enough to suction an envelope.

4.2.2 Responsibilities

The suction gripper will grip the envelopes with a readable AruCo tag or through Human Assistance. Also, while gripping the envelope, it should not damage the contents inside the envelope. At the same

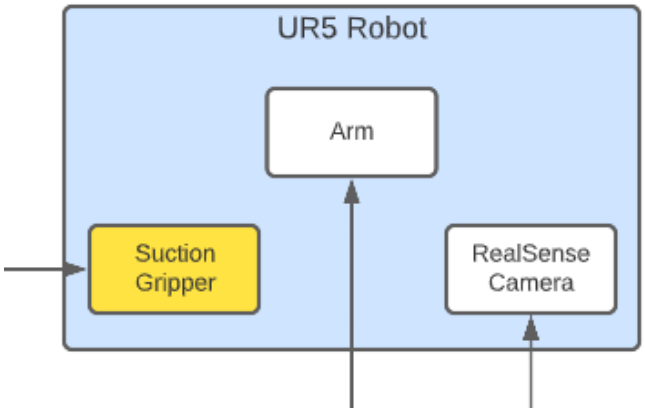

Figure 4: Suction Gripper Subsystem Description Diagram

time, the gripping should not be too light, to prevent the envelope from sliding and falling off of the gripper.

4.2.3 Subsystem Interfaces

Table 3: Suction Gripper Subsystem Interfaces

ID	Description	Inputs	Outputs
#1.1	Conditions for suction gripper to activate	AruCo Detection, Human Assistance	Suction

4.3 Camera Subsystem

The Camera subsystem will be used to scan an AruCo tag on the envelopes. This subsystem interacts with the AruCO detection subsystem in the Vision Program layer.

4.3.1 Assumptions

Camera is able to distinctly identify the AruCo tag programmed.

4.3.2 Responsibilities

The only responsibility that the Camera subsystem has is to take a clear picture of the environment. It will then proceed to send this picture to the AruCo detection subsystem in Vision Program layer.

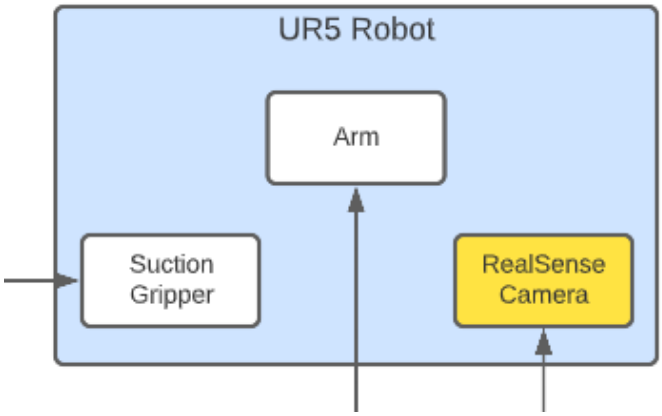

Figure 5: Camera Subsystem Description Diagram

4.3.3 Subsystem Interfaces

Table 4: Camera Subsystem Interfaces

ID	Description	Inputs	Outputs
#1.3	Image of Environment	Instructions to take picture	Picture taken

5 Movement Program Layer Subsystems

This layer contains information of the Movement program, any type of programming that has to do with picking and positions instructions is done here.

5.1 Position Instructions Subsystem

This subsystem contains our main movement functionality. This is the script portion on moving the robot to a certain location.

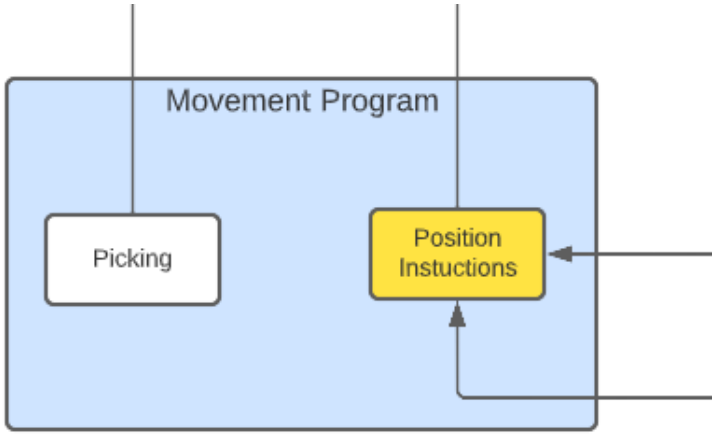

Figure 6: Position Instructions Subsystem Description Diagram

5.1.1 Assumptions

The location that is received from Vision Program layer is reachable and correct.

5.1.2 Responsibilities

This subsystem will send the instructions on where to move, to the Arm subsystem in UR5 Robot layer.

5.1.3 Subsystem Interfaces

Table 5: Position Instructions Subsystem interfaces

ID	Description	Inputs	Outputs
#1.2	Coordinate Instructions send to Arm	Movement Instructions	Arm Movement
#2.1	Coordinates based on detected AruCo tag	Coordinates	Movement Instructions
#3.1	Coordinates based on Operator	Coordinates	Movement Instructions

5.2 Picking Subsystem

This subsystem contains the functionality of when and how our suction will activate to pick up items.

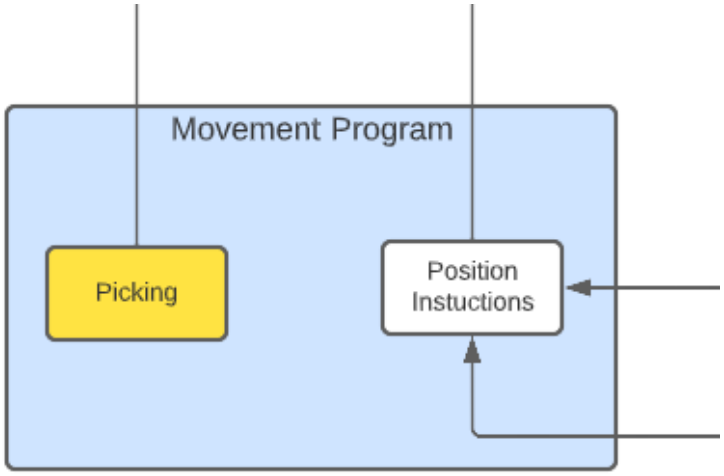

Figure 7: Picking Subsystem Description Diagram

5.2.1 Assumptions

Any item that is being picked up will have a flat surface for complete suction grip and will also be light enough to be picked up. We also assume that it will successfully pick up the object each time.

5.2.2 Responsibilities

This handles picking up an item once it comes in close enough contact.

5.2.3 Subsystem Interfaces

Table 6: Picking Subsystem Interfaces

ID	Description	Inputs	Outputs
#1.1	Conditions for suction gripper to activate	AruCO Detection, Human Assistance	Suction

6 Vision Program Layer Subsystems

This layer contains information of the Vision program, any type of programming that has to do with AruCo detection, converting pixel coordinates to real-world coordinates, and Human Assistance is done here.

6.1 AruCo Detector Subsystem

The AruCo Detector subsystem will allow users to locate the center pixel coordinate of the AruCo tag..

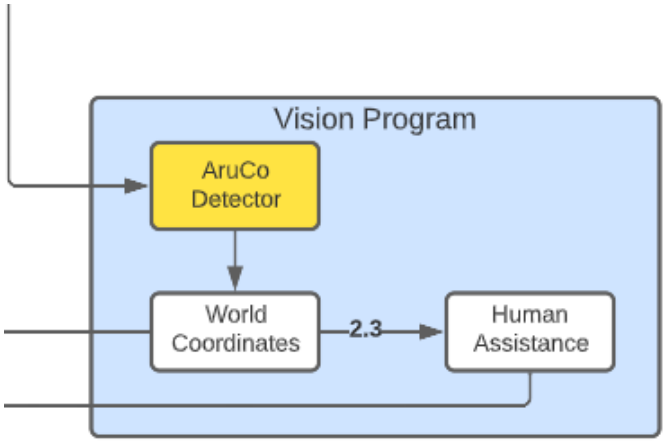

Figure 8: AruCo Detector Subsystem Description Diagram

6.1.1 Assumptions

Subsystem assumes that the image given is clear to see.

6.1.2 Responsibilities

This subsystem will find the coordintnates of the AruCo tag. It will then proceed to send the pixel values to the World Coordinates subsystem to process.

6.1.3 Subsystem Interfaces

Table 7: AruCo Detector Subsystem Interfaces

ID	Description	Inputs	Outputs
#1.3	Image of Environment	Instructions to take picture	Picture taken
#2.2	Identify Pixels	Image of Environment	Pixel Coordinates

6.2 World Coordinates Subsystem

The World Coordinates subsystem will convert given pixel coordinates from AruCo detector into World Coordinates for the Movement Program layer.

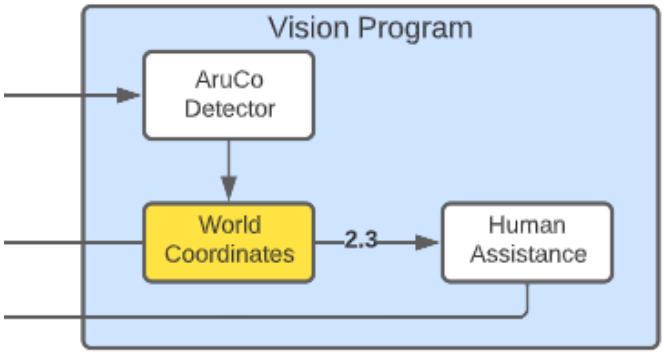

Figure 9: World Coordinates Subsystem Description Diagram

6.2.1 Assumptions

Subsystem assumes that the pixel coordinate come from a valid AruCo tag.

6.2.2 Responsibilities

Once receiving the pixel coordinates from the AruCo detection, World Coordinates subsystem will convert the pixel to world coordinates. After a series of conversion equations, it will then send it to the Movement Program layer.

6.2.3 Subsystem Interfaces

Table 8: World Coordinates Subsystem Interfaces

ID	Description	Inputs	Outputs
#2.1	Coordinates based on detected AruCo tag	Coordinates	Movement Instructions
#2.3	Help needed if no AruCo tag found	No Coordinates	Solution through Human Assistance

6.3 Human Assistance Subsystem

The Human Assistance subsystem will allow users to fix any issues that the AruCO detector is struggling with. This subsystems interacts with operator and Position Instructions in the Movement Program layer.

6.3.1 Assumptions

Subsystem assumes that the item is a valid envelope.

6.3.2 Responsibilities

The Human Assistance subsystem will take an image of the environment. In the case that there are no AruCo tags detected, an operator will feed the subsystem pixel coordinates.The subsystem will then tell

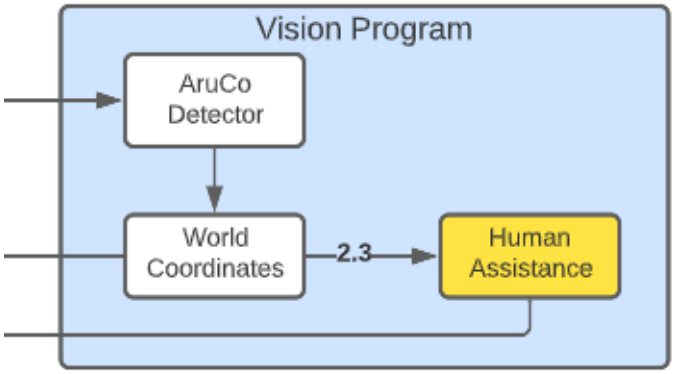

Figure 10: Human Assistance Subsystem Description Diagram

the Movement Program layer the world coordinates.

6.3.3 Subsystem Interfaces

Table 9: Human Assistance Subsystem Interfaces

ID	Description	Inputs	Outputs
#2.3	New World Coordinates to fix error	Image	New Instructions
#3.1	Coordinates based on user input	Coordinates	Movement Instructions